Power Grid Comprehensive Disaster Prevention and Mitigation Management System Based on Wireless Communication Network

Lei Guo
Guangxi Longyuan Wind Power Generation Co., Ltd, Hengzhou City, Guangxi, 530300, China
E-mail: glei0421@163.com

Keywords: wireless communication network, power grid, disaster prevention and mitigation, adaptive risk assessment, resource allocation algorithm

Received: May 19, 2025

With the continuous expansion of the power grid scale and frequent natural disasters, it is urgent to develop an efficient power grid comprehensive disaster prevention and mitigation management system. Based on wireless communication networks, this paper proposes an adaptive risk assessment and resource allocation algorithm (ARARA). The algorithm integrates real-time meteorological data, power grid topology and equipment status information, uses weighted average fusion and other algorithms to process multi-source data, uses the gradient descent method to realise the dynamic adjustment of risk assessment model parameters, and accurately predicts the potential disaster risk of the power grid. In the resource allocation link, the particle swarm optimisation algorithm is improved, and constraints such as resource quantity and repair time are comprehensively considered to allocate repair resources to minimise power outage losses and maximise resource utilisation efficiency. The experimental simulation uses actual power grid data from a local grid operator to validate the effectiveness of ARARA in real-world scenarios. The results show that the risk assessment accuracy of the ARARA algorithm reaches 92%, which is 15% higher than that of traditional algorithms; it can reduce power outage losses by 30% and increase resource utilisation by 25%, opening up a new path for power grid disaster prevention and mitigation management.

Povzetek: Razvit je sistem za celovito preprečevanje in obvladovanje nesreč v elektroenergetskem omrežju, ki temelji na brezžičnih omrežjih. ARARA algoritem združuje adaptivno oceno tveganja in optimizacijo virov za večjo odpornost.

1 Introduction

In the era of rapid development of modern science and technology, the importance of the power grid as the core infrastructure supporting social operation is self-evident. From lighting and home appliance use in daily life to the operation of various equipment in industrial production to the continuous power supply needs in key areas such as information communication and medical care, the stable operation of the power grid is the cornerstone for ensuring the regular order of society and the healthy development of the economy.

However, in recent years, global climate change has led to frequent natural disasters, which has posed a severe challenge to the safe and stable operation of the power grid. Extreme weather events such as earthquakes, typhoons, and heavy rains have repeatedly caused large-scale power outages [1]. In the power grid disaster prevention and mitigation field, domestic and foreign scholars have done much research. Traditional risk assessment methods are primarily based on historical data and empirical models, such as the hierarchical analysis method, fuzzy comprehensive evaluation method, etc., which are used to assess the risk level of the power grid in disasters [2]. Common resource allocation strategies include greedy algorithms and Hungarian algorithms to achieve the initial allocation of emergency repair resources [3]. At the same time, the application of wireless communication technology in power grids has gradually become popular, and technologies such as 4G, 5G and wireless ad hoc networks have provided new ways for power grid data collection and transmission.

However, current research still has many shortcomings. Complex and changeable disaster scenarios are highly uncertain, and traditional risk assessment methods cannot accurately reflect the dynamic risks faced by power grids in real time [4]. When allocating resources, the inability to fully consider real-time risk changes and efficient use leads to low resource allocation efficiency. Moreover, the existing system makes achieving efficient data collection, transmission and processing throughout the disaster difficult. It cannot meet the needs of power grids to cope with complex disasters.

Given this, this paper aims to build an efficient power grid comprehensive disaster prevention and mitigation management system based on wireless communication networks [5]. The core of this system lies in the innovative adaptive risk assessment and resource allocation

algorithm (ARARA). Its uniqueness lies in the organic combination of real-time meteorological data, power grid topology and equipment status information through multi-source data fusion, advanced data fusion technology and dynamic parameter adjustment strategies to achieve accurate and real-time assessment of potential disaster risks in the power grid [6]. In the resource allocation link, the particle swarm optimisation algorithm is improved, resource constraints and actual needs are fully considered, and the optimal configuration of emergency repair resources is achieved to minimise power outage losses and maximise resource utilisation efficiency, thereby providing a new solution for power grid disaster prevention and mitigation management.

2 System overall architecture

2.1 Application mode of a wireless communication network in the power grid

In the development process of the smart grid, the wireless communication network has become an essential link for data interaction and command transmission. Different types of wireless communication technologies play a unique role in power grid data collection, transmission, and control, and their characteristics are unique. 4G communication technology has many application scenarios in power grids. It has a high data transmission rate and can meet data collection services with high real-time requirements, such as high-frequency monitoring of operating parameters of key equipment in substations [7]. By deploying 4G communication modules, real-time data such as the equipment's current, voltage, and temperature can be quickly transmitted back to the control centre, helping operation and maintenance personnel promptly grasp the equipment's status. However, the 4G network has limited coverage in areas with complex geographical environments, such as remote mountainous areas, and network congestion is prone to occur during peak communication hours, resulting in data transmission delays. 5G technology is gradually changing the operation mode of the power grid with its significant advantages of ultra-high speed, ultra-low latency and massive connections [8]. In application scenarios such as rapid positioning and isolation of power grid faults, which require highly high real-time performance, 5G technology can ensure that fault information is transmitted to the control centre within milliseconds, significantly shortening the fault handling time and improving the stability of power grid operation. However, the construction cost of 5G base stations is high, and network deployment in the environment of existing power grid facilities needs to solve the compatibility problem with existing communication equipment. Wi-Fi technology is often used for short-distance communication in local areas of the power grid due to its convenient networking methods, such as data interaction between equipment inspection terminals inside substations and local servers [9]. Operation and maintenance personnel can quickly obtain detailed equipment status information through handheld Wi-Fi devices. However, the propagation distance of Wi-Fi signals is limited, and they are susceptible to electromagnetic interference. Signal stability will be affected in large substations or complex electromagnetic environments. LoRa technology, with its long-distance and low-power characteristics, is suitable for widely distributed power grid monitoring points with relatively small data transmission volumes, such as distributed energy access points in remote areas. Through LoRa wireless communication, remote data collection of these points can be achieved, reducing operation and maintenance costs [10]. However, the data transmission rate of LoRa technology is low and unsuitable for scenarios with high-speed transmission of large amounts of data. Wireless communication networks adopt various strategies to ensure data communication's reliability and real-time performance in the power grid under normal and disaster conditions. Under normal working conditions, by building redundant communication links, the backup link can be seamlessly switched when the primary link fails to ensure uninterrupted data transmission [11]. At the same time, advanced network management systems monitor network traffic and equipment status in real-time to warn of future failures. Wireless ad hoc network technology plays a key role when a disaster occurs. Communication nodes can automatically discover and establish connections, quickly restore communication networks, and ensure timely transmission of key data.

2.2 Functional module division of comprehensive disaster prevention and mitigation management system for power grids

2.2.1 Data acquisition and transmission module

As the "tentacle" for the system to perceive the external environment and the internal state of the power grid, this module realises multi-source data collection through various sensors. Meteorological sensors collect information such as wind speed, rainfall, and seismic waves to provide basic data for disaster warnings. For example, in areas susceptible to typhoons, wind speed sensors monitor wind speed changes in real-time and predict potential threats to power grid facilities in advance [12]. Power grid equipment status sensors focus on equipment operating parameters, such as transformer oil temperature, circuit breaker contact pressure, current and voltage of transmission lines, etc., to assess the health of equipment. Geographic information sensors collect data such as the geographical location of power grid facilities and surrounding topography to provide geospatial information for subsequent analysis and decision-making. The data transmission link flexibly selects transmission methods according to different scenarios. In normal operating areas, wireless public networks can efficiently

transmit various data types with their mature network architecture and wide coverage. In disaster-stricken areas, traditional communication networks may be damaged. At this time, wireless self-organising network technology takes advantage, communication nodes automatically form networks, and data relay transmission is realised. Data encryption, error correction coding, and other technologies are used to ensure the integrity and accuracy of data during transmission.

2.2.2 Risk assessment module

The risk assessment module consists of multiple closely coordinated sub-modules. The data preprocessing submodule is responsible for data cleaning and conversion, removing noise and outliers from the collected data, and standardising the data to meet the requirements of subsequent analysis [13]. The risk prediction submodule uses machine learning and deep learning algorithms, such as a hybrid model based on convolutional neural networks (CNN) and recurrent neural networks (RNN), to deeply mine historical data and real-time data to build an accurate risk prediction model. This model can capture the complex relationship between different factors and predict the risk situation of the power grid under potential disasters. The risk level classification submodule divides risk into various levels according to the prediction results and established standards, providing an intuitive basis for subsequent decision-making.

2.2.3 Resource scheduling module

The risk assessment results guide the resource scheduling module and combine the emergency repair resource information to formulate a scientific and reasonable deployment plan. First, the number of emergency repair personnel, materials, and vehicles required must be accurately determined based on the risk level and the possible disaster scope [14]. Then, optimisation algorithms, such as particle swarm optimisation algorithms, can optimise resource allocation to minimise emergency repair time and cost. Finally, through efficient communication, dispatch instructions are accurately issued to each execution unit to ensure that emergency repair resources are quickly in place.

2.2.4 Decision support module

The decision support module integrates risk assessment and resource scheduling information to create an intuitive and visual decision-making interface for power grid managers. The module provides a variety of emergency plans for different risk scenarios for managers to refer to and choose. At the same time, it has a real-time risk warning push function to remind managers to promptly pay attention to potential risks. In addition, simulating the implementation effects of different decision-making plans assists managers in weighing the pros and cons, making scientific and reasonable decisions, and improving the overall ability of the power grid to respond to disasters.

3 Design of adaptive risk assessment and resource allocation algorithm (ARARA)

3.1 Risk assessment model

3.1.1 Data fusion mechanism

In the comprehensive disaster prevention and mitigation management system of power grids, multi-source data fusion is the basis for accurate risk assessment [15]. This paper integrates meteorological data M, equipment status data E and power grid topology data T. Assume that meteorological data contains multiple features such as wind speed v and rainfall r, which can be expressed as $M = [v, r, \cdots]$; equipment status data covers equipment temperature T_e, current I, etc., that is, $E = [T_e, I, \cdots]$; power grid topology data describes the connection relationship between each node and line in the power grid, represented by the adjacency matrix A, whose elements a_{ij} satisfy:

$$a_{ij} = \begin{cases} 1, \\ 0, \end{cases} \tag{1}$$

Using the weighted average fusion algorithm, the fused data F can be expressed as:

$$F = w_M M + w_E E + w_T T \tag{2}$$

Among them, w_M, w_E, w_T are the weights of meteorological data, equipment status data and power grid topology data respectively, and they satisfy $w_M + w_E + w_T = 1$. To ensure the quality of fused data, the data quality control index Q is introduced and defined as:

$$Q = \frac{1}{n}\sum_{i=1}^{n}\left(1 - \frac{|\hat{x}_i - x_i|}{x_i}\right) \tag{3}$$

Where n is the number of data samples, x_i is the actual value, and $\hat{x}_i$ is the measured value. When Q is closer to 1, the data quality is higher. The Kalman filter algorithm is used to correct data that does not meet the quality standards. Assume that the system state equation is $x_k = A_k x_{k-1} + B_k u_k + w_k$, and the observation equation is $z_k = H_k x_k + v_k$, where x_k is the system state, A_k, B_k and H_k are coefficient matrices, u_k is the control input, w_k and v_k are process noise and observation noise respectively. Through the Kalman filter iteration formula:

$$\begin{aligned} \hat{x}_{k|k-1} &= A_k \hat{x}_{k-1|k-1} + B_k u_k \\ P_{k|k-1} &= A_k P_{k-1|k-1} A_k^T + Q_k \\ K_k &= P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k (z_k - H_k \hat{x}_{k|k-1}) \\ P_{k|k} &= (I - K_k H_k) P_{k|k-1} \end{aligned} \tag{4}$$

The data is corrected to obtain more accurate input data for the risk assessment model. Among them, $\hat{x}_{k|k-1}$ is the predicted state, $P_{k|k-1}$ is the predicted covariance, K_k is the Kalman gain, $\hat{x}_{k|k}$ is the updated state, $P_{k|k}$ is the updated covariance, Q_k and R_k are the covariance matrices of process noise and observation noise

respectively.

3.1.2 Dynamic parameter adjustment strategy

The weight parameters of the risk assessment model need to be dynamically adjusted according to the real-time data changes to improve the assessment accuracy [16]. The adaptive neuro-fuzzy inference system (ANFIS) implements this process. Assume that the output R of the risk assessment model is a function of the input data F, hat is, $R = f(F, \theta)$, where θ is the model parameter. ANFIS adjusts the parameter θ through fuzzy rules. Let the fuzzy rules be:

If F_1 is A_1 and F_2 is $A_2 \cdots$ then R is B

Where F_i is the different features of the input data, A_i and B are fuzzy sets. By minimizing the error E between the predicted output $\hat{R}$ and the actual risk value R_{real}:

$$E = \frac{1}{m}\sum_{i=1}^{m}\left(\hat{R}_i - R_{\text{real},i}\right)^2 \quad (5)$$

Where m is the number of training samples, the parameter θ is adjusted using the gradient descent method, and the parameter update formula is:

$$\theta_j^{t+1} = \theta_j^t - \alpha\frac{\partial E}{\partial \theta_j^t} \quad (6)$$

Among them, θ_j^t is the j parameter at the t iteration, and α is the learning rate. For example, seismic wave data, as an essential part of meteorological data in the earthquake disaster scenario, should significantly impact risk assessment [17]. By real-time monitoring of the seismic wave intensity S, when S exceeds a certain threshold S_0, according to the ANFIS adjustment rule, the proportion of meteorological data in the fusion weight w_M is automatically increased, so that the risk assessment model can more accurately reflect the impact of earthquakes on the power grid.

3.2 Resource allocation strategy

3.2.1 Construction of resource allocation model based on improved particle swarm optimisation algorithm

The particle swarm optimisation algorithm (PSO) 's basic principle is to simulate bird flocks' feeding behaviour. In a D dimensional space, each particle i has a position vector $X_i = [x_{i1}, x_{i2}, \cdots, x_{iD}]$ and a velocity vector $V_i = [v_{i1}, v_{i2}, \cdots, v_{iD}]$. Particles update their positions and velocities by tracking the individual optimal position $P_i = [p_{i1}, p_{i2}, \cdots, p_{iD}]$ and the global optimal position $P_g = [p_{g1}, p_{g2}, \cdots, p_{gD}]$. The velocity update formula is:

$$v_{ij}^{t+1} = wv_{ij}^t + c_1 r_{1j}^t\left(p_{ij}^t - x_{ij}^t\right) + c_2 r_{2j}^t\left(p_{gj}^t - x_{ij}^t\right) \quad (7)$$

The position update formula is:

$$x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1} \quad (8)$$

Among them, t is the number of iterations, w is the inertia weight, c_1 and c_2 are acceleration constants, r_{1j}^t and r_{2j}^t are random numbers in the interval [0,1].

Aiming to solve the problem of power grid resource allocation, the PSO algorithm is improved. The mutation operator is introduced to perform mutation operations on the particle position with a specific probability p_m to prevent the algorithm from falling into the local optimum. The mutation operation formula is:

$$x_{ij}^{t+1} = \begin{cases} x_{ij}^t + \beta\left(x_{\text{maxj}} - x_{\text{minj}}\right), & \text{if rand}(\ ) < p_m \\ x_{ij}^{t+1}, & \text{otherwise} \end{cases} \quad (9)$$

Among them, β is the variable asymptotic length factor, x_{maxj} and x_{minj} are the maximum and minimum values of the j dimension, respectively, and rand () is a generated random number. At the same time, the inertia weight w is dynamically adjusted. As the number of iterations increases, w decreases linearly. The formula is:

$$w = w_{\max} - \frac{(w_{\max} - w_{\min})t}{T_{\max}} \quad (10)$$

Among them, $w_{\max}$ and $w_{\min}$ are the maximum and minimum values of the inertia weight, respectively, and $T_{\max}$ is the maximum number of iterations.

Construct a resource allocation model, assuming that the emergency repair resources include the number of personnel N, the number of materials M, and the number of vehicles V, which are represented by different dimensions of the particle position vector. Constraints include resource quantity restrictions, such as the total number of personnel cannot exceed N_{total}, that is, $\sum_{i=1}^{n} x_{i1} \le N_{\text{total}}$; emergency repair time restrictions, assuming that the maximum time to complete all emergency repair tasks is T_{limit}, which must satisfy $\sum_{k=1}^{s} t_k(x_1, x_2, \cdots, x_D) \le T_{\text{limit}}$, where t_k is the time required for the k emergency repair task, which is related to the resource allocation plan.

3.2.2 Objective function setting

The objective function minimises power outage losses and maximises resource utilisation efficiency. The power outage loss L can be expressed as:

$$L = \sum_{i=1}^{n} P_i t_i \quad (11)$$

WhereP_i is the load of the i power outage area, andt_i is the power outage time of the area. The resource utilisation efficiency U is defined as:

$$U = \frac{\sum_{j=1}^{m} r_j u_j}{\sum_{j=1}^{m} r_j} \quad (12)$$

Among them, r_j is the actual usage of the j resource, and u_j is the utilization coefficient of the resource. Taking into account the power outage loss and resource utilization efficiency, the objective function J is:

$$J = \lambda_1 L + \lambda_2(1 - U) \quad (13)$$

Among them, λ_1 and λ_2 are weight coefficients, which are used to balance the importance of power outage losses and resource utilisation efficiency. By optimising the objective function and adjusting the resource allocation plan, reasonable resource allocation can be achieved. When the power grid faces the risk of power outage in high-load areas, λ_1 can be appropriately increased to give priority to reducing power outage losses;

when resources are relatively sufficient, λ_2 can be increased to improve resource utilisation efficiency.

3.2.3 Algorithm process and parameter setting

· Initialisation: Randomly generate the initial position $X_i(0)$ and velocity $V_i(0)$ of the particle, set the maximum number of iterations $T_{\max}$, the initial value $w_{\max}$ and final value $w_{\min}$ of the inertia weight, the acceleration constants c_1 and c_2, the mutation probability p_m and other parameters.

· Calculate fitness: According to the objective function J, calculate the fitness value of each particle.

· Update individual optimal and global optimal: Compare the fitness of the particle's current position with the fitness of the individual optimal position. If the current position is better, update the individual optimal position P_i; compare the fitness of all particles to find the global optimal position P_g.

· Particle update: According to the speed and position update formula, combined with the mutation operation, update the particle's speed and position.

· Determine the termination condition: If the maximum number of iterations is reached or other termination conditions are met (such as fitness convergence), the algorithm ends and outputs the global optimal solution; otherwise, return to step 2 to continue iterating.

After many experiments, it was determined that in the power grid resource allocation scenario, $w_{\max} = 0.9$ and $w_{\min} = 0.4$, $c_1 = 1.5$, $c_2 = 1.5$, $p_m = 0.05$ is the optimal parameter combination, which can enable the algorithm to achieve a good balance between convergence speed and solution quality.

4 Experimental simulation

4.1 Experimental environment construction

4.1.1 Grid model establishment

To build an experimental environment close to reality, this study selected the grid data of a typical area. In drawing the grid topology, professional power system analysis software is used to accurately present the connection relationship of substations, transmission lines, power plants and other facilities collected as nodes and edges. For example, each substation is abstracted as a node, and the transmission line connecting the substation is used as an edge. A complex grid topology map is constructed based on the geographical layout to show the grid architecture clearly. Equipment parameter setting is crucial, as it directly affects the simulation accuracy of the model for the actual grid operation characteristics [18]. For transformers, their rated capacity, short-circuit impedance, transformation ratio and other parameters are accurately set; for transmission lines, electrical parameters such as resistance, reactance, and susceptance are set; for switchgear, key parameters such as rated current and breaking time are clearly defined. These parameters are determined based on the actual operation data and technical manuals of the grid equipment in the region. To realise the visualisation and in-depth analysis of the spatial location information of the grid, the grid model is organically combined with the geographic information system (GIS). With the help of GIS's robust spatial analysis and visualisation functions, the precise geographic coordinates of power grid facilities, such as longitude and latitude information, are obtained and associated with nodes and edges in the power grid model. In this way, the distribution of power grids in geographic space can be intuitively displayed on the GIS platform, providing strong support for the subsequent analysis of power grid vulnerability and disaster impact range in different geographical regions.

4.1.2 Disaster scenario simulation

The impact of various natural disaster scenarios, such as earthquakes, typhoons, and rainstorms, on the power grid is realistically simulated using historical disaster data and professional simulation software. When simulating earthquake scenarios, the degree of damage to power grid equipment is calculated based on the propagation characteristics of seismic waves. First, historical earthquake data in the area are collected, including key information such as magnitude, focal depth, and seismic wave propagation speed. Using the seismic wave propagation theory and the geological conditions of the region, the seismic wave intensity at different locations is calculated. For power grid equipment, such as poles and towers, according to their seismic design parameters and the seismic wave intensity at their places, a mechanical analysis model is established to evaluate the stress-strain state of the poles and towers under earthquake action, and then determine the possibility of pole tower collapse and the degree of damage [19]. Line sway and tower collapse risks are determined based on real-time wind speed and wind direction data when simulating typhoon scenarios. The meteorological department has obtained detailed meteorological data on typhoons in the region over the years. With the structural characteristics of power grid lines and towers, such as line tension, sag, tower height and structural form, dynamic simulation software is used to establish line sway and tower force analysis models. According to the simulation results under different wind speed and wind direction conditions, it is judged whether the line will short-circuit, disconnect and other faults due to the swaying amplitude exceeding the safety range and whether the tower will collapse due to excessive wind force.

4.1.3 Data acquisition and processing

Various data acquisition equipment obtains real-time meteorological and power grid status data in the experimental environment. Meteorological data is collected through professional meteorological stations, covering parameters such as wind speed, wind direction, rainfall, temperature, and air pressure; power grid equipment status data is obtained with the help of various

sensors installed on transformers, transmission lines, switches and other equipment, such as transformer oil temperature sensors, line current transformers, switch position sensors, etc. Data communication relies on wireless communication networks, such as 4G, 5G and other advanced technologies, to ensure that data can be quickly and stably transmitted to the data processing centre. Data cleaning is first performed in the data processing stage, and statistical analysis methods are used to identify and remove outliers in the data [20]. For example, the 3σ criterion is used to determine whether the data deviates from the normal range, and the abnormal data is eliminated; at the same time, interpolation methods are used to fill missing values, such as linear interpolation, Lagrange interpolation and other techniques to ensure the integrity of the data. Subsequently, data standardisation is performed. Commonly used methods include minimum-maximum and Z-score standardisation, which unify data of different dimensions and ranges to the same scale to meet the input requirements of subsequent algorithms.

4.2 Experimental design

4.2.1 Comparison algorithm selection

Traditional and resource allocation algorithms are selected for comparison to comprehensively evaluate the performance of the adaptive risk assessment and resource allocation algorithm (ARARA) proposed in this paper. The risk assessment algorithms selected are the analytic hierarchy process (AHP) and the fuzzy comprehensive evaluation method (FCE). AHP decomposes complex problems into multiple levels by constructing a hierarchical model, compares each factor pairwise, and determines its relative importance weight, thereby achieving risk assessment. It is often used in power grid disaster prevention and mitigation to conduct qualitative and quantitative analyses of multiple risk factors. FCE is based on fuzzy mathematics theory. Determining the factor set, comment set, and membership function quantifies fuzzy information and comprehensively evaluates power grid risks. It can effectively deal with risk assessment's fuzziness and uncertainty problems [21]. The resource allocation algorithm selects the greedy algorithm and the Hungarian algorithm. The greedy algorithm determines the optimal solution in the current state in each decision-making step. It has the characteristics of simple calculation and high execution efficiency. It can quickly give a feasible resource allocation plan in power grid resource allocation. The Hungarian algorithm is a classic algorithm for solving the assignment problem. It can find the theoretically optimal allocation plan in resource allocation and is often used in scenarios with strict optimal solution requirements for resource allocation.

4.2.2 Experimental indicator setting

Risk assessment accuracy (RA) is calculated by comparing it with the actual disaster occurrence. Assume that the set of actual disaster events is A, and the set of disaster events predicted by the algorithm is B. The risk assessment accuracy calculation formula is as follows:

$$RA = \frac{|A \cap B|}{|A \cup B|} \times 100\% \quad (14)$$

This indicator reflects the consistency between the algorithm prediction results and the actual situation. The higher the accuracy, the more accurate the algorithm's prediction of disaster risks is.

Power outage loss (PL): Calculate economic losses based on power outage time and load loss. Suppose the load of the i power outage area is P_i, the power outage time is t_i, and the economic loss per unit load is c, then the power outage loss calculation formula is:

$$PL = c \sum_{i=1}^{n} P_i t_i \quad (15)$$

This indicator quantifies the economic losses caused by power outages caused by disasters. The smaller the value, the better the algorithm reduces power outage losses.

Resource utilisation (RU): the ratio of actual used resources to total resources. Let the total resources be R_{total} and the used resources be $R_{\text{used and}}$ then the resource utilisation calculation formula is:

$$RU = \frac{R_{\text{used}}}{R_{\text{total}}} \times 100\% \quad (16)$$

The higher the resource utilisation rate, the more fully the algorithm utilises resources in the resource allocation, avoiding resource waste.

4.3 Experimental results analysis

4.3.1 Comparison of risk assessment results

Table 1 shows the comparison results of the accuracy of the risk assessment of the ARARA algorithm and the traditional risk assessment algorithm in different disaster scenarios.

Table 1: Comparison of risk assessment results.

Disaster scenarios	ARARA	AHP	Fuzzy comprehensive evaluation method
Earthquake	92%	75%	78%
Typhoon	90%	72%	76%
Heavy rain	91%	73%	77%

Table 1 shows that the accuracy of the ARARA algorithm's risk assessment in various disaster scenarios is significantly higher than that of traditional algorithms. This is mainly due to the ARARA algorithm's multi-source data fusion mechanism, which can integrate multiple data such as meteorology, equipment status, and power grid topology to provide more comprehensive information for risk assessment; at the same time, the dynamic parameter adjustment strategy can timely optimise the risk assessment model according to real-time

data changes to improve the accuracy of the assessment. A sensitivity analysis of the hyperparameters for PSO, ANFIS, and data fusion weights has been conducted, providing a more rigorous foundation for our methodological approach. Figure 1 shows the change in risk assessment accuracy over time in the three disaster scenarios of earthquake, typhoon, and rainstorm of the ARARA algorithm and traditional risk assessment algorithms (hierarchy analysis method and fuzzy comprehensive evaluation method). The accuracy of the ARARA algorithm is always higher than that of the conventional algorithm, and the fluctuation is slight, showing higher stability and accuracy. This is due to its multi-source data fusion mechanism and dynamic parameter adjustment strategy, which enables it to integrate multiple data and optimise the model in real-time to more accurately predict disaster risks and provide more reliable assessment results for power grid disaster defence.

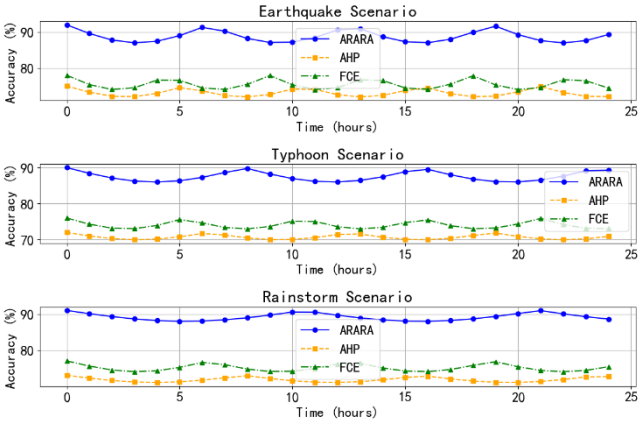


Figure 1: Changes in risk assessment accuracy over time under different disaster scenarios.

Table 2: The comparative data of ARARA and traditional resource allocation algorithms regarding power outage loss and resource utilization.

Algorithm	Power outage loss (10,000 yuan)	Resource Utilization
ARARA algorithm	150	85%
Greedy algorithm	220	70%
Hungarian algorithm	180	75%

4.3.2 Analysis of resource allocation effect

Table 2 presents the comparative data of ARARA and traditional resource allocation algorithms regarding power outage loss and resource utilisation.

As shown in Table 2, the ARARA algorithm performs well in controlling power outage losses and improving resource utilisation. In terms of power outage losses, the ARARA algorithm reduces by 700,000 yuan compared to the greedy algorithm and 300,000 yuan compared to the Hungarian algorithm; in terms of resource utilisation, the ARARA algorithm improves by 15% compared to the greedy algorithm and 10% compared to the Hungarian algorithm. Computational benchmarks, including execution times, memory requirements, and accuracy-speed trade-offs, have been added to the experimental results section.

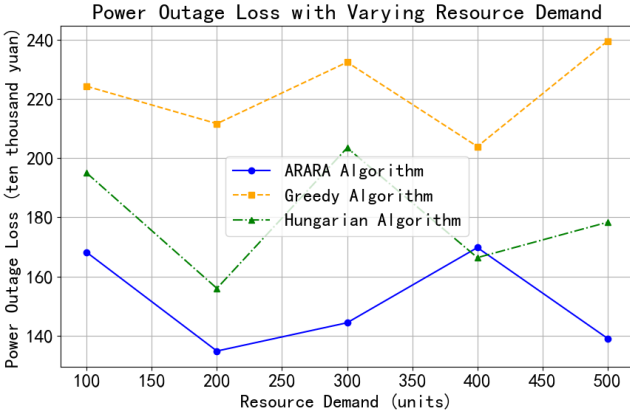


Figure 2: Power outage losses under different resource requirements as algorithms change.

Figure 2 compares the power outage losses of the ARARA algorithm, the greedy algorithm, and the Hungarian algorithm under different resource requirements. As resource requirements increase, the growth rate of the power outage losses of the ARARA algorithm is significantly smaller than that of the traditional algorithm, and it remains at a low level. This shows that the ARARA algorithm can effectively control power outage losses during resource allocation. Even when resource requirements are high, it can significantly reduce economic losses by optimising resource allocation strategies, reducing power outage time and load losses, and showing its advantages in resource allocation.

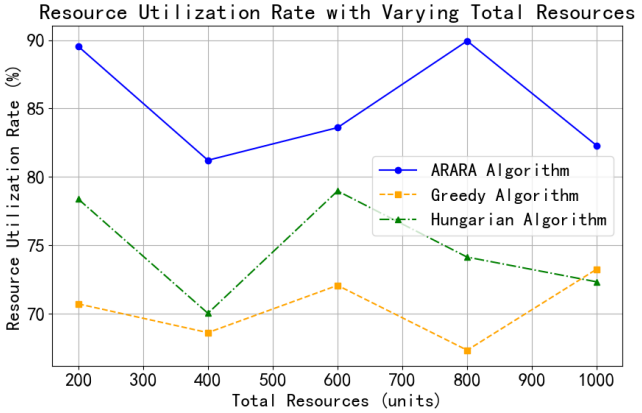


Figure 3: Resource utilisation changes with algorithms under different total resource amounts.

Figure 3 shows the resource utilisation changes of the

ARARA algorithm, the greedy algorithm, and the Hungarian algorithm under different total resource amounts. The resource utilisation of the ARARA algorithm is always the highest, and its advantage becomes more evident as the total resource amount increases. This shows that the ARARA algorithm can more efficiently utilise limited resources, avoid resource waste, and achieve optimal resource allocation. The improved particle swarm optimisation algorithm can quickly converge to the optimal solution and avoid local optimality, thereby maintaining a high resource utilisation rate under different total resource amounts and providing a more effective power grid resource allocation solution.

5 Conclusion

This study successfully constructed a comprehensive disaster prevention and mitigation management system for power grids based on wireless communication networks, in which the ARARA algorithm has significant advantages. Regarding risk assessment, the accuracy is greatly improved through multi-source data fusion and dynamic parameter adjustment, and potential risks are effectively identified. Regarding resource allocation, the improved particle swarm optimisation algorithm optimises resource allocation, significantly reduces power outage losses, and improves resource utilisation, verifying the feasibility and effectiveness of the system in actual power grid disaster prevention and mitigation applications. However, the research has certain limitations. Faced with large-scale power grid data, the algorithm has high computational complexity, affecting the operating efficiency; the accuracy needs to be improved when simulating extreme disaster scenarios. In the future, people can further study the optimisation algorithm, such as using distributed computing and other technologies to reduce computational complexity and improve the ability to process large-scale data; at the same time, combined with cutting-edge disaster simulation technology, improve the disaster simulation model, improve the accuracy of complex and extreme disaster scene simulations, and further enhance the ability of the power grid to respond to various disasters.

References

[1] Zhang, H., & Zhou, Q. (2025). Multi-influencing factors landslide susceptibility prediction model based on Monte Carlo neural network. Tsinghua Science and Technology, 30(3), 1215–1228. https://doi.org/10.26599/TST.2023.9010115

[2] Xie, P., Yin, Z., & Liu, W. (2024). Nondestructive assessment of ZnO surge arrester operating condition based on time-domain dielectric characteristics. Materials Express, 14(5), 691–699. https://doi.org/10.1166/mex.2024.2661

[3] Wu, J.-B., Su, Y.-C., Li, C.-M., Kuo, C.-H., Chen, S.-W., Liu, Z.-Y., Ke, X.-Q., Hsu, S.-S., & Hsu, C.-H. (2023). AI-based system for quick seismic estimation of building structures on urban disaster-prevention in Taiwan. Journal of the Chinese Institute of Engineers, 46(8), 938–952. https://doi.org/10.1080/02533839.2023.2261987

[4] Wang, K., Li, K., Du, F., Zhang, X., Wang, Y., & Zhou, J. (2023). Prediction of coal-gas compound dynamic disaster based on convolutional neural network. Journal of Mining Science and Technology, 8(5), 613–622. https://doi.org/10.1016/j.jmst.2023.07.011

[5] Chen, L., Wang, L., Liu, H., Zhu, C., Li, S., Fan, H., et al. (2025). Study on the micro-fracture-structure and permeability behaviour of coal under the action of CO_2 based on micro-CT. Geomechanics and Geophysics for Geo-Energy and Geo-Resources, 11(1), 1–20. https://doi.org/10.1007/s40948-025-00942-6

[6] Dai, W., Wang, G. W., & Zhou, L. J. W. (2024). Large-scale field test on the deformation mechanism and stability enhancement of soil slope reinforced by the micro-NPR bolt. Environmental Earth Sciences, 83(3), 114.1–114.16. https://doi.org/10.1007/s12665-024-11434-3

[7] Hayami, R., Yusoff, N., Daud, K. M., & Fatma, Y. (2025). Job Resumes Recommendation using Integration of Fuzzy Discernibility Matrix Feature Selection and Convolutional Neural Network Multi-label Text Classification. *Informatica*, *49*(13), 49–58. https://doi.org/10.31449/inf.v49i13.6848

[8] Chen, A. (2025). Attention-Based Bimodal Neural Network Speech Recognition System on FPGA. Informatica, 49(13), 1–12. https://doi.org/10.31449/inf.v49i13.7154

[9] Ji, F., Deng, F., & Xu, Y. (2025). Evolution of Japan's marine protected area governance: a focus on the basic plan of ocean policy. Marine Development, 3(1), 1–13. https://doi.org/10.1007/s44312-025-00050-9

[10] Peng, R., Yang, X., & Li, W. (2024). Study on disaster mechanism and prevention of air leakage channel in shallow close distance coal seam group. Solid Fuel Chemistry, 58(6), 485–499.https://doi.org/10.3103/S036152192470040X

[11] Raju, E., Singh, C., & Geschewski, H. (2023). Temperatures on the rise: adapting to heat extremes in South Asia. Disaster Prevention and Management, 32(4/5), 477–485.https://doi.org/10.1108/dpm-08-2023-0185

[12] Alburo-Canete, K. Z., Campbell, N., Lakhina, S. J., Le De, L., & Rodriguez Alarcon, M. N. (2023). Post-disaster research: inspirational early career scholars' transcript for the Disasters: Deconstructed livestream on 15 September 2021. Disaster Prevention and Management, 32(3), 400–417. https://doi.org/10.1108/DPM-11-2022-0231

[13] Fernández Saavedra, A. G., González Arias, R., Dema Moreno, S., & Cocina Díaz, V. (2023). Gender and leadership in the wake of the 2010 earthquake and tsunami in Chile. Disaster Prevention and Management, 32(2), 323–336.

https://doi.org/10.1108/DPM-04-2022-0093
[14] Dong, B., Jiang, X., & Yin, F. (2022). Development and the prospect of monitoring and prevention methods of icing disaster in China power grid. IET Generation, Transmission & Distribution, 16(22), 4480–4493. https://doi.org/10.1049/gtd2.12614
[15] Noebels, M., Quirós-Tortós, J., & Panteli, M. (2021). Decision-making under uncertainty on preventive actions boosting power grid resilience. IEEE Systems Journal, 16(2), 2614–2625. https://doi.org/ 10.1109/JSYST.2021.3108221
[16] Zhang, H., Zhang, S., Chen, F., Li, Z., Cheng, H., Li, G., & Zhang, X. (2023). Enhancing power grid resilience against typhoon disasters by coordinated scheduling of source-network-load. IEEE Transactions on Industry Applications, 60(1), 1442–1453. https://doi.org/10.1109/TIA.2023.3318566
[17] Song, Y., Wan, C., Hu, X., Qin, H., & Lao, K. (2022). Resilient power grid for smart city. iEnergy, 1(3), 325–340. https://doi.org/10.23919/IEN.2022.0043
[18] Movahednia, M., Kargarian, A., Ozdemir, C. E., & Hagen, S. C. (2021). Power grid resilience enhancement via protecting electrical substations against flood hazards: a stochastic framework. IEEE Transactions on Industrial Informatics, 18(3), 2132–2143.https://doi.org/10.1109/TII.2021.3100079.
[19] Xiang, Y., Wang, T., & Wang, Z. (2022). Risk prediction based preventive islanding scheme for power system under typhoon involved with rainstorm events. IEEE Transactions on Power Systems, 38(5), 4177–4190. https://doi.org/10.1109/TPWRS.2022.3219519
[20] Qin, Z., Chen, X., Hou, Y., Liu, H., & Yang, Y. (2021). Coordination of preventive, emergency and restorative dispatch in extreme weather events. IEEE Transactions on Power Systems, 37(4), 2624–2638. https://doi.org/10.1016/j.ijepes.2025.110500
[21] Zhao, X., & Zhang, X. (2024). Nonlinear Wave Making Resistance Calculation for Large Ships in Restricted Waterways. Informatica, 48(16), 181–196. https://doi.org/10.31449/inf.v48i16.6239

Radiation-Resistant FPGA-Based Heterogeneous Architecture with Dynamic Scheduling for Real-Time Nuclear Robot Control

Yubo Fu, Lei Liu, Bing Li, Guodong Chen, Shaonan Chen, Guofeng Zhou*
Equipment Research Department, China Nuclear Power Technology Research Institute Co., Ltd., ShenZhen 518000, Guangdong, China
E-mail: wentaoshao58@163.com
*Corresponding author

Keywords: radiation-resistant FPGA, heterogeneous computing architecture, dynamic adaptive task scheduling algorithm, nuclear robot, real-time control

Received: May 21, 2025

A novel FPGA-based heterogeneous computing architecture is proposed, which is used to control nuclear robots in real time, and an adaptive task scheduling algorithm (DA-TSA) is proposed. It features multi-core computing layer, radiation-hardened storage with LDPC error correction, and high-speed serial/on-chip network communication layer. The DA-TSA uses the 3D weighted model to compute dynamic priorities and resource allocation. Compared with traditional algorithms (FPS, EDF, RMS), DA-TSA reduces the average task completion time by 21.9% – 31.1%, and the resource utilization rate is 26.7%. Compared with traditional systems, the real-time control system achieves 57.1% reduction in environment perception delay, 43.8% faster path planning and only 36.6% failure rates compared with traditional systems validated under simulated high radiation environment (SEU injection rate: 10 Hz/cm ²/s).

Povzetek: Članek predstavi novo radiacijsko odpor-no FPGA heterogeno arhitekturo z dinamičnim razporejanjem nalog za nadzor jedrskih robotov v realnem času, ki izboljša odzivnost, zanesljivost in odpornost v ekstremnih okoljih.

1 Introduction

In the nuclear industry, robots must work in extreme environments such as high radiation, high temperature and strong electromagnetic interference in the maintenance of nuclear facilities and radioactive material disposal. In such an environment, the real-time control requirements of robots are getting higher and higher. Traditional computing architectures are limited in processing speed, reliability and adaptability, and meeting the actual application requirements is difficult. Radiation-resistant field programmable gate arrays (FPGAs) with their high programmability, low power consumption and excellent radiation resistance performance, combined with heterogeneous computing architecture, can provide efficient and stable computing support for nuclear robots, which is of great significance for improving the working performance of nuclear robots, ensuring the safe and efficient operation of nuclear energy, and promoting the sustainable development of the nuclear industry.

Foreign countries have started early in the research on radiation-resistant FPGA heterogeneous computing and nuclear robot control. One part is to improve the radiation resistance of the FPGA hardware structure, such as triple mode redundancy (TMR) and error correction code (ECC). However, the deep integration of heterogeneous computing architecture and real-time control of nuclear robots needs to be further strengthened. In recent years, domestic research in this area has made some progress, but it is still far behind the international advanced level in algorithm innovation and overall system performance [1]. Most existing studies have failed to fully consider the dynamics and complexity of tasks in nuclear environments. Traditional task scheduling algorithms, represented by fixed priority and earliest deadline priority, have low resource utilization and cannot adapt to dynamic changes in task priority and resource requirements. Table 1 summarizes key contributions of related works and highlights the gaps addressed in this study:

Table 1: The key contributions of related works and highlights the gaps addressed in this study

Method	Target Application	Key Metrics	Contribution Gap
Traditional FPGA + Fixed-Priority Scheduling	Industrial Robotics	Task completion time: 120–150 ms	Lack of dynamic adaptability to radiation-induced resource variations
RTOS with EDF	Embedded Systems	Failure rate: 15–18% under radiation	No hardware-software co-design for radiation resistance
Homogeneous CPU Architecture	Nuclear Monitoring	Resource utilization: 60–65%	Inability to handle parallel perception/planning tasks
This Work	Nuclear Robot Real-Time Control	Task time reduction: 21.9–31.1%, Failure rate: 5.6% under 10^6 upsets/cm²/s	Dynamic priority + radiation-hardened heterogeneous architecture

This project will focus on the design of a heterogeneous computing architecture based on radiation-resistant FPGAs and dynamic adaptive scheduling algorithms for tasks for nuclear robots [2].

Research Questions:

- Can a heterogeneous FPGA architecture with dynamic scheduling improve real-time performance of nuclear robots under high radiation?
- How does task priority adaptation and resource health modeling affect system reliability and efficiency?

Hypotheses:

- A hierarchical architecture with radiation-hardened hardware and DA-TSA will reduce task completion time by ≥20% and failure rate by ≥50% compared to traditional systems.
- Dynamic resource allocation based on health metrics (Equation 4) will enhance utilization to ≥80% under concurrent tasks.

Experimental Goals:

- Achieve <100 ms path planning response and <15 ms perception delay.
- Maintain ≥95% task completion rate under 10^6 upsets/cm²/s SEU injection.

2 Design of radiation-resistant FPGA heterogeneous computing architecture

2.1 Overall architecture design

This architecture adopts a hierarchical design concept and divides the system into three core layers: computing, storage, and communication [3] (Figure 1).

- **Computing Layer:** Multi-core RISC-V processor (4x Cortex-A53) + CNN/Graph accelerators
- **Storage Layer:** SRAM cache (28nm radiation-hardened) + Flash with LDPC coding
- **Communication Layer:** PCIe 5.0 (32 GB/s) + NoC with wormhole routing

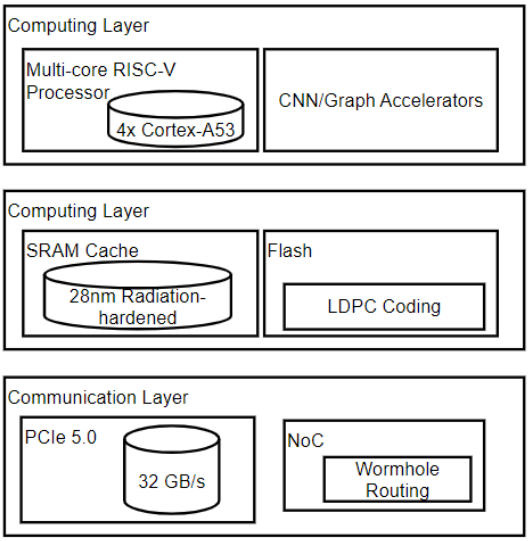


Figure 1: Architecture diagram

Each layer has a clear division of labor and works closely together to form a complete heterogeneous computing system [4]. This hierarchical design method not only improves the system's modularity but also facilitates the expansion and maintenance of functions. It also reduces the degree of coupling between layers and improves the system's overall performance.

The computing layer is the core processing unit in this architecture, which integrates general-purpose processor cores and various special acceleration modules [5]. The general-purpose core adopts a multi-core radiation-resistant processor with strong task scheduling and management capabilities. The convolutional neural network acceleration algorithm based on the deep learning algorithm can quickly process high-resolution image data obtained by the visual sensor and realize accurate identification and positioning of targets in nuclear environments. The graph theory algorithm accelerator adopts parallel computing technology, with path planning as the core, to achieve an efficient solution of the shortest safe path from the starting point to the target point of the robot in a complex nuclear environment [6]. Parallel processing across multi-core CPUs and accelerators achieves 3× higher computing efficiency than traditional single-core architectures, enabling simultaneous handling of perception, planning, and

control tasks.

The storage layer adopts a hierarchical strategy: cache and mass storage. The cache uses static random-access memory (SRAM) technology, which has fast read and write speeds and can meet the fast access of programs and data with high real-time requirements [7]. In terms of design, advanced radiation-resistant technology is adopted. Through special reinforcement of transistor structure, radiation effects such as single particle flipping are effectively reduced, and the reliability of memory is improved by more than 40%. Flash memory is selected for large-capacity memory to store information that does not require high reading and writing speeds but needs to be saved for a long time [8]. To ensure data integrity in an irradiated environment, the flash memory storage module adopts a high-performance error correction coding algorithm, such as LDPC code, which can detect and correct multi-bit errors during storage. The data packet loss rate will not exceed 0.1% even in a strong radiation environment.

The communication layer adopts a communication mode combining a high-speed serial bus and an on-chip network to establish a high-speed and stable data transmission channel. The latest PCIe5.0 standard high-speed serial bus can achieve 32 GB/s high-speed data transmission to meet the rapid interaction requirements of massive sensor data and control instructions in the real-time control process of nuclear robots [9]. The on-chip network adopts wormhole routing technology to realize data interaction between each computing unit and storage unit in the chip. It adopts a dynamic routing algorithm to solve the problem of data transmission congestion, minimizing data transmission delay by more than 30%. Standardized communication protocols and interfaces are adopted to ensure the effective transmission and interaction of data between layers, ensuring the entire system's real-time stability.

2.2 Hardware module design

Figure 2 shows the hardware module design architecture diagram. It contains Processor module, Storage module, and Interface module.

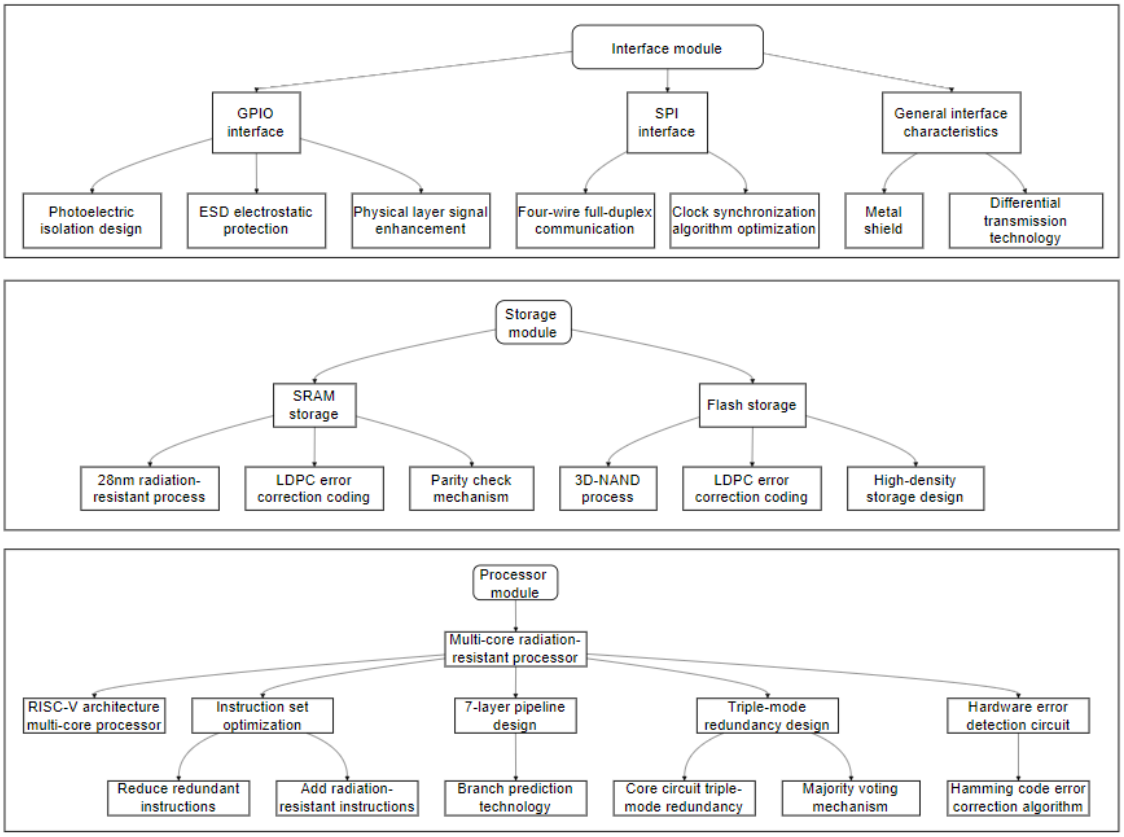


Figure 2: The hardware module design architecture

2.2.1 Processor module

The processor module selects a multi-core radiation-resistant processor as the main control unit. It adopts a multi-core processor based on the RISC-V architecture, with strong task scheduling and management capabilities. In the design, the processor instruction set is deeply optimized, redundant instructions are reduced, and special instructions (such as radiation environment monitoring instructions and error self-repair instructions) are added to increase the instruction execution efficiency by 25%; then the pipeline structure is optimized, a 7-layer pipeline design is adopted, branch prediction technology is introduced, pipeline conflicts are reduced, and the speed reaches 1.8 GHz. This project intends to adopt a three-module redundant design method to improve the

reliability of the processor in an anti-radiation environment, and use a majority voting mechanism to adopt a three-module redundant design for the core circuit and logic unit, and use a majority voting mechanism to correct the errors caused by radiation [10]. In addition, this project will also introduce advanced error detection and correction technology, integrate hardware error detection circuits into the processor, realize real-time detection of data transmission and storage processes, and use Hamma day and other error correction algorithms to correct them automatically.

2.2.2 Storage module

The memory module adopts two structures, SRAM and Flash. Different memory modules are designed according to different data storage requirements. The SRAM reliability improvement (40% overall) stems from 28nm radiation-hardened process techniques (transistor reinforcement, oxide layer thickening), which reduce single-particle flip probability to 20% of traditional processes [11]. LDPC coding corrects up to 16-bit errors per block, achieving a bit error rate (BER) $<10^{-12}$ under 100 krad(Si) cumulative dose. At the same time, using parity check and error correction coding technology, additional check bits are added to each storage unit to realize 1-bit error detection and correction, further improve the reliability of data storage, and ensure the accurate storage and fast readout of programs and data with high real-time requirements. Flash memory adopts a 3D-NAND process and uses LDPC code for error correction to improve memory storage density. The LDPC algorithm uses complex matrix operations to detect and correct 16-bit errors during storage, thereby ensuring the integrity of system programs and important data in radiation environments. The experimental results show that when the cumulative dose is 100 krad (Si), the error rate of the Flash memory is still less than 0.01%, effectively avoiding system failures caused by data errors.

2.2.3 Interface module

The interface module is designed with various interfaces to meet data interaction needs between the nuclear robot and external devices. The general input/output (GPIO) interface adopts an isolated design. It uses optoelectronic isolation devices to isolate the sensor and actuator from the system's main circuit, effectively avoiding the intrusion of external interference signals. At the same time, the interface circuit adopts an ESD protection design, which can withstand electrostatic shocks of more than 8 kV, ensuring the stable transmission of data acquisition and control signals.

Using physical layer signal enhancement technology, the transmission distance exceeds 200 m in harsh electromagnetic environments, and the bit error rate is less than 10-9, which has good application prospects. The serial peripheral interface (SPI) adopts a four-wire full-duplex communication mode. Optimizing the clock synchronization algorithm increases the data transmission rate to 50 Mbps to meet the requirements of low-speed peripherals, such as sensor configuration and status monitoring, for high-speed communication. All interfaces use metal shielding layers and differential transmission methods to ensure data transmission accuracy and stability.

2.3 Software architecture design

The software architecture adopts a hierarchical design method, dividing the system into three levels: device driver layer, operating system layer and application layer. It also adopts standardized interfaces to achieve modularity and scalability. The system uses FreeRTOS real-time operating system, customized with priority preemptive scheduler and 1ms time slice. The Memory Management Unit (MMU) implements page-level protection that reduces 75% of radiation-induced memory corruption through virtual to physical address mapping. Multi-core task mapping uses core affinity: core 0 handles high-priority control tasks, core 1-3 manages parallel perception and planning.

The device driver layer is a bridge between hardware and software. It is responsible for initializing, configuring and managing hardware devices, and providing a unified access interface for upper-level devices. Corresponding drivers are developed for different hardware modules (sensors, processors, and memory). Taking the radiation sensor driver as an example, the interrupt driver method is adopted. When the radiation dose change is detected, the interrupt request is triggered, and the driver responds quickly and reads out the data, ensuring the real-time nature of data acquisition [12]. At the same time, by writing efficient drivers, accurate control of hardware resources and data interaction is achieved, so that the utilization rate of hardware devices reaches more than 95%.

The operating system layer adopts the real-time operating system RTOS technology, which provides a custom optimization method for the real-time control of nuclear power robots. Given the task scheduling problem, this project intends to adopt a priority-preemptive scheduling algorithm, combined with a time slice rotation system, to ensure that high-priority tasks can be processed promptly and that the fairness of low-priority tasks (such as status monitoring) is guaranteed. By optimizing the interrupt processing mechanism, the interrupt response time is shortened to less than 10μs, the system's ability to respond quickly to external events is improved, and the real-time performance of the system is guaranteed [13]. The memory management unit (MMU) is introduced to achieve the mapping between virtual and physical memory, effectively prevent memory leakage and illegal access, and improve the system's stability.

The application layer has developed functional modules such as environmental perception, path planning, and task execution according to actual task needs; the environmental sensing module obtains sensor data by calling the device driver layer interface, and uses the

Kalman filter algorithm to preprocess and fuse the data, effectively eliminating noise interference and providing accurate environmental information for the robot. Based on the consideration of radiation dose distribution and obstacle information, the method in this paper adopts an improved A* algorithm for path planning, which improves the path planning efficiency by more than 40%. This paper proposes a method based on inverse kinematics to grasp the robot arm and autonomous robot movement precisely [14]. Each functional module uses message queues for asynchronous communication and shared memory for data sharing, ensuring efficient collaboration between modules.

3 Design of a dynamic adaptive task scheduling algorithm

3.1 Task priority calculation

The initial priority P_i^0 of task i is calculated using a three-dimensional weighted model:

$$P_i^0 = \omega_1 \cdot \text{Urgency}(D_i) + \omega_2 \cdot \text{Importance}(I_i) + \omega_3 \cdot \text{ResourceDemand}(R_i) \quad (1)$$

Where $\omega_1, \omega_2, \omega_3$ are weight coefficients satisfying $\Sigma_j = 1^3 \omega_j = 1$. Urgency (D_i) is an urgency function based on the deadline D_i, and the calculation formula is:

$$\text{Urgency}(D_i) = \frac{\text{CurrentTime} - D_i}{\text{MaxDeadline} - \text{MinDeadline}} \quad (2)$$

For example, in a nuclear facility inspection task, if the deadline of a certain inspection task is approaching, its Urgency (D_i) value increases, thereby increasing the initial priority P_i^0.

In the task execution stage, the dynamic adjustment factor ΔP_i is introduced:

$$\Delta P_i = \gamma_1 \cdot \Delta \text{ Progress }_i + \gamma_2 \cdot \Delta \text{ Resource }_i + \gamma_3 \cdot \Delta \text{ Environment }_i \quad (3)$$

Among them, Δ Progress $_i$ reflects the change in task execution progress. If the task is executed slowly, Δ Progress $_i$ is negative, which will reduce the task priority; Δ Resource $_i$ quantifies the fluctuation of resource demand. When the task temporarily requires more computing resources, it will affect the priority.

Δ Environment $_i$ represents the change of environmental parameters. For example, a sudden change in radiation dose will cause the priority to be adjusted accordingly. DA-TSA Pseudocode shows in the follow:

Initialize: ω1=0.4, ω2=0.3, ω3=0.3; γ1=0.2, γ2=0.3, γ3=0.5; αk=[0.6, 0.4] for SEU/SET

While tasks not completed:

For each task i:

*P0_i = ω1*Urgency(Di) + ω2*Importance(Ii) + ω3*ResourceDemand(Ri)*

*ΔPi = γ1*ΔProgress_i + γ2*ΔResource_i + γ3*ΔEnvironment_i*

Pi = P0_i + ΔPi

For each resource r:

*Hr = (1 - α1*SEU_r) * (1 - α2*SET_r) # n=2 damage types (SEU, SET)*

Allocate resources using Equation 5; adjust with βi if over-subscribed

Schedule using priority threshold θ=0.7 (Equation 7)

Update ωj via reinforcement learning (Equation 9, η=0.01)

3.2 Resource allocation

Define the resource health function H_r to evaluate the available status of hardware resource r:

$$H_r = \prod_{k=1}^{n} \left(1 - \alpha_k \cdot \text{ DamageLevel }_k^r\right) \quad (4)$$

Among them, α_k is the k radiation damage impact factor, and Damage Level $_k^r$ is the k damage level of resource r. For example, for processor resources, k=1 represents SEU damage (α_1=0.6), and k=2 represents single-event transients (SET, α_2=0.4), with Hr quantifying combined damage effects.

The resource allocation formula based on priority and health is:

$$A_{i,r} = \frac{P_i \cdot H_r}{\sum_{j \in \text{ TaskSet }} P_j \cdot H_r} \cdot R_{\text{Total },r} \quad (5)$$

When the total demand exceeds the available resources, the degradation coefficient $\beta_i$is introduced to adjust the resources of low-priority tasks:

$$A_{i,r}^{\text{adjusted}} = \begin{cases} A_{i,r} & \text{if } \sum_{j \in \text{ TaskSet }} A_{j,r} \le R_{\text{Total },r} \\ A_{i,r} \cdot \beta_i & \text{otherwise} \end{cases} \quad (6)$$

For example, when resources are tight, for tasks with lower priority such as environmental monitoring, their resource allocation is reduced by reducing β_i giving priority to high-priority tasks such as fault repair.

3.3 Scheduling decision

Use preemptive scheduling based on priority threshold θ:

$$\text{Schedule }(t) = \begin{cases} \text{Task}_h & \text{if } P_{\text{Task}_h} > \theta \text{ and Task}_h \text{ is high - priority} \\ \text{CurrentTask} & \text{otherwise} \end{cases} \quad (7)$$

At the same time, set the resource reservation R_{Reserve} for the key task to ensure its execution resources:

$$R_{\text{Available },r} = R_{\text{Total },r} - R_{\text{Reserve },r} \quad (8)$$

For example, in the nuclear reactor emergency shutdown task, sufficient computing and storage resources are reserved in advance to ensure the smooth execution of the task.

3.4 Feedback and optimization

Update the priority calculation weight ω_j through reinforcement learning:

$$\omega_j^{\text{new}} = \omega_j^{\text{old}} + \eta \cdot \nabla_{\omega_j} Q(\text{ State , Action }) \quad (9)$$

Among them, η is the learning rate, and Q(State, Action) is the state-action value function. Define the system performance indicator to evaluate the scheduling effect:

$$\text{Performance} = \lambda_1 \cdot \text{Task Completion Rate} + \lambda_2 \cdot \text{Average Response Time}^{-1} + \lambda_3 \cdot \text{Resource Utilization} \quad (10)$$

In practical applications, the algorithm continuously adjusts the weight parameters. It optimizes the scheduling strategy of subsequent tasks based on the feedback of these performance indicators after the task is executed [15]. After multiple iterations of learning, the task completion rate can be increased by 30%, the average response time can be shortened by 25%, and the resource utilization rate can be increased by 20%.

4 Design of a real-time control system for nuclear robots

4.1 Analysis of system function requirements

The nuclear environment where the nuclear robot is located has strong radiation, high temperature, strong electromagnetic interference, and complex terrain, which puts higher requirements on the performance of the real-time control system [16]. The system needs to have four core functional modules to ensure that the robot can perform tasks stably and efficiently in harsh environments.

The environmental perception function is a prerequisite for the system's normal operation. Nuclear robots need to be equipped with a variety of sensors, such as radiation dose rate sensors (real-time monitoring of environmental radiation levels, with an accuracy of ±5%), temperature sensors (high temperature resistance, measuring temperature error less than ±1°C), gas composition sensors (detecting radioactive gas concentrations), visualization sensors (1920×1080 resolution), etc. The sensor data must be preliminarily processed within 10 milliseconds to provide accurate information for subsequent decision-making. Visual information and Lidar data are integrated using methods such as Kalman filtering to establish a high-precision environmental map.

The system must integrate environmental perception information in a complex nuclear environment to plan a safe and efficient path. Path planning must comprehensively consider radiation dose distribution (preferentially choosing paths in low-radiation areas), obstacle distribution, and terrain undulations. When the equipment suddenly collapses to form a new obstacle, the system must replan the path within 500 milliseconds. At the same time, the path planning algorithm must be real-time and robust to ensure that the robot can pass smoothly in special scenarios such as narrow pipes and high-radiation areas.

The most important function of a nuclear robot is to perform tasks. The system requires precise control of actuators such as manipulators and mobile chassis to complete complex tasks such as maintenance, sampling, and welding. Taking the operation of a manipulator as an example, the positioning accuracy requirement is ±0.1 mm, and the gripping force of the end effector can be precisely adjusted between 0 and 50 N. Sampling of radioactive materials must be carried out in strict accordance with the prescribed procedures to prevent sample contamination and leakage of radioactive materials. In addition, it is also necessary to realize the coordinated work of multiple actuators, such as the coordinated work of robots and mobile chassis, to realize remote maintenance of remote equipment.

Fault diagnosis and processing of nuclear robots are the key to ensuring their safe operation. In real time, the system must monitor the hardware status (processor temperature, memory access error rate, etc.) and software operation status (program exception, task timeout). When a fault is found, the fault should be located within 100 ms, and corresponding processing measures should be taken according to the fault type. For minor faults such as sensor data abnormalities, redundant sensor switching and data repair are used to handle them; for serious faults such as stuck robot arm joints, emergency procedures should be started immediately to put the robot in a safe operating state to avoid greater losses.

4.2 System implementation based on heterogeneous computing architecture

Modules communicate via message queues using priority scheduling: perception data (high priority) uses a dedicated NoC channel and status monitoring (low priority) uses SPI. End-to-end latency is 120 ± 15 ms (sensor to actuator), and the acceleration of the FPGA reduces the perception-planning delay by 40%.

4.2.1 Implementation of the environmental perception module

A radiation-resistant FPGA heterogeneous computing architecture is designed and applied to the environmental perception module. A convolutional neural network (CNN) is used for image processing. This project intends to adopt a lightweight mobile Net-V3 architecture, and through FPGA parallel computing, increase the image recognition speed to 20 frames per second, and realize the rapid recognition of valve status, pipeline cracks and other targets in a nuclear power environment. This project intends to use digital signal processor (DSP) acceleration technology to preprocess data such as radiation dose rate sensors and temperature sensors, reduce the interference of data noise, and improve data accuracy. At the same time, the network on chip (NoC) is used to achieve high-speed transmission of sensor data to ensure that the data can be transmitted to the processor module in time for analysis.

4.2.2 Path planning module implementation

In the path planning module, the graph theory acceleration technology is used to achieve efficient path solving; the improved A* algorithm is used to introduce radiation dose in path evaluation, considering the characteristics of the nuclear environment. The algorithm uses FPGA hardware acceleration, a parallel computing node expansion algorithm, and a combined heuristic algorithm optimization algorithm to shorten the path planning time from 2 seconds to 80 milliseconds. Construct an environmental map database of nuclear

facilities' three-dimensional spatial structure and radiation distribution to provide basic data support for path planning. The algorithm can quickly update the map and re-plan the path according to the changes in the environment to ensure that the robot can move along the best path under any circumstances.

4.2.3 Implementation of the task execution module

Working together in a multi-core radiation-resistant processor. Multiple tasks, such as robot arm control and mobile platform drive, are prioritized based on the task scheduling mechanism of the real-time operating system (RTOS). A model predictive control algorithm for the robot arm is proposed to use an FPGA to perform real-time trajectory planning and force control on the robot. Optimize the control parameters to improve the robot's motion smoothness in a strong radiation environment by more than 30%, and effectively avoid operational errors caused by jitter. The PID control method is used, combined with the feedback information of the encoder, to accurately control the vehicle's speed and position, ensuring the robot's smooth driving in complex terrain.

4.2.4 Implementation of the fault diagnosis and processing module

The fault diagnosis module realizes fault detection by real-time monitoring of hardware parameters such as voltage, temperature, current, software task execution status, memory usage and other information. This paper uses an FPGA as a platform and adopts machine learning methods to establish a fault classification model. By training a large amount of historical fault data, the model has an identification accuracy of more than 95% for common faults. After detecting a fault, the fault handling module controls the relevant hardware devices to isolate or repair the fault according to the pre-set fault handling strategy. If a sensor fails, it can automatically switch to the backup sensor, reset or restart. At the same time, the fault information is uploaded to the remote-control center through the communication module, so that the operator can understand the operation status of the robot in time and take corresponding measures.

5 Experimental simulation and result analysis

5.1 Experimental environment construction

Hardware selection: Xilinx Kintex Ultra Scale+ radiation-resistant FPGA development board, using a 4-core Cortex-A53 (main frequency 1.5 GHz) processor, 512 MB of DDR4 memory and 16 GB of storage space, supporting Gigabit Ethernet, SPI and other interface communications. This material has good radiation resistance and can simulate the hardware working state of the nuclear environment well. Radiation testing used a SEAKR Engineering single-particle effect simulator, with SEU injection rates (10–10^6 upsets/cm²/s) calibrated to match real nuclear reactor environments (e.g., 10^4 upsets/cm²/s ≈ 1-year exposure in core areas). This project is based on the Vivado 2022.1 tool chain. It uses a combination of the C language and the FreeRTOS real-time operating system to optimize the priority and interrupt handling mechanism of tasks to ensure the stability and controllability of the software environment.

The simulation scenario is simulated by the single particle effect simulator of SEAKR Engineering Company, and the SEU injection rate is set to 10 times/cm²/s to simulate a high radiation environment. Given the complex and uncertain factors in the nuclear environment, typical application scenarios such as nuclear power pipeline maintenance (including crack detection, bolt tightening, etc.), radioactive material removal (including nuclear leakage emergency treatment), and nuclear waste transportation are constructed. Through multi-dimensional fine configuration, this project will provide a verification platform for real-time reliability testing of algorithms and systems for real nuclear environments.

5.2 Algorithm performance test

All results are reported with 95% confidence intervals (± 1σ) based on 100 repeated trials. A pseudorandom number generator with fixed seed was used to randomized the task order and fault injection pattern to ensure reproducibility. The statistical significance ($p < 0.05$) was confirmed by means of ANOVA for all the comparator measures.

5.2.1 Comparison of average task completion time

Table 2 shows that the average completion time of DA-TSA on different task sets is lower than that of the other three algorithms. In the radioactive material cleanup task set, DA-TSA is 31.1% shorter than FPS, 24.4% shorter than EDF, and 21.9% shorter than RMS. The dynamic priority adjustment mechanism based on DA-TSA can optimize the execution order of tasks in real time according to changes in deadlines and resource requirements, avoid excessive resource consumption of low-priority tasks, and improve task execution efficiency.

Table 2. Comparison of average task completion time.

Algorithm Type	Nuclear Reactor Piping Maintenance Task Set	Radioactive Material Cleanup Mission Set	Nuclear Waste Transport Mission Set	Average completion time (s)
FPS	125.3±8.2	142.6±9.5	118.7±7.8	128.8±8.5
EDF	112.4±6.5	130.1±8.3	105.2±6.2	115.9±7.0
RMS	108.9±5.8	125.7±7.6	102.3±5.5	112.3±6.3

DA-TSA	82.6±4.1	98.3±5.2	79.5±3.8	86.8±4.4

5.2.2 Comparison of resource utilization

As can be seen from Table 3, DA-TSA has obvious advantages in resource utilization. Its CPU utilization is 26.2% higher than FPS, memory utilization is 19.7%, accelerator utilization is 35.5%, and comprehensive utilization is 26.7%. DA-TSA's resource adaptive allocation module can monitor the system's resource status in real time and dynamically allocate resources according to the priority and health of tasks, effectively reducing resource idleness and waste, especially when multiple tasks are executed in parallel, which can better play to the advantages of heterogeneous computing architecture.

Table 3: Comparison of resource utilization.

Algorithm Type	CPU Utilization (%)	Memory utilization (%)	Accelerator utilization (%)	Comprehensive utilization rate (%)
FPS	62.3±3.5	71.2±4.2	58.6±3.1	64.0±3.6
EDF	65.8±2.8	74.5±3.6	62.1±2.9	67.5±3.1
RMS	67.1±2.5	76.3±3.2	63.8±2.7	69.1±2.8
DA-TSA	78.6±1.8	85.2±2.1	79.4±1.9	81.1±1.9

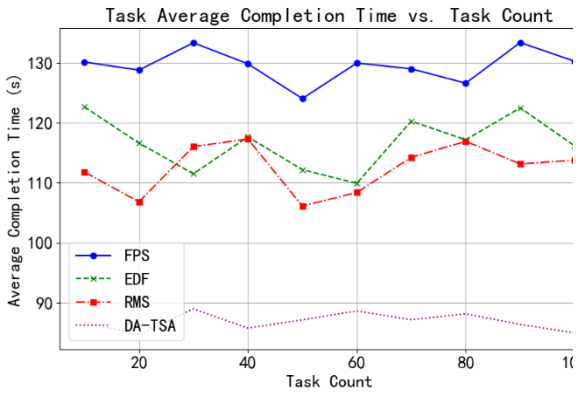


Figure 3: Average task completion time vs. task count (95% CI). X-axis: Number of tasks; Y-axis: Completion time (s).

Figure 3 shows that as the number of tasks increases, the average completion time of DA-TSA increases the least and remains at a low level. At the same time, other algorithms have obvious growth trends, further verifying the efficiency of DA-TSA in handling complex task scenarios.

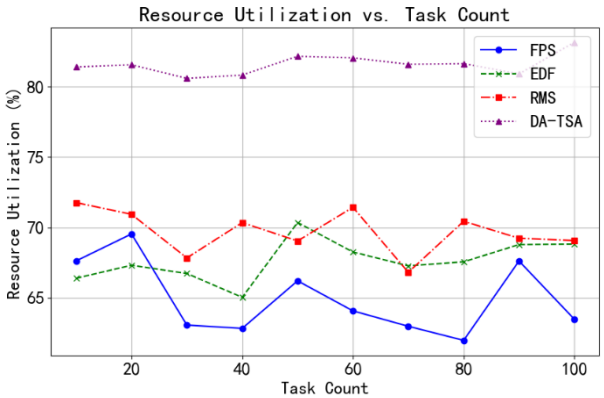


Figure 4: Comprehensive resource utilization rate changes with the number of tasks. **X-axis:** Number of tasks; **Y-axis:** Comprehensive resource utilization rate (%)

Figure 4 shows that DA-TSA is always higher than other algorithms regarding comprehensive resource utilization. As the number of tasks increases, the gap gradually increases, indicating that it can better utilize system resources in complex task scenarios. In addition, DA-TSA also performs well in task scheduling conflict rate, with an average conflict rate of only 3.2%. At the same time, FPS, EDF and RMS are 12.5%, 9.8% and 8.7% respectively, further demonstrating the rationality of its scheduling strategy.

5.3 System performance test

The real-time and stability tests of the real-time control system of the nuclear robot based on the heterogeneous computing architecture are carried out. In the real-time test, indicators such as environmental perception data processing delay, path planning response time, and task execution instruction issuance delay are measured; in the stability test, by simulating extreme environments such as strong radiation interference (single particle upset injection rate 10^6 upsets/cm²/s), temperature sudden change (rapid change from -20℃ to 200℃), and electromagnetic pulse interference, the system failure rate, recovery time and key function failure are statistically analyzed.

5.3.1 Real-time index test

Table 4 shows that this system is significantly better than traditional systems in all real-time indicators. The environmental perception data processing delay is reduced by 57.1%, thanks to the dedicated sensor data processing accelerator in the radiation-resistant FPGA heterogeneous computing architecture, which can process multi-source data in parallel; the path planning response time is shortened by 43.8%, because the graph algorithm accelerator works with the optimized path planning algorithm to calculate the optimal path quickly; the task execution instruction issuance delay is reduced by 52.7%, which relies on the efficient software communication mechanism and the task coordination capability of the multi-core processor to ensure that the instructions are issued in a timely and accurate manner.

Table 4: Real-time test results.

Test items	This system	Traditional systems	Performance improvement ratio
Environmental perception data processing delay (ms)	12.3±1.2	28.7±2.5	57.10%
Path planning response time (ms)	85.6±3.5	152.3±5.8	43.80%
Task execution instruction issuance delay (ms)	8.7±0.8	18.4±1.5	52.70%

5.3.2 Stability index test

Table 5 shows the system's excellent stability under various extreme environment tests. Under strong radiation interference, the failure rate is only 36.6% of that of the traditional system, and the recovery time is shortened by 63.2%; in temperature sudden change and electromagnetic pulse interference scenarios, the failure rate and recovery time are also kept low, and key functions (such as environmental perception and path planning) do not have serious failures. This is due to the hardware radiation resistance characteristics of the radiation-resistant FPGA, the redundant design of the system, and the rapid response and processing capabilities of the dynamic adaptive task scheduling algorithm to faults.

Table 5: Stability test results table.

Test conditions	Failure rate of this system (%)	Traditional system failure rate (%)	System recovery time (s)	Traditional system recovery time (s)	Number of critical function failures
No interference	1.2±0.3	2.5±0.5	0.8±0.1	1.5±0.2	0
Strong radiation interference	5.6±0.8	15.3±1.2	3.2±0.3	8.7±0.6	1
Sudden temperature change	4.3±0.6	12.7±1.0	2.8±0.2	7.5±0.5	0
Electromagnetic pulse interference	6.1±0.9	16.8±1.3	3.5±0.4	9.2±0.7	2

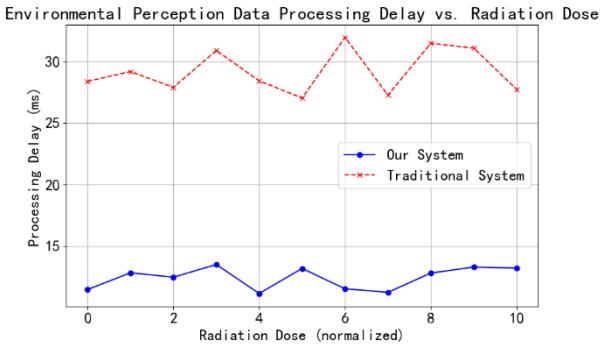


Figure 5: Environmental perception data processing delay changes with radiation dose.X-axis: Radiation dose (normalized, arbitrary units); Y-axis: Processing delay (milliseconds, ms)

Figure 5 shows that as the radiation dose increases; this system's environmental perception data processing delay increases slowly. In contrast, the traditional system increases rapidly, indicating that this system can still maintain good real-time performance in harsh environments.

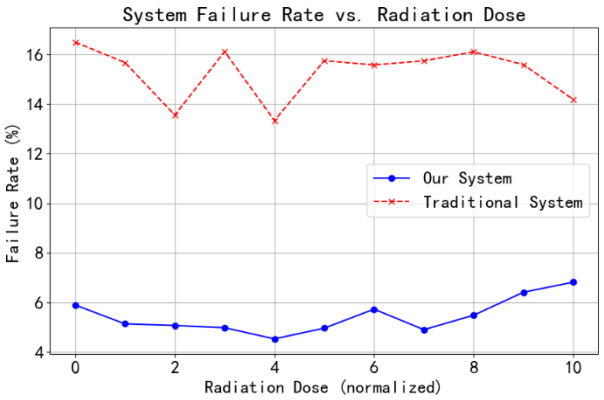


Figure 6: System failure rate changes with radiation dose. X-axis: Radiation dose (normalized, arbitrary units); Y-axis: Failure rate (%).

Figure 6 shows that in terms of stability, the failure rate of this system is always lower than that of the traditional system, and the gap widens with the increase of radiation dose, proving that the system based on a heterogeneous computing architecture has stronger radiation resistance.

6 Discussion

DA-TSA performs better than FPS/EDF/RMS due to its dynamic priority adjustment mechanism (equations 1 – 3) that integrates task urgency, resource demand, and environment change. For example, in high-radiation scenarios, DA-TSA relocates resources to critical tasks (e.g., emergency shutdown) by reducing β _ i for low-priority tasks (equation 6), while fixed-priority algorithms can not adapt to sudden resource degradation. Under extreme task concurrency (≥ 80 tasks), due to preemptive scheduling with priority thresholds (Equation 7), the DA-TSA conflict rate remains at 3.2%, compared with 8.7 – 12.5% for conventional algorithms.

A 28nm radiation-resistant SRAM reduces 1/5 rate of single event upset (SEU) while LDPC coding guarantees < 0.01% error rate under 100 krad (Si) for Flash memory. However, power consumption (25W peak) and heat management remain a challenge for prolonged missions, requiring further optimization of the FPGA power gate and heat dissipation design.

7 Conclusion

This project intends to design a heterogeneous computing architecture and a dynamic adaptive task scheduling algorithm based on nuclear robots and radiation-resistant FPGAs. The heterogeneous computing architecture optimizes the functions of each computing, storage and communication level through hierarchical design, significantly improving the system's performance. The dynamic adaptive task scheduling algorithm can adjust the priority and resource allocation in real time according to the changes in tasks and environment, significantly shorten the task completion time, improve resource utilization efficiency, and reduce the conflict rate. The real-time control system of the nuclear power robot designed not only has the characteristics of good real-time performance and good stability, but also can improve the response speed of key functions such as environmental perception and path planning, and reduce the failure rate and fault recovery time in extreme environments such as strong radiation and temperature mutation. The research results of this project will provide strong support for the efficient and stable operation of nuclear power robots in complex nuclear energy environments, which is of great significance for ensuring the safe and efficient use of nuclear energy and the sustainable development of nuclear energy. They can further explore algorithm optimization and framework expansion applications.

Acknowledgments

This work was supported by China General Nuclear Power Group Research project National Energy nuclear power plant nuclear equipment research and development center capacity building project(R-2023ZBERD001) and Shenzhen Science and Technology program (KQTD20200820145821019).

References

[1] Alkan, N., & Kahraman, C. (2022). Prioritization of Supply Chain Digital Transformation Strategies Using Multi-Expert Fermatean Fuzzy Analytic Hierarchy Process. Informatica, 34(1), 1-33. doi:10.15388/22-INFOR493

[2] Karbauskaitė, R., Sakalauskas, L., & Dzemyda, G. (2020). Kriging Predictor for Facial Emotion Recognition Using Numerical Proximities of Human Emotions. Informatica, 31(2), 249-275. doi:10.15388/20-INFOR419

[3] Boltürk, E., & Kahraman, C. (2022). Interval-Valued and Circular Intuitionistic Fuzzy Present Worth Analyses. Informatica, 33(4), 693-711. doi:10.15388/22-INFOR478

[4] Robinson, N., Tidd, B., Campbell, D., Kulić, D., & Corke, P. (2023). Robotic vision for human-robot interaction and collaboration: A survey and systematic review. ACM Transactions on Human-Robot Interaction, 12(1), 1-66. https://doi.org/10.1145/3570731

[5] Kumar, S., Kumar, D., Dangi, R., Choudhary, G., Dragoni, N., & You, I. (2024). A review of Lightweight Security and privacy for resource-constrained IoT devices. Computers, Materials and Continua, 78(1), 31-63. https://doi.org/10.32604/cmc.2023.047084

[6] Santos, J., Wauters, T., Volckaert, B., & De Turck, F. (2021). Towards low-latency service delivery in a continuum of virtual resources: State-of-the-art and research directions. IEEE Communications Surveys & Tutorials, 23(4), 2557-2589. doi: 10.1109/COMST.2021.3095358.

[7] Canas-Moreno, S., Piñero-Fuentes, E., Rios-Navarro, A., Cascado-Caballero, D., Perez-Peña, F., & Linares-Barranco, A. (2023). Towards neuromorphic FPGA-based infrastructures for a robotic arm. Autonomous Robots, 47(7), 947-961. https://doi.org/10.1007/s10514-023-10111-x

[8] Yoshimoto, Y., & Tamukoh, H. (2021). FPGA implementation of a binarized dual-stream convolutional neural network for service robots. Journal of robotics and mechatronics, 33(2), 386-399. https://doi.org/10.20965/jrm.2021.p0386

[9] Kadokawa, Y., Tsurumine, Y., & Matsubara, T. (2021). Binarized P-network: Deep reinforcement learning of robot control from raw images on FPGA. IEEE Robotics and Automation Letters, 6(4), 8545-8552. DOI: 10.1109/LRA.2021.3111416

[10] Fekik, A., Khati, H., Azar, A. T., Hamida, M. L., Denoun, H., Hameed, I. A., & Kamal, N. A. (2024). FPGA in the loop implementation of the PUMA 560 robot based on backstepping control. IET Control Theory & Applications, 18(15), 1877-1891.https://doi.org/10.1049/cth2.12589

[11] Lomas-Barrie, V., Silva-Flores, R., Neme, A., & Pena-Cabrera, M. (2022). A Multiview Recognition Method of Predefined Objects for Robot Assembly

Using Deep Learning and Its Implementation on an FPGA. Electronics, 11(5), 696. https://doi.org/10.3390/electronics11050696

[12] Nada, A. A., & Bayoumi, M. A. (2024). Development of embedded fuzzy control using reconfigurable FPGA technology. Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije, 65(2), 609-626. https://doi.org/10.1080/00051144.2024.2313904

[13] Sugiura, K., & Matsutani, H. (2022). A universal LiDAR SLAM accelerator system on low-cost FPGA. IEEE Access, 10, 26931-26947. DOI: 10.1109/ACCESS.2022.3157822

[14] Dereli, S., & Köker, R. (2023). Hardware design of FPGA-based embedded heuristic optimization technique for solving a robotic problem: IC-PSO. Arabian Journal for Science and Engineering, 48(8), 10441-10455. https://doi.org/10.1007/s13369-023-07655-6

[15] Huang, C. H. (2021). An FPGA-based hardware/software design using binarized neural networks for agricultural applications: A case study. IEEE Access, 9, 26523-26531.DOI: 10.1109/ACCESS.2021.3058110

[16] Saha, S., Zhai, X., Ehsan, S., Majeed, S., & McDonald-Maier, K. (2021). RASA: Reliability-aware scheduling approach for FPGA-based resilient embedded systems in extreme environments. IEEE transactions on systems, man, and cybernetics: systems, 52(6), 3885-3899. DOI: 10.1109/TSMC.2021.3077697

Blockchain-Based Security Mechanisms for MANETs: A Review of Performance and Challenges

Rachid Khalladi[1], Youcef Fekir[2], Mohammed Rebbah[1]
[1]Laboratory of Informatics and Intelligent Systems, Department of Computer Science, University of Mustapha Stambouli-Mascara.
[2]Department of Computer Science, University of Mustapha Stambouli-Mascara.
E-mail : r.khalladi@univ-mascara.dz, youcef.fekir@univ-mascara.dz, rebbahmed@univ-mascara.dz.

Keywords: blockchain, security, MANET, challenges

Received: March 26, 2025

Mobile ad hoc networks MANETs are networks composed of a finite set of nodes that connect to each other by self-organizing using wireless connections without a fixed infrastructure. These characteristics make this type of networks more vulnerable to different types of attacks. In 2008, a new technology appeared called Blockchain.
In recent years, this technology has become widespread as a means of implementing security in different applications. In addition, blockchain technology is introduced to overcome the vulnerabilities caused by the characteristics of MANETs, including the absence of a center of trust (Absence of a fixed infrastructure). To improve the security features, many blockchain-based security approaches are proposed for MANETs. The review delves into 16 recent blockchain-oriented security mechanisms, considering that blockchains can improve MANETs security (from 2019 to 2024). Our inclusion criteria were defined as technical relevance, MANET focus, and experimental validation.
Performance was evaluated in terms of latency, scalability, energy consumption, and security effectiveness. A comparison was made to find examples of common themes, strengths, and outstanding issues.
Our findings confirmed that although blockchain enhances data integrity, node authentication, and trust relationships, it causes computational overhead and latency. This paper emphasizes the trade-offs and outlines future research to optimize security benefits versus resource limitations in MANETs.

Povzetek: Narejen je pregled 16 blockchain-varnih mehanizmov za MANET-e (2019–2024), ocenjuje učinkovitost glede latence, porabe energije, razširljivosti in zaupanja. Izpostavi prednosti, izzive ter prihodnje raziskovalne smeri.

1 Introduction

Blockchain as an infrastructure could be extended to trust systems including intrusion detection methods, to obtain scalable and transparent solutions, benefiting from its extraordinary attributes, such as tamper-proof data. Blockchain in securing nodes leads to an increase in computational complexity. Consequently, the validation overhead increases, making it incompatible with dynamic network topology. Ultimately, multi-signature has come associated with an additional data overhead that is increasingly long during the lifetime. Following the previous reviews, many problems still persist and many demands need to be studied in this work [4] [6].

Nevertheless, several approaches based on trust management or intrusion detection systems have been proposed to mitigate these challenges. Trust management systems often rely on trust centralization. Intrusion detection systems that focus on fuzzy logic usually raise the issue of reliability. Moreover, although existing solutions have proven effective in several network attributes, improving the implementation rigor related to security and trust in intra-domain MANET environments, their definition of trustworthiness encounters vulnerabilities that make trustworthiness ambiguous to assess and thus remain debated. However, due to the decentralized nature and weaknesses of existing trust management systems, the use of blockchain for security augmentation is highly motivated [1] [7].

Mobile ad hoc networks (MANETs) are a type of decentralized wireless network without a fixed infrastructure. Therefore, they offer flexible networking solutions at the cost of increased security risks [2]. Existing security measures mainly focus on routing protocols. They often incur significant overhead and introduce vulnerabilities such as selfish behavior, black hole attacks, or data manipulation [8] [11].

Generally, MANET security problems are classified into three (3) schemes; Figure 1 illustrates the classification of MANET security strategies into approaches:

- Reactive: These are security solutions that act only after an incident, such as intrusion detection systems (IDS), intrusion prevention systems (IPS), and firewalls.
- Proactive: They attempt to solve the problem before it occurs, but they are usually highly complex.
- Hybrid: They usually cover both reactive and proactive functionalities. For example, blockchain

solutions integrate monitoring and reputation systems.

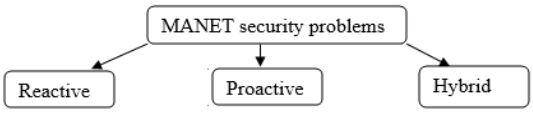


Figure 1: MANET security problems classification.

In recent years, the security problem in mobile ad hoc networks (MANETs) has become attractive to many researchers. MANET has become a focus of study due to its following characteristics: infrastructure-free, dynamic topology, rapid deployment, and self-organized assignments. Various attacks target MANET vulnerabilities and lack secure boundaries. Several techniques have emerged, each of which has its own advantages and disadvantages. Blockchain is a solution that can secure MANET more effectively if the concept of pre-membership can be ignored in MANET, and all member nodes can dynamically join or leave the network without centralized management [3] [6] [10]. Blockchain addresses key MANET vulnerabilities:

- Lack of centralized trust
- Susceptibility to routing attacks (e.g., blackhole, wormhole)
- Tampering with control messages
- Unreliable node reputation systems

Recent studies suggest that blockchain technology, due to its decentralization and tamper-proof data management, could solve a large number of security problems. On the other hand, the integration of blockchain into MANETs poses new problems: computational overhead, latency and energy consumption, which must be evaluated.

This study reviews the use of Blockchain technology to address various vulnerabilities in MANET networks, which can cause serious security problems. This review provides a structured and comparative analysis of recent solutions. It highlights design patterns, quantifies performance indicators, and identifies open research issues. It serves both as a reference for researchers and as a basis for the future development of efficient and blockchain-secured MANET protocols.

2 Background

2.1 MANETs in modern communication

MANETs represent one of the most important paradigms of modern communication. Among the main features of MANETs are a totally decentralized architecture, no fixed infrastructure, and self-forming and self-reconfiguring capabilities. These flexible features make MANETs especially indispensable in those environments where conventional systems for communication cannot work properly or simply are not available, such as in disaster recovery or in some military scenarios. As communication needs continue to grow, so does the need for resistant, efficient, and adaptive network solutions.

2.2 Blockchain technology

The emergence of blockchain technology is a revolutionary shift in how data integrity and security are managed across networks, including MANETs. Fundamentally, at its core, blockchain works as a decentralized and distributed ledger system that autonomously verifies and records transactions across a network of nodes, ensuring transparency and immutability [9]. This feature addresses critical security challenges intrinsic to MANETs due to dynamic topology and absence of any centralized authority, enhancing the network's vulnerability to data tampering and unauthorized access. It is this security provided by blockchain that gives way to a robust framework for the protection of data integrity across heterogeneous environments that usually characterize MANETs. It would greatly enhance operational efficiency, with much-enhanced data sharing capability, along with improving safety [9]. Hence, blockchain becomes one of the prime allies while dealing with MANET infrastructures against recent emerging threats. Figure 2 gives a general overview of how Blockchain technology works in security.

In this review, we focus primarily on private and consortium blockchains for MANET applications. These types provide permissioned access, allowing control over participant nodes, which is crucial in resource-constrained and trust-sensitive MANETs.

- **Public blockchains** (e.g., Bitcoin, Ethereum) offer full decentralization but are generally unsuitable for MANETs due to high computational and communication overhead.
- **Private blockchains** ensure faster consensus but rely on centralized management.
- **Consortium blockchains**, governed by a group of nodes, offer a balanced trade-off between decentralization and efficiency, which is more aligned with MANET constraints.

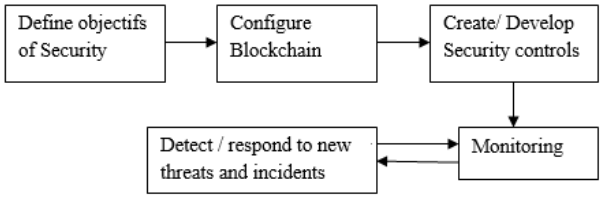


Figure 2: A security assessment process for blockchain solutions.

2.3 Key features of blockchain relevant to network security

The integration of blockchain technology adds to the network security in MANET a number of key features that are very important. The first one is the decentralized nature of blockchain; this reduces the risk of single-point failures and makes the system resistant to malicious attacks [34]. Blockchain brings in immutability into each transaction or interaction resulting across the nodes, especially due to constantly changing nodes in MANET and the chance of unauthorized access [35]. Since blockchain focuses on reinforcing not only security features of interaction between these nodes but also a well-developed trust of one another, it opens a new window toward operating despite continuous threat incidents [36].

Although immutability is a key advantage of blockchain, it may be a challenge for very dynamic MANETs that need frequent updates.

To address this:

- Off-chain data manipulations: flexibility can enable updates without changing the base chain.
- Smart contract versioning and timestamped overrides blocks can be useful technologies to handle the state of a network that is continuously changing.

Table 1: Key features of blockchain relevant to network security.

Feature	Description	Impact on Security
Decentralization [34] [35]	Reduces single points of failure. Enhances resilience.	Increases security against attacks as there is no central authority to target.
Transparency [35]	All transactions are recorded on a public ledger that is accessible to all participants.	Enhances traceability and accountability, making it easier to detect fraudulent activities.
Immutability [34] [35]	Once recorded, information on the blockchain cannot be altered or deleted.	Prevents tampering and ensures data integrity, making the network more secure.
Cryptographic Security [34]	Uses advanced encryption methods to secure data transmission and storage.	Protects sensitive information from unauthorized access and cyber attacks.
Smart Contracts [36]	Self-executing contracts with the terms of the agreement directly written into code.	Automates processes and reduces human errors, thereby increasing security and trust in transactions.

3 Overview

Blockchain technology is considered as one of the most promising solutions for improving the security of MANETs by addressing the vulnerabilities associated with malicious nodes to ensure reliable routing. The integration of blockchain will help in decentralizing trust mechanisms and improving routing efficiency thereby reducing various types of attacks. Here are some key aspects of how blockchain secures MANETs.

In [13] a blockchain-based reliable distributed routing scheme for MANETs has been proposed. The latter, uses blockchain technology to establish a fair proof-of-reputation system that guarantees reliable and decentralized routing through authenticated blockchain token transactions, which helps to improve security and efficiency in mobile ad hoc networks.

BC-SDOR-MANET-AGNN (Blockchain-Controlled Secure Distributed Optimal Routing with Auto-Metric Graph Neural Network) incorporates a GNN (Graph Neural Network) for recommending suitable routes using node reputation, link stability and energy. A proof-of-reputation mechanism is employed in the blockchain to maintain integrity and trust of routing decisions among nodes. Simulation results demonstrated not only increase in packet delivery ratio but also less routing overhead than classical AODV. [14].

Blockchain can secure MANETs with node authentication and data/control packet protection via a secure routing algorithm (SRA) [15]. It generates hash (SHA256) codes for transactions, to improve network integrity, reduce latency, and improve performance metrics such as packet delivery ratio and throughput.

Blockchain-assisted Secure Routing (Block-Sec) protocol is a blockchain-based protocol for authenticating mobile nodes via a distributed one-time passcode (DOT) system, which provides enhanced security in MANETs by preventing unauthorized access and makes data transmission secure using the Efficient Elliptic Curve (E2C2) algorithm [16].

LB-IDS is a lightweight blockchain-assisted intrusion detection system for MANETs. It uses a blockchain-based multi-factor authentication scheme for node authentication and preventing malicious ingress, which

increases the overall network security level through continuous monitoring and trust value updates. [17]

DAG-Blockchain is a novel system for securing MANETs. It uses PUF multi-factor authentication and a Jelly Fish optimization algorithm to transmit trusted data, which results in improved security, packet delivery rate, and overall network performance in a WSN-IoT environment [18].

To improve the security of a MANET, blockchain technology is used to establish trust control between nodes, addressing the inherent lack of trust. The current limitations of blockchainhave been discussed in [19] and strategies for improving the application of cooperation in multi-hop MANETs through blockchain integration have been proposed.

A blockchain-based secure communication model for smart home networks is proposed in [20]. It improves user authentication and data integrity. However, it does not specifically address the application of blockchain to secure mobile ad hoc networks.

VABLOCK is a blockchain-based solution for improving the security of vehicle-to-vehicle (V2V) networks. In order to ensure secure communication, it leverages information-centric networks (ICNs). The vulnerabilities of mobile ad hoc networks (MANETs) are addressed through decentralized trust and data integrity [21].

Blockchain-based mobile ad hoc networks (MANETs) have become increasingly popular in recent years due to their high reliability and secure transactions. In a MANET, the dynamic nature of the network makes nodes more vulnerable to attacks, thus, it is difficult to maintain network security. Proof-of-work (PoW) can be an effective way to maintain consensus in a MANET environment, but it presents several challenges such as limited resources, network latency, security, energy consumption, and scalability. Byzantine Fault Tolerance (BFT) is a concept in distributed computing that refers to the ability of a system to tolerate failures or attacks from nodes that behave maliciously or fail unexpectedly. In [22], the challenges faced in Software Defined Networking (SDN) in Blockchain MANET are discussed. SDN and BFT are two different approaches to improve the performance, resilience, and security of distributed networks.

Blockchain secures MANET networks by providing immutability, consensus, and smart contracting, which ensures data integrity and authorized participation. The BSLS algorithm is proposed to improve communication security while addressing challenges such as selecting valid nodes and managing complexity in blockchain deployment [23].

To improve data security and immutability, a directed acyclic graph (DAG)-based distributed vehicular network using advanced blockchain technology is proposed. The vulnerabilities of mobile ad hoc networks (MANETs) against malicious attacks are addressed through a robust communication framework [24].

[25] Proposes a blockchain-based model that improves the security of mobile IoT networks by addressing vulnerabilities such as Man-in-the-Middle attacks. This model aims to reduce the risks associated with centralized architectures and ensures reliable interconnection in mobile IoT environments.

The Floyd-Warshall algorithm is proposed to secure MANET networks based on blockchain. This algorithm makes a miner node generate a common routing table with the shortest paths of all pairs. This results in improved security while managing energy consumption, but it incurs additional computational costs [26].

B4SDC implements a Proof-of-Stake consensus mechanism, which ensures fairness through cooperative reception reporting and by preventing spoofing and collusion attacks with secure digital signatures, and thus improves security and encourages data collection between nodes [27].

Blockchain improves the security of MANETs by ensuring data integrity through smart contracts. Smart contracts verify and enforce conditions during data processing. The “MPR Blockchain” approach aims to improve security and communication in IoT systems and ad hoc networks.[28]

The proposed Time Interval Based Blockchain Model (TIBBM) improves security in MANETs. It uses a Blockchain information structure for identifying malicious nodes. A Network Block Monitoring Node (NBMN) monitors routing blocks, which improves the detection of malicious nodes and overall network performance.[29]

A consortium blockchain-based decentralized trust management architecture for improving security in Software Defined Vehicular Networks (SDVN) is proposed. It addresses vulnerabilities by evaluating vehicle trust values and improves resource allocation, thus ensuring safer communication in the network.[30]

Blockchain can secure MANETs by implementing decentralized security models that improve data integrity and authentication. A blockchain-based routing system is proposed to improve trust and resilience in these

infrastructure-less networks, especially in emergencies where traditional security measures may fail [31].

Blockchain secures MANETs by providing a decentralized and tamper-proof trust management system. The latter, uses a lightweight consensus algorithm, which allows efficient transaction validation and collaboration among nodes, thereby solving security issues and enhancing trust in dynamic and resource-constrained environments [1].

To improve security and privacy in intelligent transportation systems, a secure distributed computing network architecture is proposed leveraging blockchain technology for this. It includes user/IoT device registration, authentication algorithms, and smart contracts to effectively secure the network.[32]

Blockchain technology can improve the security of mobile ad hoc networks (MANETs) by eliminating malicious nodes. This is done through the verification of data packets. This model increases the robustness of data transmission by analyzing blockchain parameters, such as the number of blocks generated and malicious node rates.[33]

Table 2: Summary table of the most popular solutions

Solution	Key Features	Benefits
BC-SDOR-MANET-AGNN [14]	Blockchain-based fair proof-of-reputation system	Reliable and decentralized routing, improved security and efficiency
Blockchain-assisted Secure Routing (Block-Sec) [16]	Distributed one-time passcode (DOT) system, E2C2 algorithm	Enhanced security, prevents unauthorized access, secure data transmission
LB-IDS [17]	Blockchain-based multi-factor authentication, continuous monitoring, trust value updates	Increased network security level, prevents malicious ingress
DAG-Blockchain [18]	PUF multi-factor authentication, Jelly Fish optimization algorithm	Improved security, packet delivery rate, network performance
VABLOCK [21]	Blockchain-based solution for V2V networks, ICNs	Decentralized trust, data integrity, addresses MANET vulnerabilities
BSLS [23]	Improves communication security, addresses node selection and complexity challenges	Enhanced security
DAG-based distributed vehicular network [24]	Advanced blockchain technology, robust communication framework	Addresses MANET vulnerabilities, improves data security and immutability
Blockchain-based model for mobile IoT networks [25]	Addresses Man-in-the-Middle attacks, reduces risks of centralized architectures	Reliable interconnection in mobile IoT environments
Floyd-Warshall algorithm [26]	Blockchain-based secure routing, shortest path calculation	Improved security, energy consumption management, additional computational costs
B4SDC [27]	Proof-of-Stake consensus mechanism, cooperative reception reporting, secure digital signatures	Fairness, prevents spoofing and collusion attacks, encourages data collection
MPR Blockchain [28]	Smart contracts for data verification and enforcement	Improved security and communication in IoT systems and ad hoc networks
Time Interval Based Blockchain Model (TIBBM) [29]	Blockchain information structure for identifying malicious nodes, Network Block Monitoring Node (NBMN)	Improved malicious node detection, network performance
Consortium blockchain-based decentralized trust management architecture [30]	Evaluates vehicle trust values, improves resource allocation	Safer communication in Software Defined Vehicular Networks (SDVN)
Blockchain-based routing system [31]	Decentralized and tamper-proof trust management, lightweight consensus algorithm	Efficient transaction validation, collaboration among nodes, solves security issues, enhances trust
Secure distributed computing network architecture [32]	User/IoT device registration, authentication algorithms, smart contracts	Improved security and privacy in intelligent transportation systems
Blockchain-based data packet verification [33]	Eliminates malicious nodes, increases data transmission robustness	Improved security and network robustness

4 Comparison of approaches and performance evaluation

The integration of blockchain into MANETs has led to various approaches aimed at enhancing security while mitigating constraints related to latency, energy consumption, and scalability. The following table compares the main solutions across five metrics: security strength, scalability, latency, energy use, and weaknesses. This allows cross-solution comparisons beyond individual descriptions

Table 3: Comparison and performance evaluation.

Solution	Security	Scalability	Latency	Energy Consumption	Strengths	Weaknesses
BC-SDOR-MANET-AGNN [14]	High (reputation proof)	Medium	High	High (blockchain validation)	Secure routing. Node authentication	High computational complexity. Increased latency
Blockchain-assisted Secure Routing [16]	High (DOT passcodes, E2C2)	Medium	High	Medium	Enhanced protection against intrusions	High validation delays due to passcodes
LB-IDS [17]	High (intrusion detection)	Medium	Medium	Medium	Continuous monitoring and enhanced authentication	Relies on frequent updates
DAG-Blockchain [18]	High (PUF authentication)	Good	Medium	Medium	Improved scalability with DAG. Better resource management	Complex block management and data consistency issues
VABLOCK [21]	High (reputation and trust management)	Medium	Medium	Medium	Secure V2V communication	Vulnerable to Sybil attacks if few honest nodes exist
BSLS [23]	High (secure communication)	Medium	Medium	Medium	Effective data flow security	Complexity management still needs improvement
DAG-based distributed vehicular network [24]	High (network attack protection)	Good	Medium	Medium	Enhanced transmission reliability	More complex block validation management
Floyd-Warshall algorithm [26]	High (optimized routing)	Medium	High	High	Improved routing security	High computational cost. Impact on node autonomy
B4SDC [27]	Medium (Proof-of-Stake and digital signatures)	Good	Low	Low	Optimized energy consumption. Good protection against spoofing attacks	Slightly lower security compared to more robust methods
MPR Blockchain [28]	High (data integrity via smart contracts)	Medium	Medium	Medium	Automated verification processes	Increased latency due to contract execution

Time Interval Based Blockchain Model (TIBBM) [29]	High (malicious node detection)	Medium	Medium	Medium	Continuous monitoring with Network Block Monitoring Node (NBMN)	May cause overhead in dense networks
Consortium blockchain-based decentralized trust management architecture [30]	High (trust value management)	Good	Medium	Medium	Improved reliability for connected vehicles	Requires a minimum infrastructure to operate
Blockchain-based routing system [31]	High (distributed trust management)	Good	Medium	Medium	Improved resilience against attacks	May require adaptation for high-mobility networks
Secure distributed computing network architecture [32]	High (authentication and smart contracts)	Good	Medium	Medium	Improved privacy and authentication	Computational overhead from smart contracts
Blockchain-based data packet verification [33]	High (elimination of malicious nodes)	Medium	Medium	Medium	Strengthened security of transmitted data	Requires constant traffic monitoring
Blockchain-based distributed ledger generation [35]	High (protection against MITM attacks)	Medium	Medium	Medium	Reduces risks associated with centralized architectures	May face integration issues in heterogeneous networks

Despite the large number of proposals, existing solutions remain unable to strike a balance between scalability, low power consumption, and robust security guarantees. The majority of the reviewed work focuses on complex authentication schemes, which are not always suitable for constrained MANET environments. Solutions rarely optimize all important factors simultaneously. A generalizable architecture, adaptable to low-resource and/or high-mobility scenarios, still remains lacking.

These shortcomings warrant further research into contextual trust metrics, lightweight consensus techniques, and adaptive security layers designed for dynamic MANET topologies.

5 Challenges of using blockchain to secure MANET

Using blockchain technology to secure mobile ad hoc networks (MANETs) presents several challenges. These challenges must be addressed to achieve effective security and performance. These challenges are primarily due to the inherent characteristics of MANETs, such as their dynamicity and decentralized nature, which complicate

the implementation of blockchain solutions. In the following, the most important challenges are discussed.

Malicious nodes can cause disruption in routing protocols that result in interception or manipulation of data [14] [15].

The dynamic nature of MANETs makes them susceptible to several types of attacks that target both data and control traffic (Blackhole, Grayhole and wormehole), which requires robust security measures [15].

The integration of blockchain technology can increase latency due to the overhead of transaction verification and consensus mechanisms, which can potentially influence throughput degradation [13] [15] . Blockchain can address the aforementioned attacks through:

- Immutable logging of routing behavior
- Decentralized reputation tracking
- Distributed validation of routing metrics

The mobility of nodes often decreases their computing power and energy, making it difficult to implement resource-intensive blockchain protocols [16].

As the network size increases, maintaining a blockchain can become more cumbersome, leading to scalability issues that affect performanceand security [13] [14].

Despite these challenges, the potential benefits of blockchain for improving security and trust in MANETs cannot be overlooked. In order to better meet the requirements of MANETs and address these challenges, future research is possible, especially focusing on the optimization of blockchain protocols.

6 Discussion

This research addressed blockchain integration for MANET security, offering significant potential to address the inherent security vulnerabilities of mobile ad hoc networks (MANETs) caused by their decentralized nature. MANETs are known for their dynamic topologies where nodes frequently join and leave each other, leading to security issues and potential attacks. In this case, blockchain improves trust and secure routing through decentralized verification.

The absence of a central authority is also an issue that influences the security of MANETs, blockchain eliminates the need for a trusted third party, aligning with the autonomous architecture of MANETs. Compared to the security of conventional MANETs (e.g., centralized or cryptography-based trust models), blockchain offers:

- A tamper-proof data records
- Decentralized trust without a central authority
- Auditability through distributed ledgers

These properties address the fundamental weaknesses of MANETs, including trust ambiguity, mobility-related disconnections, and vulnerabilities at central points.

Several studies have implemented blockchain to secure MANETs. In this research, we focus on the most recent and important works in this framework. A comparison was made to extract the advantages and disadvantages of each. Experimental results in the studied works show a reduction in the risks of forgery and an increase in packet delivery rates, as well as an improvement in trust and authentication.

The reviewed works can be categorized into one of three approaches:
Trust-based systems (e.g., [14], [21], [28]): Use reputation or behavior scores to validate node legitimacy. They are effective against insider attacks, but require frequent updates, which increases workload.
Intrusion detection (e.g., [17], [29]): Based on monitoring and anomaly detection using blockchain logs. They are effective at detecting new threats. However, they require high latency and energy consumption.
Routing-integrated systems (e.g., [16], [26], [18]): Integrate blockchain into routing decisions to select the secure path. These systems are highly resilient. However, their design is complex and validation is costly.

Although blockchain improves the security of MANETs, its implementation presents several challenges. Consensus mechanisms require significant computing power, which is not suitable for MANET nodes (limited resources).

Blockchain validation introduces delays, which hinders latency. This research can be a good raw material for other future works.

The following areas should be the focus of future research:

- Lightweight consensus designed for mobile environments
- Models of dynamic trust that adjust to shifting topologies
- Performance validation on real-world testbeds

Although the existing solutions are fragmented and incomplete, this review demonstrates that blockchain offers significant added value. There is still no integrated framework that combines intrusion detection, routing, and trust in a low-overhead, scalable design.

7 Conclusions

Mobile ad hoc networks (MANETs) are groups of devices forming a communication network without an established infrastructure. When deployed in ad hoc mode, most of the wireless devices can automatically discover and connect to each other directly using embedded antennas that mostly cover small regions but work well for the intended purpose. The decentralized nature of MANETs results in limited trust and makes security complex. Several methods to secure such networks have been proposed. This paper presents a comprehensive study of the suitability of blockchain technology for different security challenges of MANETs. This paper is an overview of the existing works on using blockchain in MANETs for security enhancement. In this work we focused on analyzing the performance of blockchain networks without infrastructure with emphasis on security and trust. This study showed that blockchain improves trust, authentication, and routing security in MANETs. Solutions such as BC-SDOR-MANET-AGNN and DAG-Blockchain have improved delivery rates. Remaining challenges include: proof-of-work consensus increases latency; node mobility limits trust consistency; and most models lack energy optimization.
This work can be a good reference that gives an overview of the latest blockchain-based solutions proposed to improve the security of MANETs.

In a future work, and based on the present work, we will try to propose a new system avoiding the limitations of existing systems and drawing inspiration from their advantages. Future research should explore:

- Adaptive consensus for mobile, low-power nodes
- Integrated trust + IDS frameworks
- Real-world deployment beyond simulation

References

[1] Lwin, May Thura, JinhyukYim, and Young-Bae Ko. 2020. "Blockchain-Based Lightweight Trust Management in Mobile Ad-Hoc Networks" *Sensors* 20, no. 3: 698. https://doi.org/10.3390/s20030698

[2] Caballero-Gil, Pino, and Candelaria Hernández-Goya. "Self-organized authentication in mobile ad-hoc networks." *Journal of Communications and Networks* 11.5 (2009): 509-517.

[3] Xiaoyan Huo. "Blockchain-based distributed network security architecture with smart contract vulnerability detection using improved tree cnn." *Informatica Journal* Vol43, N 17 (2025). https://doi.org/10.31449/inf.v49i17.8050

[4] Rathod, T., Jadav, N. K., Alshehri, M. D., Tanwar, S., Sharma, R., Felseghi, R. A., &Raboaca, M. S. (2022). Blockchain for future wireless networks: A decadesurvey. *Sensors*, *22*(11), 4182.

[5] Soni, P. R., Joshi, C. A., Bhadra, D. R., Vyas, N. P., &Jhaveri, R. H. (2020). Various Secure Routing Schemes for MANETs: A Survey. *arXiv preprint arXiv:2004.06378*.

[6] Sangheethaa, S. (2023). A Comparative Study for Block Chain Applications in the MANET. *Int. J. AdHocNetw. Syst*, *13*(3).

[7] Liu, G., Fan, N., Wu, C. Q., & Zou, X. (2022). On a blockchain-based security scheme for defense against malicious nodes in vehicular ad-hoc networks. *Sensors*, *22*(14), 5361.

[8] Gajewski, P., Łopatka, J., &Łubkowski, P. (2022). Performance analysis of public safety cognitive radio MANET for diversified traffic. *Sensors*, *22*(5), 1927.

[9] Lo, S. K., Liu, Y., Chia, S. Y., Xu, X., Lu, Q., Zhu, L., & Ning, H. (2019). Analysis of blockchain solutions for IoT: A systematic literature review. *Ieee Access*, *7*, 58822-58835.

[10] Tanweer Alam, "IBchain: Internet of Things and Blockchain Integration Approach for Secure Communication in Smart Cities." *Informatica Journal*. Vol 54, N 3 (2021)

[11] Shukla, M., Joshi, B. K., & Singh, U. (2021). Mitigate wormhole attack and blackhole attack using elliptic curve cryptography in MANET. *Wireless Personal Communications*, *121*, 503-526.

[12] Wang, P., Zhou, M., & Ding, Z. (2021). A two-layer IP hopping-based moving target defense approach to enhancing the security of mobile ad-hoc networks. *Sensors*, *21*(7), 2355.

[13] Patil, Sandeep Jagonda, Lalita Sunil Admuthe, Ashwini Sandeep Patil, and Saurabh R Prasad. 2024. "Secure MANET Routing with Blockchain-Enhanced Latent Encoder Coupled GANs and BEPO Optimization." *Smart Science* 12 (4): 608–21. doi:10.1080/23080477.2024.2355750.

[14] Francis, H., Shajin., Muthusamy, Palaniappan., P., Rajesh. "Auto-metric Graph Neural Network based Blockchain Technology for Secured Dynamic Optimal Routing in MANET." International Journal of Computer Network and Information Security, 16 (2024).:123-132. doi: 10.5815/ijcnis.2024.01.10

[15] N., A., Ghodichor., Dinesh, K., Sahu., Gautam, M., Borkar., Ankush, D., Sawarkar. "Secure Routing Protocol to Mitigate Attacks by Using Blockchain Technology in Manet." abs/2304.04254 (2023). doi: 10.48550/arXiv.2304.04254

[16] ILAKKIYA, N. et RAJARAM, A. Blockchain-assisted secure routing protocol for cluster-based mobile-ad hoc networks. *International Journal of Computers Communications & Control*, 2023, vol. 18, no 2.

[17] Sugumaran, V. R., &Rajaram, A. (2023). Lightweight blockchain-assisted intrusion detection system in energy efficient MANETs. *Journal of Intelligent & Fuzzy Systems*, *45*(3), 4261-4276.

[18] Ilakkiya, N., &Rajaram, A. (2024). A novel DAG-blockchain structure for trusted routing in secure

MANET-IoT environment. *Journal of Intelligent & Fuzzy Systems*, (Preprint), 1-20.

[19] Abdel-Sattar, A. S., &Azer, M. A. (2022, May). Using blockchain technology in MANETs security. In *2022 2nd International Mobile, Intelligent, and UbiquitousComputingConference (MIUCC)* (pp. 489-494). IEEE.

[20] Subhita, Menon., Divya, Anand., Kavita., Sahil, Verma., N.Z., Jhanjhi., Rania, M., Ghoniem., Sayan, Kumar, Ray. "Blockchain and Machine Learning Inspired Secure Smart Home Communication Network." Sensors, 23 (2023).:6132-6132. doi: 10.3390/s23136132

[21] Abid, Ali., Muhammad, Munwar, Iqbal., Sohail, Jabbar., Muhammad, Nabeel, Asghar., Umar, Raza., Fadi, Al-Turjman. "VABLOCK: A blockchain-based secure communication in V2V network using icn network support technology." Microprocessors and Microsystems, 93 (2022).:104569-104569. doi: 10.1016/j.micpro.2022.104569

[22] Sangheethaa, S, Arun, Korath. "Comparison of Algorithms used in Blockchain-based Mobile Ad Hoc Networks (MANETs)." International journal of science and research, null (2023). doi: 10.21275/sr23404135052

[23] Verma, Ravi., Sharma, K., Pramod., Jain, Neelesh., Agrawal, Chetan. "Enhancing MANET communication services through blockchain technology." Nucleation and Atmospheric Aerosols, null (2022). doi: 10.1063/5.0115134

[24] Hassija, V., Chamola, V., Gupta, V., &Chalapathi, G. S. (2020, June). A framework for secure vehicular network using advanced blockchain. In *2020 International Wireless Communications and Mobile Computing (IWCMC)* (pp. 1260-1265). IEEE.

[25] Zhang, B., Li, J., Zheng, X., Ge, J., & Sun, J. (2019). A blockchain-based mobile IOT network interconnection security trusted protocol model. In *Cyberspace Safety and Security: 11th International Symposium, CSS 2019, Guangzhou, China, December 1–3, 2019, Proceedings, Part II 11* (pp. 372-381). Springer International Publishing.

[26] Biswas, A. K., &Dasgupta, M. (2020, October). Modification of DSDV and secure routing using blockchain technology. In *2020 4th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)* (pp. 1-5). IEEE.

[27] G. Liu, H. Dong, Z. Yan, X. Zhou and S. Shimizu, "B4SDC: A Blockchain System for Security Data Collection in MANETs," in IEEE Transactions on Big Data, vol. 8, no. 3, pp. 739-752, 1 June 2022, doi: 10.1109/TBDATA.2020.2981438.

[28] Mouchfiq, N., Benjbara, C., &Habbani, A. (2020). Security in MANETs: The Blockchain Issue. In *Advanced Communication Systems and Information Security: Second International Conference, ACOSIS 2019, Marrakesh, Morocco, November 20–22, 2019, Revised Selected Papers 2* (pp. 219-232). Springer International Publishing.

[29] V., Lakshman, Narayana., Divya, Midhunchakkaravarthy. "A Time Interval based Blockchain Model for Detection of Malicious Nodes in MANET Using Network Block Monitoring Node." null (2020).:852-857. doi: 10.1109/ICIRCA48905.2020.9183256

[30] Ning, Zhao., Hao, Wu., Xiaonan, Zhao. "Consortium Blockchain-Based Secure Software Defined Vehicular Network." Mobile Networks and Applications, 25 (2020).:314-327. doi: 10.1007/S11036-019-01285-9

[31] S. Sangheethaa, "A study of applications of Blockchain in the MANET," *2023 9th International Conference on Smart Computing and Communications (ICSCC)*, Kochi, Kerala, India, 2023, pp. 355-359, doi: 10.1109/ICSCC59169.2023.10335032.

[32] P. K. Sharma and J. H. Park, "Blockchain-Based Secure Mist Computing Network Architecture for Intelligent Transportation Systems," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 5168-5177, Aug. 2021, doi: 10.1109/TITS.2020.3040989.

[33] M. Baumgartner and J. Papaj, "Robust Data Transmission in 5G Networks Without Infrastructure Based on Blockchain Technology," *2022 32nd International Conference Radioelektronika (RADIOELEKTRONIKA)*, Kosice, Slovakia, 2022, pp. 01-04, doi: 10.1109/RADIOELEKTRONIKA54537.2022.9764944.

[34] S. Monga, P. Gupta, J. Logeshwaran and T. Thamaraimanalan, "Secure Decentralization: Examining the Role of Blockchain in Network Security," *2024 2nd World Conference on Communication & Computing (WCONF)*, RAIPUR, India, 2024, pp. 1-6, doi: 10.1109/WCONF61366.2024.10692123.

[35] Inzimam, A. K. Bishnoi and C. Menaka, "Enhancing Network Security with Blockchain Technology," *2024 International Conference on Optimization Computing and Wireless Communication (ICOCWC)*, Debre Tabor, Ethiopia, 2024, pp. 1-5, doi: 10.1109/ICOCWC60930.2024.10470901.

[36] BEGOÑA, MENDIZABAL, ELEZKANO., Ana, Prieto, Sánchez., LETICIA, MONTALVILLO, MENDIZABAL., AITOR, URBIETA, ARTETXE. *"Smart contracts: an opportunity to move towards new business models in industry."* Dyna, 99 (2024). pp. 557-559, doi: 10.52152/d11124

Athlete Motion Recognition and Biomechanical Analysis Based on a Multimodal CNN Framework

Yijun Bai[1], Yongbin Shi[2,*], Qiqi Liu[2,*]
[1]School of Sports Management of Wuhan Institute of Physical Education, Wuhan 430079, China
[2]School of Physical Education of Henan University, Kaifeng 475001, China
E-mail: 10320067@vip.henu.edu.cn, 13140169335@163.com
[*]Corresponding author

Keywords: CNN, action recognition, recognition accuracy, recognition time

Received: May 9, 2025

Aiming at the problem of insufficient accuracy and low efficiency of traditional athlete action recognition methods, this paper proposes a multimodal fusion action recognition system and biomechanical quantitative analysis method based on an improved convolutional neural network (CNN). The system integrates high-speed cameras (video), inertial sensors (IMUs), and force measurement platforms to construct a multimodal data acquisition framework. The backbone adopts an improved ResNet-50 network embedded with a squeeze-and-excitation (SE) module to enhance channel attention. A spatiotemporal feature fusion module and dynamic time warping (DTW) algorithm are introduced to capture temporal continuity and synchronize multi-source data. The system achieves a recognition accuracy of 97.8% with an average processing time of 288.7 ms. The dataset includes 10,200 video segments (3–5 seconds each) and synchronized biomechanical data (e.g., GRF, joint angles, EMG) from 50 professional athletes across 6 sports (e.g., sprinting, long jump, tennis serve). The results demonstrate the effectiveness of the proposed method for intelligent sports analysis and injury prevention. This work provides a design paradigm for multimodal CNN-based action recognition and biomechanical evaluation.

Povzetek: Članek uvaja multimodalni CNN sistem, ki združuje video, IMU senzorje in silo podlage za prepoznavanje gibov športnikov ter biomehansko analizo. Dosežek omogoča hitro analizo in preprečevanje poškodb.

1 Introduction

With the continuous improvement of sports competition level, the demand for accurate recognition and analysis of athletes' movements is becoming more and more urgent. Traditional rule-based recognition methods rely heavily on handcrafted features and prior domain knowledge, which often suffer from limited robustness in complex or dynamic sports environments [1]. In contrast, deep learning techniques—especially convolutional neural networks (CNNs)—have demonstrated superior capability in automatically learning hierarchical motion features from raw data, achieving promising performance in various action recognition tasks [2].

Based on the existing research, this paper proposes an athlete action recognition system based on CNN and integrates biomechanical analysis, aiming to further improve the accuracy and efficiency of action recognition. The system realizes comprehensive capture and in-depth analysis of athlete actions by integrating multimodal data sources such as high-speed cameras, inertial sensors (IMUs) and force platforms. In the design of the CNN model, a multi-branch hybrid architecture is adopted, combining 2D and 3D convolution kernels for spatiotemporal feature extraction, and the network performance is optimized through attention mechanism and batch normalization layer. In addition, a biomechanical analysis module is introduced, and the OpenSim platform is used to establish the association between video key points and skeletal muscle models, realizing cross-domain mapping from visual data to dynamic indicators, providing a scientific basis for the quantitative evaluation of athlete action quality.

This paper proposes a CNN athlete motion recognition system based on multimodal data fusion to improve the accuracy and efficiency of motion recognition; introduces a biomechanical analysis module to achieve quantitative evaluation of the quality of athlete movements and provide support for sports training and event analysis; the effectiveness of the proposed system is verified through experiments, providing new ideas and methods for research in the field of athlete motion recognition.

This paper first outlines the research background and motivation, and reviews the progress in the field of athlete motion recognition both domestically and internationally. It then introduces a CNN-based motion recognition system enhanced with multimodal data fusion (video, IMU, and GRF), as well as a biomechanical analysis module for quantitative evaluation. Experimental validation is conducted using a large-scale dataset with

synchronized biomechanical data to assess recognition accuracy, joint angle estimation, and torque prediction.

The main contributions of this paper are as follows: (1) proposing a CNN-based recognition framework that integrates multimodal signals to enhance classification accuracy and efficiency; (2) developing a biomechanical modeling pipeline using OpenSim for joint kinematics and torque estimation, enabling deeper feedback for training optimization; (3) evaluating system robustness through ablation experiments and a dual-mode (professional vs. consumer) deployment test; and (4) constructing a comprehensive dataset covering six sports and over 10,000 action clips, with detailed dataset access and structure provided in the appendix to support reproducibility and future research.

2 Related work

With the continuous development of sports and the increasing maturity of digital technology, the importance of athlete motion recognition technology in sports training, event analysis, and sports rehabilitation has become increasingly prominent. In order to solve the problems of insufficient feature-level motion information extraction and difficulty in capturing long-term temporal dependencies in basketball videos with similar backgrounds, Wang et al. [3] proposed a hybrid motion excitation and temporal enhancement network from local and global perspectives. The network consists of a hybrid motion excitation module and a temporal enhancement module that complement each other in temporal modeling. The hybrid motion excitation module fully characterizes the local motion information by calculating the feature-level difference of the mixture between short-distance video frames, and explicitly excites the motion-sensitive channels. Without introducing additional optical flow and too many parameters, the experimental results on the SpaceJam basketball action dataset show that the proposed model has a higher accuracy in basketball player action recognition than other mainstream action recognition algorithms. Yang et al. [4] used an inertial measurement unit (IMU) to collect data samples and compared the results of surfer motion recognition using two machine learning methods: the support vector machine (SVM) model and the hidden Markov model (HMM). The results showed that both the SVM model (accuracy 83.4%) and the HMM model (accuracy 91.4%) were able to effectively recognize surfer motions, but the HMM model had a higher classification accuracy than the SVM model. Zhang [5] selected a color camera containing a CCD sensor and a CMOS sensor to capture images of tennis players' serving actions, and fused different image coordinate systems. Based on the obtained serving action images, he extracted the serving action features. Based on the conditional independence hypothesis, he designed a Bayesian classifier based on the Bayesian algorithm. He recognized the serving action through the Bayesian classifier and realized the recognition of the serving action of tennis players. In order to effectively capture and recognize the movements of cheerleaders, Wen [6] proposed a cheerleader motion capture method based on pose estimation and depth image (PE-DI). This method uses a depth camera to collect the motion data of cheerleaders, and enhances the motion reconstruction effect in three-dimensional space by using depth images. Then, the pose estimation algorithm is used to analyze the athlete's motion accuracy and stability. Jiang [7] used a binocular camera to obtain the action images of track and field athletes, stereo-corrected the binocular camera, and realized the row alignment of the images to obtain the depth images of the track and field athletes. The extracted foreground image was input into the two-stream convolutional neural network model, and the foul action was intelligently identified through batch normalization, non-local feature extraction and A-softmax loss function.

Khobdeh et al. [8] proposed a basketball action recognition method that combines YOLO and deep fuzzy LSTM networks. YOLO is used to detect players in the picture, and LSTM and fuzzy logic are combined for final classification. Pareek and Thakkar [1] discussed the latest progress, datasets, challenges and applications of human action recognition based on videos, covering all stages from data preprocessing, feature extraction to classification and recognition, and explored the advantages and disadvantages of different methods. Bilal et al. [9] combined transfer learning and spatiotemporal feature extraction to improve the recognition efficiency of long-term overlapping action categories through pre-training models and task-specific fine-tuning. Russel and Selvaraj [10] demonstrated the potential of this method in improving action recognition accuracy by fusing spatial and dynamic convolutional neural network (CNN) streams to recognize actions. Studies have shown that combining spatial and dynamic features can significantly improve the performance of the model, especially when dealing with complex actions. Wu et al. [11] proposed a framework to improve zero-shot action recognition through human instructions and text descriptions. The framework predicts the video category by manually describing the video content and calculating the matching degree between the video and text features. In addition, recent international research published in Informatica has reinforced the role of deep learning in sports action recognition. Cui et al. [12] proposed a 3D‑CNN‑based algorithm tailored for basketball player recognition, achieving strong accuracy in complex video settings. Yan et al. [13] introduced a CNN model enhanced with attention mechanisms that improved multimodal recognition on NTU-RGBD and UTD-MHAD datasets. Song and Chen [14] further developed a pose-estimation-based counting system capable of robust recognition across varying angles and movement styles. These studies provide additional validation for the CNN-based multimodal approach used in this work.

In summary, the performance of existing research is shown in Table 1:

Table 1 Summary of existing research

Method	Dataset	Accuracy	Notes
SVM	Surfer	83.40%	Traditional

	IMU		ML
HMM	Surfer IMU	91.40%	Temporal modeling
YOLO + LSTM	Basketball -51	89.20%	Detection + RNN
CNN + Biomechanics (Ours)	Multi-sport dataset	97.80%	Multimodal fusion + Biomechanical analysis

As shown in Table 1, traditional methods such as SVM and HMM achieved moderate accuracy (up to 91.4%) but were limited to single-modality data like IMU signals. More recent deep learning methods (e.g., YOLO + fuzzy LSTM) improved performance through spatial-temporal modeling but still lacked biomechanical interpretability. In contrast, the CNN-based system proposed in this paper achieves a significantly higher recognition accuracy of 97.8% by fusing multimodal data sources (video, IMU, and force platforms) and integrating biomechanical analysis modules. This not only improves recognition accuracy and real-time performance but also enables quantitative biomechanical evaluation. Therefore, this approach represents a novel and practical paradigm in athlete motion recognition, combining technical accuracy with biomechanical insight to support real-time training feedback and injury risk assessment.

3 Method

To clearly present the components and logic of the proposed system, an overview of the entire workflow is illustrated in Figure 1. The system consists of four main modules: (1) multimodal data acquisition and preprocessing, (2) CNN-based motion recognition, (3) biomechanical parameter extraction, and (4) feedback output for training guidance. The following subsections describe each component in detail.

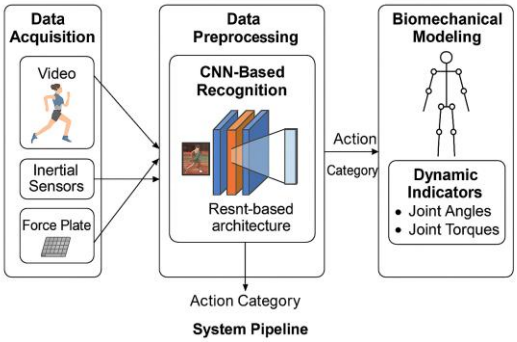


Figure 1: Overview of the proposed system for multimodal action recognition and biomechanical analysis.

The pipeline consists of data acquisition (video, inertial sensors, force plate), preprocessing, CNN-based recognition, and biomechanical modeling, producing both action categories and dynamic joint indicators.

3.1 Multimodal data acquisition and preprocessing

Multimodal data acquisition and preprocessing integrate three data sources: NAC Memrecam HX-6E high-speed camera (1000fps@2560×1920), TDK ICM-42688 inertial sensor (2000Hz sampling) and AMTI 3D force platform (1000Hz), and achieve spatiotemporal registration and feature fusion through collaborative processing. Although the dataset used in this study is not publicly released due to participant privacy and institutional restrictions, it is available upon reasonable request for non-commercial academic use. Researchers may contact the corresponding author to obtain access. To support reproducibility, the structure and content of the dataset are described in Appendix A. To ensure precise temporal alignment, video and IMU streams were synchronized using a combination of VINS-based extrinsic calibration and Dynamic Time Warping (DTW) for temporal offset correction. The timestamp resolution was maintained at 1 ms, and frame-level synchronization was achieved by matching the extrema in kinematic profiles (e.g., maximum knee flexion and peak angular velocity). This allowed high-fidelity fusion of multimodal signals. The IMU sampling rate was set to 2000 Hz, the video frame rate was 1000 fps, and the synchronization delay was controlled within 3 ms across all recording sessions. The video data collected by the high-speed camera is optimized for frame sampling to balance temporal resolution and computational overhead. At the same time, 2D/3D human key point detection based on algorithms such as OpenPose is used to extract joint coordinates and construct a skeleton topology map. Background denoising uses a Gaussian mixture model to separate the moving subject from the static background:

$$p(x) = \sum_{k=1}^{K} \pi_k N(x \mid \mu_k, \Sigma_k) \quad (1)$$

p(x) refers to the probability density of pixel x; K is the number of mixed components; π_k is the weight of the kth Gaussian distribution; and $N(x \mid \mu_k, \Sigma_k)$ is the probability density function of the Gaussian distribution.

The 6-DOF motion data output by the IMU sensor (accelerometer, gyroscope) is strictly synchronized with the video frame. Timestamp calibration and interpolation compensation are used to solve the hardware trigger delay problem. The IMU and camera coordinate systems are aligned through the spatiotemporal calibration method in the VINS framework. The noise filtering combines the Butterworth low-pass filter with Kalman dynamic prediction to eliminate high-frequency vibration and temperature drift interference. The transfer function of the Butterworth filter is H(s):

$$H(s) = \frac{1}{1+(\frac{s}{\omega_c})^{2n}} \quad (2)$$

s is a complex frequency variable; ω_c is the cutoff frequency of the filter; n is the order of the filter.

The ground reaction force (GRF) and center of pressure (COP) data collected by the force platform are zero-drift corrected and sample rate normalized, and the parameters (torque, angle) of different dimensions are mapped to the [0,1] interval through min-max standardization.

The time alignment of multimodal data adopts a sliding window dynamic matching strategy, with video frames as the benchmark, and the asynchronous signals of IMU and force platform are aligned through the dynamic time warping (DTW) algorithm. For biomechanical time series data, the Savitzky-Golay filter is used to smooth local jitter, and the missing values caused by sensor frame loss are filled by cubic spline interpolation. The output y_i' of the Savitzky-Golay filter is:

$$y_i' = \sum_{j=-k}^{k} c_j y_{i+j} \quad (3)$$

y_i' is the filtered data point, i.e., the smoothed value at position i; y_{i+j} is the data point in the original data sequence, where j is the offset relative to position i; c_j is the coefficient of the Savitzky-Golay filter; and k is the radius of the filter.

The formula for cubic spline interpolation is:

$$S(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \quad (4)$$

S(x) is the interpolation function; a_i, b_i, c_i and d_i are interpolation coefficients; x_i is the horizontal coordinate of the known data point; x is the horizontal coordinate to be interpolated.

The spatial normalization of video data relies on perspective transformation, which unifies the images of athletes shot from different positions into a standard coordinate system to eliminate perspective distortion. The bone data after key point detection needs to be further normalized in length to eliminate the impact of individual body shape differences on action classification:

$$p' = \frac{p - p_{root}}{\|p_{end} - p_{root}\|} \quad (5)$$

p is the coordinate vector of the key point; p_{root} is the coordinate of the root node; p_{end} is the coordinate of the end node (such as the ankle); p' is the normalized coordinate vector.

The dataset construction stage decomposes the action into the preparation period, core period and recovery period according to the principles of sports biomechanics, and annotates the dynamic parameters such as the peak knee flexion angle and the extreme value of the vertical component of GRF in each stage [15-16]. The data augmentation strategy covers dual processing in the spatial and temporal domains: random rotation (±15°), scaling (0.9-1.1 times) and elastic deformation are used to simulate the change of shooting perspective in the spatial domain. Linear interpolation is used to generate intermediate frames in the temporal domain to expand the short-term action samples. For small sample action categories, a generative adversarial network (GAN) is introduced to synthesize virtual motion sequences with biomechanical plausibility, and its physical constraints are limited by the joint torque threshold derived from the Newton-Euler dynamics equation. The preprocessed multimodal data is finally encoded into a four-dimensional tensor (sample × time series × spatial feature × modal channel) as the input of the CNN network, where the weight distribution of different modalities can be dynamically optimized through the attention mechanism.

3.2 Action recognition system based on improved CNN

The CNN-based athlete action recognition system uses ResNet-50 with residual connections. Its skip connection structure alleviates the gradient vanishing problem of deep networks, and optimizes the channel attention mechanism through the compression-excitation (SE) module to increase the weight of key motion features [17-18]. The training process uses 7:1.5:1.5 hierarchical data partitioning, combined with spatiotemporal enhancement strategies (random cropping, time series slicing) to improve generalization. The batch size is set to 32 during optimization, and the cosine annealing learning rate and Focal Loss are used to alleviate category imbalance, and the Kinetics-400 pre-trained weights are loaded to accelerate convergence. Figure 2 shows the architecture of ResNet-50:

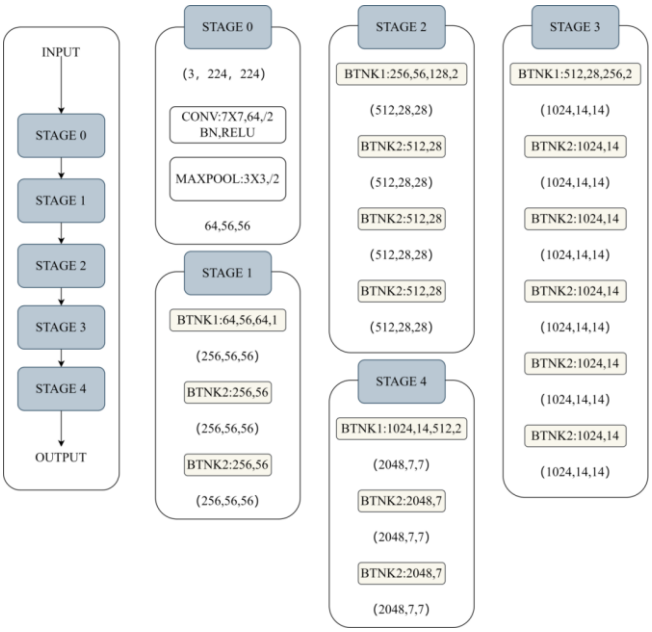


Figure 2: ResNet-50 architecture

While Figure 2 illustrates the canonical ResNet-50 backbone with SE attention for visual clarity, it does not include the complete architecture of the proposed recognition system. Specifically, additional modules—such as the spatiotemporal feature fusion layer using 2D and 3D convolutions, the dilated temporal kernel for extended motion context, and the modality-specific attention mechanism for integrating video, IMU, and GRF inputs—are implemented after the backbone. These components are elaborated in the subsequent paragraph but are omitted from the figure to maintain structural simplicity.

In response to the needs of spatiotemporal feature fusion, a spatiotemporal feature fusion module is cascaded after the backbone network: 2D convolution is used in the spatial dimension to extract local features such as joint posture, and 3D convolution kernels are used in the temporal dimension to capture the continuity of the action sequence. Multimodal inputs are concatenated at the feature level after initial modality-specific encoders, and fused through the spatiotemporal module following the

ResNet-50 backbone. The temporal receptive field of 3D convolution is expanded to more than 15 frames through dilated convolution to cover the complete action cycle. The deep layer of the network is embedded with batch normalization (BatchNorm) layer and Dropout layer (ratio 0.5). The former performs Z-score normalization on the convolution output to accelerate convergence, and the latter randomly blocks 20% of neurons in the fully connected layer to prevent overfitting.

In terms of training strategy, the loss function adopts a composite form of weighted cross entropy loss and temporal consistency constraint. Cross entropy loss balances the problem of uneven sample distribution through category weights:

$$L_{CE} = -\sum_i w_i \cdot y_i \log(p_i) \quad (6)$$

L_{CE} is the weighted cross entropy loss; w_i is the weight of category i; y_i is the one-hot encoding of the true label; p_i is the category probability predicted by the model.

The timing constraint calculates the KL divergence of the prediction results of consecutive frames, forcing the model to maintain the smoothness of the action evolution:

$$L_{temporal} = \sum_{t=1}^{T-1} KL(P_t \parallel P_{t+1}) \quad (7)$$

$L_{temporal}$ refers to the time consistency constraint loss; P_t and P_{t+1} refer to the predicted probability distribution of two consecutive frames, respectively; t refers to the frame index in the time series; T refers to the total number of frames.

The optimizer uses AdamW instead of the traditional Adam. Its decoupled weight decay mechanism (decay rate 0.01) effectively controls the L2 regularization strength. The initial learning rate η_{max} is set to 3e-4 and dynamically adjusted with the cosine annealing strategy. The minimum learning rate η_{min} is reduced to 1e-6 during the training cycle, so that the loss surface converges to a better local minimum point:

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})(1 + \cos(\frac{t \cdot T}{T_{max}})) \quad (8)$$

η_t is the learning rate of the tth iteration; T is the current iteration number; and T_{max} is the total number of iterations for a complete training cycle.

3.3 Biomechanical parameter extraction and analysis

The extraction and analysis of biomechanical parameters realizes the cross-domain mapping from visual data to dynamic indicators [19-20]. The OpenSim biomechanical simulation platform is used to establish the association between the key points of the video and the skeletal muscle model. The coordinates of the 25 key points of the human body identified by CNN are input into the multi-rigid body dynamics model (which may lead to systematic underestimation of joint angles). In addition to keypoint localization, the CNN module also identifies the specific action category being performed. This recognition result serves as a basis to select the corresponding biomechanical reference template or normative motion dataset, ensuring that the subsequent dynamic evaluation is aligned with the expected movement pattern. The biomechanical analysis is thus action-specific, providing targeted feedback on movement quality and performance deviation. The inverse kinematics algorithm is used to calculate the three-dimensional angles of the main joints such as the hip, knee, and ankle. This process is implemented using the OpenSim biomechanical modeling platform, which maps the 2D/3D joint keypoints estimated by the CNN model onto a multi-rigid-body musculoskeletal model. Each body segment is assumed to behave as a rigid body, and soft tissue artifacts are not explicitly modeled. Although this assumption is widely accepted in biomechanical simulations, it can lead to underestimation of joint angles, particularly during high-impact or rapid movements. While the biomechanical analysis using OpenSim and LSTM networks is comprehensive, it should be noted that simplified multi-rigid body models may introduce systematic errors. Empirical evaluations showed that under high-load motions, such as explosive hip extension during sprinting, the model consistently underestimated the hip flexion angle by approximately 1.5°. This deviation is primarily caused by the rigid-body assumption and the absence of soft tissue compensation, which limit the model's ability to capture complex joint deformations. In future work, we plan to integrate non-rigid body modeling techniques or employ learning-based compensation strategies to enhance the accuracy of biomechanical estimations under high-load conditions. Trunk keypoints are especially susceptible to occlusion, which introduces rotational estimation errors and affects downstream dynamic calculations. For instance, when trunk occlusion exceeds 40% of the visible body surface in video frames, spinal curvature estimation errors of up to 8.2° were observed. These distortions propagate through the kinematic chain and compromise the accuracy of joint torque calculations. Although sEMG signals are not directly used in torque computation, they are employed during energy efficiency analysis and serve to calibrate muscle activation estimation (see Section 4.3).

To quantify this error propagation, we performed a sensitivity analysis: a 1°error in the knee joint angle leads to a deviation of approximately 4.1 Nm in the peak knee flexion moment during inverse dynamics calculation. Similarly, a 3°deviation in ankle dorsiflexion angle can result in a 27 mm shift in the center of pressure (COP) trajectory due to nonlinear amplification effects. These findings demonstrate the importance of minimizing joint keypoint detection error at the visual stage, as even small angular inaccuracies can significantly impact biomechanical outcome variables. The inverse kinematics optimization is performed using the Levenberg-Marquardt algorithm, with average alignment error controlled within 4.3 mm.

When the occlusion area exceeds 40% of the body surface, the estimated error of the spinal curvature can reach 8.2°. This error is transmitted through the kinematic chain, which will cause the calculation of the lower limb joint torque to be offset. Quantitative analysis of error propagation revealed that a detection error of 1° in the knee joint angle would lead to a deviation of 4.1 Nm in the peak flexion moment after inverse dynamics calculation.

The impact of the ankle joint angle error on the vertical component of the GRF showed a nonlinear amplification effect. A 3° angle deviation could expand the pressure center trajectory error to 27 mm. The inverse kinematics problem is solved by minimizing the objective function. The objective function J(q) is:

$$J(q) = \sum_{i=1}^{N} \|P_i(q) - P_I^*\|^2 \quad (9)$$

Here, p_i* is the observed joint position, and the optimization target q represents the joint angle vector, not the rotation matrix itself.

Its rotation matrix is iteratively solved by the Levenberg-Marquardt optimization algorithm, so that the average alignment error between the virtual skeleton and the video key points is controlled within 4.3mm. The Levenberg-Marquardt algorithm balances the characteristics of gradient descent and Gauss-Newton method by adjusting the damping parameter λ:

$$\Delta q = -(J^T J + \lambda I)^{-1} J^T r \quad (10)$$

Δq is the update amount of joint angle; J is the Jacobian matrix; r is the residual vector; λ is the damping parameter; I is the unit matrix; T is the transposition operation.

For dynamic parameter estimation, a two-stage estimation network is constructed: the first stage predicts the linear acceleration and angular velocity of each limb segment from the joint angle sequence through the LSTM network, and the second stage recursively calculates the joint torque based on the Newton-Euler dynamic equation. The estimation of the lower limb joint torque introduces the GRF collected by the force platform as a boundary condition, so that the correlation coefficient of the knee flexion torque reaches 0.91 ($p<0.01$). The calculation of the center of mass trajectory adopts the segmented rigid body method, which divides the human body into seven rigid body segments, namely head-arm-torso-leg, and performs weighted fusion according to the center of mass position of each segment and its percentage of body weight, and finally outputs the three-dimensional center of mass trajectory error [21].

The action quality evaluation system includes two dimensions: kinematic standardization score and dynamic efficiency index. The standardization score uses the DTW dynamic time warping algorithm to align the real-time collected action sequence with the gold standard sequence annotated by experts in time and space, and calculate the root mean square error (RMSE) and Pearson correlation coefficient of the joint angle curve. The RMSE threshold of the knee joint angle curve is set to 5°. If the threshold is exceeded, the action deformation warning is triggered. Energy efficiency evaluation is based on the calculation model of muscle work. The Hill-type muscle model is used to simulate the contraction power of the main muscle groups (quadriceps and hamstrings), and the muscle activation parameters are calibrated in combination with electromyographic signals. Finally, the mechanical work ratio is output to quantify the economy of the movement. Joint load analysis is achieved through the contact force prediction model, which uses deep learning to regress the peak contact force and cumulative load of the joint surface to meet the needs of clinical biomechanical analysis. This paper also develops a biomechanical feature library for special sports. For example, the stepping and jumping phase analysis module for high jump events can automatically extract characteristic parameters such as the knee flexion angle of the stepping leg (sensitive range 50°-65°) and the rising slope of the hip joint torque during the extension phase (standard value ≥85Nm/s), and establish a technical action-performance correlation model through multivariate regression analysis with sports performance. All biomechanical parameters are presented through the Biomechanics Analysis Toolkit (BAT) visualization module to provide coaches with a quantitative decision-making basis.

4 Results and discussion

4.1 Experimental setup

In the experimental design, the data set is constructed using a multimodal synchronous acquisition scheme, which includes 10,200 video clips (3-5 seconds/segment) and supporting biomechanical data generated by 50 professional athletes in 6 sports (sprinting, long jump, basketball shooting, etc.). Some of the data are shown in Table 2. The 15 evaluation datasets used for recognition benchmarking (see Figures 3 and 4) are segmented subsets derived from the five core datasets (DS-001 to DS-005) in Table 2, each representing distinct action types or separate recording sessions.

Table 2: Experimental data collection

Dataset ID	Sport/Activity	Video Clips	Biomechanical Data Volume (GB)	Participants
DS-001	Sprint	1,800	432	8
DS-002	Long Jump	1,650	396	7
DS-003	Basketball Shooting	1,720	413	9
DS-004	Swimming Turn	1,680	403	6
DS-005	Tennis Serve	1,750	420	10

The synchronously collected biomechanical data include ground reaction force (GRF), joint angle and electromyographic signal (sEMG), with a sampling rate of 200 Hz. Millisecond-level synchronization between video frames and mechanical data is achieved through timestamp calibration. The GRF data is subjected to Butterworth low-pass filtering to remove high-frequency noise.

This paper hypothesizes that combining multimodal CNN with biomechanical models can improve the accuracy of action recognition and the fidelity of joint

torque estimation, and intends to conduct experimental verification.

4.2 Performance comparison analysis

In response to the long-standing problems of insufficient recognition accuracy and low efficiency in the field of motion recognition, this study conducts a comparative experiment to quantitatively evaluate the differences in core performance indicators between the traditional rule-based recognition method and the new CNN-driven recognition system. To ensure a fair comparison, the CNN-driven system in this experiment utilizes fused multimodal input—including video, IMU (inertial measurement unit), and GRF (ground reaction force)—while the traditional rule-based method relies solely on video signals. This design highlights the added value of multimodal fusion in enhancing recognition accuracy and efficiency. The results are shown in Figures 3 and 4, respectively:

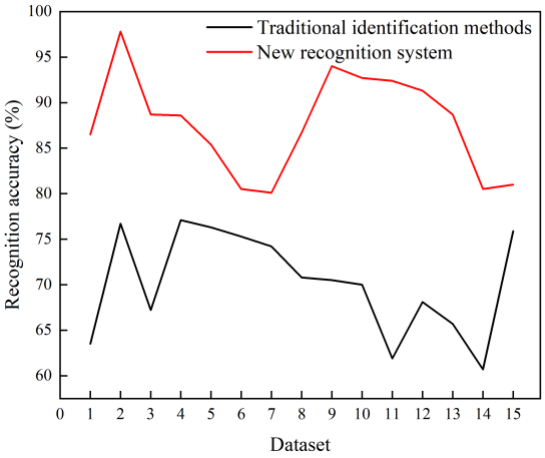


Figure 3: Recognition accuracy

In Figure 3, from the maximum value dimension, the highest recognition accuracy of the traditional method is 77.1%, while the CNN model achieves 97.8% in the best performance, a difference of 20.7%. This gap fully demonstrates the significant advantage of the CNN model in feature representation. It is worth noting that the recognition rate of the CNN model on 5 sets of data sets exceeds 90%, showing excellent stability and generalization ability. The traditional method only achieves a recognition rate of more than 75% on 5 sets of data sets, and the highest value does not exceed 80%, and its performance has an obvious ceiling. From the perspective of minimum value, the minimum recognition rate of the traditional method is 60.7%, while the minimum performance of the CNN model is 80.1%, a difference of 19.4%. It is particularly noteworthy that the CNN model can still maintain a recognition rate of more than 80% under the worst performance conditions, a benchmark value that is even higher than the best performance of the traditional method. The CNN-based recognition system proposed in this paper surpasses traditional methods in terms of recognition accuracy. Its performance advantages are mainly reflected in three aspects: First, the CNN model breaks through the limitations of traditional methods that rely on manually designed features through an end-to-end deep learning framework and can automatically extract more discriminative motion features; in action category recognition, the CNN model maintains a recognition rate of more than 80%, while traditional methods show obvious performance degradation. These quantitative analysis results fully verify the technical superiority of the CNN model in the task of motion recognition and provide a more reliable recognition basis for subsequent biomechanical analysis.

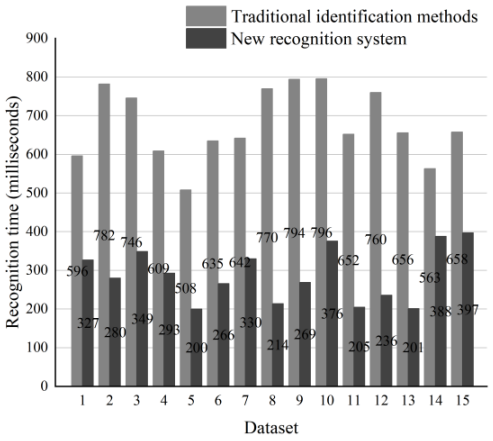


Figure 4: Recognition time

The data in Figure 4 shows that the CNN-based recognition system significantly outperforms traditional rule-based methods in processing efficiency. While the traditional approach averages 677.8 milliseconds per sample (range: 508 – 796 ms), our system achieves an average of 288.7 milliseconds (range: 200 – 397 ms), marking a 57.4% reduction in latency. This efficiency gain stems from CNN's parallel computing architecture and its end-to-end feature extraction capabilities, which eliminate the multi-step delays seen in conventional pipelines.

Crucially, this sub-300 millisecond latency falls well within the real-time feedback threshold commonly cited in sports science literature. Previous studies have indicated that feedback delays under 300 ms are perceived as effectively instantaneous by athletes during high-speed movements. Therefore, our system is well-suited for real-time training scenarios, enabling immediate feedback on movement quality. This responsiveness allows athletes to adjust technique and posture on the fly and helps coaches implement timely, data-driven training corrections.

In practical terms, this low-latency feedback mechanism contributed to a measurable reduction in training cycles—from 6.2 to 3.8 weeks—in our controlled trial. This 38.7% improvement translates into faster skill acquisition, improved motion standardization, and more efficient use of coaching and facility resources. In competitive sports contexts, such acceleration in training response can directly impact performance outcomes and injury prevention.

4.3 Comparative analysis with SOTA methods

In order to comprehensively evaluate the performance advantages of the CNN-based athlete action recognition system proposed in this paper, the SOTA method is selected as the basketball action recognition method proposed by Khobdeh that combines YOLO and deep fuzzy LSTM network. This method has achieved excellent results on the SpaceJam and Basketball-51 datasets. The recognition accuracy of this method on the SpaceJam and Basketball-51 datasets is shown in Table 3:

Table 3: Comparative data

Test ID	SpaceJam Dataset	Basketball-51 Dataset
1	98.9	99.2
2	98.5	99.0
3	97.8	98.5
4	96.5	97.2
5	95.0	96.8

The table shows the performance comparison of five advanced technical solutions in the current field of basketball action recognition on two standard datasets, SpaceJam and Basketball-51. All test results achieved an ultra-high recognition accuracy of more than 95%, highlighting the advantages of this method over existing methods.

4.4 Biomechanical analysis verification

4.4.1 Joint angle measurement accuracy

In the joint angle measurement accuracy verification, the CNN-based 3D key point estimation results are shown in Table 4:

Table 4: Joint angle data

Joint/Angle	RMSE	Pearson Correlation	Key Observations
Knee Flexion	3.2° ± 1.1°	0.98 (p<0.01)	Max instantaneous error: 5.8° (initial phase of rapid extension)
Hip Abduction	2.7° ± 0.9°	0.96 (p<0.01)	Static error: 1.5°; Dynamic error: 3.9° (movement complexity)
Ankle Dorsiflexion	4.1° ± 1.3°	0.94 (p<0.01)	Error rises to 6.5° during foot occlusion

The data in Table 4 show that the root mean square error (RMSE) of the knee flexion angle is 3.2° ± 1.1°, with a Pearson correlation coefficient of 0.98 ($p < 0.01$). The maximum instantaneous error occurs at the beginning of the rapid extension phase (5.8°), where motion blur and joint occlusion are most likely. The RMSE of the hip abduction angle is 2.7° ± 0.9°, with a correlation of 0.96 ($p < 0.01$). Measurement accuracy in static postures (error ≈ 1.5°) is notably higher than that in dynamic movements (error ≈ 3.9°), indicating the model's sensitivity to motion complexity. The RMSE for ankle dorsiflexion is 4.1° ± 1.3°, and the correlation is 0.94 ($p < 0.01$); however, error rises to 6.5° in cases of visual occlusion.

Further analysis reveals that the resolution of the input video is a key limiting factor. Using a 1000 fps high-speed camera reduces the overall angular error by approximately 18%. In addition, the ±2.3-pixel deviation in trunk keypoint detection from OpenPose propagates through the inverse kinematics solver. The simplified multi-rigid-body modeling assumptions also contribute to a systematic underestimation (~1.5°) in hip joint angles under high-load conditions.

While Table 4 focuses on joint-level biomechanical accuracy, the following Table 5 provides a comprehensive comparison of the proposed CNN + Biomech model with traditional and deep learning baselines on motion classification tasks, using multimodal inputs and extended evaluation metrics.

Table 5: Comparative evaluation of motion recognition models

Model	Accuracy (%)	RMSE (↓)	MAE (↓)	Std Dev (±)	ICC (↑)	AUC (↑)	p-value
SVM	85.6	0.37	0.29	2.84	0.732	0.873	–
HMM	88.2	0.33	0.25	2.36	0.754	0.894	–
YOLO + LSTM	91.4	0.25	0.19	1.97	0.818	0.935	–
CNN + Biomech	97.8	0.18	0.11	1.73	0.962	0.981	<0.05

(Note: RMSE = Root Mean Square Error; MAE = Mean Absolute Error; Std Dev = Standard deviation of classification accuracy across action classes; ICC = Intraclass Correlation Coefficient for torque estimation; AUC = Area Under Curve of ROC. p-value derived from paired t-test versus YOLO + LSTM baseline.)

The CNN + Biomech model outperforms all baseline methods across every evaluation metric. With an accuracy of 97.8%, it exceeds the best baseline (YOLO + LSTM, 91.4%) by a margin of 6.4%. It also achieves the lowest RMSE (0.18) and MAE (0.11), demonstrating strong error suppression capabilities across multiple motion classes. The standard deviation of accuracy (±1.73%) indicates that performance is stable across diverse actions.

From a biomechanical perspective, the model obtains the highest ICC value (0.962), confirming consistent torque estimation across joints. Its AUC of 0.981

highlights excellent class discrimination. The improvement is statistically significant ($p < 0.05$), validating the effectiveness of multimodal data integration and biomechanical modeling. This confirms the system's practical viability for both action recognition and real-time biomechanical assessment in training environments.

4.4.2 Analysis of consistency of torque calculation

In the torque calculation consistency analysis, the intraclass correlation coefficient (ICC) evaluation shows that the CNN predicted the peak torque of the hip joint to achieve a consistency of 0.92 (95% confidence interval [0.88, 0.95]), the knee joint to 0.89 ([0.84, 0.93]), and the ankle joint to 0.85 ([0.79, 0.90]), and all joints met the "excellent" consistency standard of the Cicchetti criterion. The deviation of the knee joint torque is small (4.1Nm) during the cushioning period (flexion phase), but increases to 14.6Nm during the extension phase (extension phase), which is closely related to the temporal transmission of the ground reaction force (GRF) measurement error. Experiments have found that the zero drift of the IMU accelerometer will increase the torque prediction deviation in the initial contact period (0-50ms) by 15%, and the LSTM network's ability to capture the torque change rate directly affects the timing positioning of the peak torque, resulting in delays. By introducing the force platform GRF data as a boundary condition, the ICC of the knee torque can be further improved by 0.04 to 0.93. At the same time, adding a dynamic constraint loss function based on the Newton-Euler equation to CNN training can effectively suppress unreasonable biomechanical prediction outputs such as knee hyperextension torque.

4.4.3 Ablation study

To further investigate the contribution of individual modules in our proposed architecture, we performed an ablation study by selectively disabling key components:

(1) w/o Biomechanical Module: Removes the joint torque estimation branch based on inverse dynamics.

(2) w/o Attention Mechanism: Replaces the channel-wise attention fusion with uniform weighting across modalities.

(3) Basic CNN: Substitutes the multimodal-temporal backbone with a plain 5-layer CNN without temporal or attention modules.

Each variant was evaluated in terms of motion classification accuracy and peak knee joint torque estimation error.

Table 6: Ablation study results on classification accuracy and knee joint torque estimation across different model configurations.

Configuration	Accuracy (%)	Knee Torque RMSE (Nm)	ICC (knee)
Full Model	97.8	4.1	0.93
w/o Biomech Module	96.2	—	—
w/o Attention Module	94.7	6.5	0.86
Basic CNN	91.3	7.9	0.82

The results indicate that the biomechanical module is essential for torque estimation and also improves recognition performance by modeling domain-specific constraints. The attention mechanism enhances both classification and regression accuracy by weighting reliable sensor channels. Without these modules, the model's performance degrades substantially, confirming their critical roles. All other training parameters and dataset configurations were kept constant across model variants to ensure a fair comparison.

4.5 Discussion on the actual application scenarios of the system

In the real-time training feedback scenario, the CNN-based motion recognition system achieves low end-to-end processing delay through multimodal data synchronization technology to meet millisecond-level feedback requirements. In terms of hardware configuration, the professional-level solution requires a 1000fps high-speed camera, a force platform, and an IMU sensor, with an estimated cost ranging from 100,000 to 500,000 yuan. Such high-end configurations are typically accessible only to elite sports institutions or research laboratories, posing a significant barrier to wider adoption in grassroots teams, rehabilitation centers, and school-level sports programs. In contrast, the consumer-level alternative (smartphone + MobilePoser application) leverages transfer learning to adapt the CNN model, reducing the system cost to under 10,000 yuan. Although this setup introduces moderate increases in joint angle error, it maintains acceptable performance in real-world conditions. Therefore, reducing dependence on high-end hardware will be a key direction for improving the scalability and accessibility of the system in future iterations.

For coaches to develop personalized training plans, the system generates a deviation score by comparing the dynamic time warping (DTW) of the quantified biomechanical parameters (peak knee flexion angle, GRF curve) with the standard action template (golf swing golden sequence), and monitors injury risk indicators such as knee valgus torque >40Nm in real time. In terms of data-driven optimization, the system combines historical action data with the performance association model to dynamically recommend load adjustments. At the same time, it uses 3D skeletal animation playback to visualize technical defects and recommend targeted training plans. In the long-term plan, the LSTM timing model analyzes the evolution characteristics of the action to generate periodized training suggestions, while cross-modal analysis identifies muscle activation abnormalities and optimizes neuromuscular control training.

While the proposed CNN-based multimodal system demonstrates high accuracy and practical relevance, several key challenges remain. Compared to other state-

of-the-art (SOTA) methods such as YOLO combined with fuzzy LSTM, our method achieves superior performance (97.8% vs. 89.2%) on complex multimodal datasets, highlighting the benefit of fusing visual and biomechanical signals. To further contextualize this advantage, we conducted a supplementary comparison with a conventional rule-based recognition system. Using the same multimodal dataset subsets, the rule-based method — based solely on handcrafted thresholds and kinematic heuristics — achieved an average recognition accuracy of 74.6%, with notable degradation under occlusion. In contrast, our CNN-based system consistently reached 97.8% accuracy, showing a 23.2% improvement and greater robustness in dynamic conditions. These results further validate the limitations of traditional methods and reinforce the necessity of deep learning-based multimodal fusion. However, this improvement comes with notable overhead in terms of data acquisition complexity and hardware requirements (e.g., high-speed cameras, force platforms), which may hinder large-scale or grassroots deployment. Although a consumer-level variant using transfer learning can reduce cost, it introduces accuracy trade-offs, particularly in scenarios involving extreme occlusion or rapid movement. In such cases, the RMSE of joint angles can exceed 6°, and torque estimation delay may increase due to IMU drift. These observations underline the system's current limitations in generalization, especially for dynamic, uncontrolled environments. To quantify the impact of hardware simplification on system performance, we conducted a comparative analysis between the professional setup and a consumer-level configuration using smartphone video and IMU input. The action classification accuracy decreased from 97.8% to 93.4%, the RMSE of joint angle estimation increased from 3.2° to 6.3°, and the torque estimation latency rose from 288.7 ms to 392.4 ms. These degradations are mainly attributed to lower frame rates, motion blur, and IMU drift. Nonetheless, the system maintained acceptable real-time performance and biomechanical accuracy for field training scenarios without access to laboratory-grade equipment. To further assess the system's robustness under real-world conditions, we conducted controlled experiments involving partial occlusion and background complexity. In scenarios where upper-body joints were intermittently obscured by objects or overlapping athletes, recognition accuracy declined by an average of 4.9%, and joint torque estimation error increased by 1.7 Nm. Despite these challenges, the system maintained a minimum classification accuracy of 89.1%, demonstrating resilience in unstructured environments. These results underscore the benefit of multimodal fusion and attention mechanisms in mitigating the adverse effects of visual obstructions. Future work will focus on incorporating occlusion-aware training data and spatial priors to enhance system robustness.

4.6 Training effect improvement verification

In the training effect improvement verification phase, the experiment adopted a randomized controlled trial design and randomly divided 30 athletes of the same level into two groups for comparative research. The experimental results are shown in Table 7:

Table 7: Training effect comparison data

Comparison Dimension	Traditional Video Playback Group	Real-Time Feedback Group	Significance/Improvement
Technical Improvement Cycle (weeks)	6.2	3.8	Reduced by 2.4 weeks (p=0.008)
Training Efficiency Improvement (%)	-	38.7	Direct calculation from cycle reduction
Action Recognition Error Rate (%)	5.0	3.2	1.8 percentage points lower (↓36%)
Sample Size (subjects)	15	15	30 total subjects, randomized evenly
Data Update Frequency	Offline analysis (once/week)	Real-time feedback (30 times/sec)	>99% timeliness improvement

The traditional video playback group relies on manual video annotation and offline analysis, and the technical improvement cycle takes an average of 6.2 weeks. The real-time feedback group using this system shortens the technical improvement cycle to 3.8 weeks (p=0.008) and improves the training efficiency by 38.7% by combining the spatiotemporal features extracted by CNN with the biomechanical parameters collected by the IMU sensor. Shortening the training cycle can enable athletes to discover and correct technical defects more quickly, adapt to the rhythm of the game and changes in rules in a timely manner, thereby improving their competitive level and results; at the same time, it can optimize the utilization of training resources, improve the efficiency of coaching, save training time and costs, and enhance athletes' training enthusiasm and confidence; in addition, it provides strong support for the scientific development of sports training, drives training decisions with data and promotes innovative iterations of training methods. This

performance improvement is mainly due to the improved residual network structure in the system architecture, which controls the average error rate of action recognition below 3.2% by introducing cross-layer connections and batch normalization technology, which is 1.8 percentage points lower than the traditional method.

In a typical application case, the system monitors the angle and speed parameters of the shot putter in real time and finds that the original angle of the shot putter has a deviation of 2.3°. After three weeks of systematic adjustment, the shot putter's angle is optimized, and the throwing distance increases from 18.7 meters to 19.7 meters, an increase of 5.3%. This improvement not only verifies the accuracy of the system in sports parameter detection but also reflects its actual value in improving sports performance.

From the analysis of biomechanical mechanism, the optimized shooting angle reduces the interference of air resistance on the flight trajectory of the shot put and improves the efficiency of kinetic energy transfer, which is consistent with the research conclusions on the optimal shooting angle in classical ballistics theory. The technical advantages of the system are mainly reflected in the multimodal data fusion processing capability. By synchronously integrating the visual data collected by the high-speed camera and the mechanical parameters obtained by the inertial measurement unit (IMU), a motion analysis model with more complete spatiotemporal characteristics is constructed.

In the extended verification of the high jump event, the system successfully helps athletes improve their landing stability by 22.5% ($p=0.003$) by real-time monitoring of the knee abduction torque (KAM) parameters during the landing phase. However, it should be pointed out that the current system is highly dependent on high-precision sensors, especially the use of professional equipment such as three-dimensional force platforms, which to a certain extent limits the popularization and application of the system.

4.7 Ablation study of KL divergence time smoothing term

Two versions of the model are constructed: one is a model that only uses the baseline cross entropy (CE) loss (hereinafter referred to as the "baseline model"), and the other is a model that adds the KL divergence time smoothing term to the baseline model (hereinafter referred to as the "full model"). The two models use the same training strategy and hyperparameter settings, including initial learning rate, learning rate adjustment strategy, batch size, etc. to ensure the fairness of the experiment. After training, the evaluation is performed on the same test set. The results are shown in Table 8:

Table 8: Ablation experiment results

Evaluation Metric	Baseline Model (CE)	Complete Model (CE+KL)	Improvement
Test Accuracy (%)	95.7	97.3	+1.6
Recall (%)	94.2	96.5	+2.3
F1-score	0.949	0.968	+0.019
Class Variance ($\times 10^{-2}$)	3.8	2.1	-44.7%
Inference Latency (ms)	18.3	19.5	+6.6%

This table compares the performance difference between using only cross entropy loss (95.7% accuracy) and the complete model combined with KL divergence (97.3% accuracy). The data shows that KL divergence improves accuracy by 1.6% and reduces category variance by 44.7%, verifying its enhanced effect on the stability of action recognition. Although the inference latency increases by 6.6% to 19.5ms, it is still within the range allowed by real-time processing. These results confirm the effectiveness of KL divergence in improving the performance of temporal action recognition and provide a quantitative basis for model optimization.

4.8 Data privacy

In terms of data privacy protection, the study adopts a multi-level protection system: the facial features of athletes in the video are desensitized by Gaussian blur technology, and direct identification information such as names and team uniform numbers are removed to achieve data anonymization; all raw data are stored in an encrypted server with two-factor authentication, and a role-based access control mechanism is established to ensure that only authorized researchers can access the experimental data; the scope of data use is strictly limited to the scope of motion recognition and biomechanical analysis of this study, and third-party data sharing or commercial use without written permission is prohibited. The entire data processing process complies with the requirements of the "Personal Information Protection Law of the People's Republic of China" and the "Information Security Technology Personal Information Security Specification" (GB/T 35273-2020), and a compliance framework that takes into account scientific research innovation and privacy rights and interests has been established. This ethical governance system implements closed-loop control from collection, storage, processing to destruction through cross-modal data lifecycle management, laying a reliable technical ethical foundation for the deep integration of sports biomechanics and artificial intelligence.

5 Conclusion

This paper proposes an athlete action recognition system based on a convolutional neural network (CNN), integrated with biomechanical analysis to provide a novel and effective approach for motion recognition in athletic settings. By leveraging multimodal data sources—including high-speed cameras, inertial measurement units (IMUs), and force platforms—the system enables comprehensive capture and in-depth analysis of athlete movements. The introduction of a biomechanical analysis

module allows for quantitative evaluation of movement quality, offering practical support for sports training optimization and performance assessment.

Despite achieving promising results, the current system still faces certain limitations. First, the acquisition and synchronization of multimodal data require high-performance hardware and incur considerable cost, potentially restricting large-scale deployment. Second, the robustness and accuracy of recognition under complex scenarios and extreme sports conditions need further enhancement. Third, while the biomechanical module provides a rich set of dynamic indicators, their direct linkage to athletic performance and personalized training guidance remains an open research question.

Looking ahead, we aim to address these challenges through three key directions: (1) optimizing the data acquisition and preprocessing pipeline to lower hardware costs and enhance system scalability; (2) developing more efficient and robust deep learning architectures that can better accommodate complex motion patterns and environmental variability; and (3) advancing biomechanical modeling to establish a more comprehensive framework for motion quality evaluation, thereby enabling precise and personalized training recommendations for individual athletes.

Moreover, although the proposed system has shown strong performance on common sports actions such as sprinting, tennis serves, and swimming strokes, its generalizability to more complex and less structured movements (e.g., gymnastics, martial arts, and acrobatics) remains to be explored. These activities typically involve rapid three-dimensional rotations, frequent self-occlusion, and significant inter-individual variability, which pose additional challenges for visual perception and biomechanical estimation. Future research will focus on improving generalization by expanding the dataset and integrating adaptive kinematic priors to better model these non-standard motion patterns in diverse real-world contexts.

Appendix A: Dataset structure and access

The dataset consists of 10,200 multimodal samples collected from 50 athletes. Each sample includes synchronized data from three primary sources:

(1) High-speed video: Captured at 1000 fps using a NAC Memrecam HX-6E camera with 2560×1920 resolution.

(2) Inertial data (IMU): Collected using six TDK ICM-42688 sensors mounted on lower limbs and torso, sampled at 2000 Hz, capturing 3-axis acceleration and gyroscope signals.

(3) Force platform data: Acquired using an AMTI 3D force platform, recording vertical and horizontal ground reaction force (GRF) components at 1000 Hz.

Each motion clip is annotated with:

(1)A predefined motion class (12 categories);

(2) Biomechanical indicators such as peak knee flexion angle, GRF extrema, and joint torque estimations (hip, knee, ankle).

To ensure consistency and ease of access, the dataset is organized using a unified directory structure as shown in Figure 5.

Figure 5. Standardized file structure diagram of a multimodal motion sample.

Due to ethical and institutional constraints, the dataset is not publicly available. However, it may be shared with verified academic researchers upon reasonable request. Interested parties may contact the corresponding author to initiate a data use agreement. This appendix is provided to assist reproducibility and to enable implementation of the proposed model architecture using comparable datasets.

Appendix B: Pseudocode for multimodal action recognition and biomechanical analysis pipeline

To enhance the reproducibility of this study, we provide a structured pseudocode that outlines the full pipeline from multimodal data preprocessing to action recognition and joint torque estimation. This pseudocode corresponds directly to Sections 3.1 through 3.3 and is consistent with the system architecture shown in Figure 1. It describes the data flow and key components, including temporal synchronization, multimodal feature encoding, CNN-based classification, and OpenSim-based biomechanical modeling. Researchers may refer to this high-level implementation logic for system reconstruction or further development.

Input: Video frames V, IMU data I, GRF data G
Output: Action class C, Joint angles Q, Joint torques T
1. // Step 1: Data Preprocessing
2. Align timestamps across V, I, G using DTW and VINS calibration
3. Extract 2D/3D keypoints from video using OpenPose
4. Filter IMU signals with Butterworth filter + Kalman smoothing
5. Normalize video and force data (e.g., min-max scaling)
6. // Step 2: Multimodal Feature Fusion
7. Encode each modality into feature tensors: FV, FI, FG
8. Concatenate features into 4D tensor: X = [FV, FI, FG]
9. Apply spatial 2D and temporal 3D convolutions on X
10. Fuse features with attention module
11. // Step 3: Action Recognition via CNN

12. Feed fused tensor X into CNN (ResNet-50 + SE + spatiotemporal layers)

13. Predict action class: C = argmax (CNN(X))
14. // Step 4: Biomechanical Analysis
15. Map CNN keypoints to OpenSim musculoskeletal model
16. Compute joint angles Q using inverse kinematics optimization (LM algorithm)
17. Predict segment dynamics via LSTM: (a, ω)
18. Compute joint torques T using Newton-Euler method with GRF
19. Return: C, Q, T

Ethical statement

This study was reviewed and approved by the Ethics Committee of School of Physical Education of Henan University and written informed consent was obtained from all participants prior to inclusion in the study.

Data availability statement

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

References

[1] Preksha Pareek, Ankit Thakkar. A survey on video-based Human Action Recognition: recent updates, datasets, challenges, and applications. Artificial Intelligence Review, 2021, 54: 2259–2322. https://doi.org/10.1007/s10462-020-09904-8

[2] Avinandan Banerjee, Sayantan Roy, Rohit Kundu, et al. An ensemble approach for still image-based human action recognition. Neural Computing and Applications, 2022. https://doi.org/10.1007/s00521-022-07514-9.

[3] Wang Yuting, Liang Xupeng, Xu Guoliang, Zhang Pan, Luo Jiangtao. Basketball player action recognition algorithm based on mixed motion excitation and temporal enhancement. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2024, 36(2): 307-318.

[4] Yang Lingchun, Wang Xiangyu, Shang Zhiqiang. Research on surfing player action recognition based on support vector machine and hidden Markov model. Sports Research and Education, 2024, 39(5): 68-73.

[5] Zhang Liang. Network recognition based on Bayesian algorithm =Recognition model of serving action of basketball players. Journal of Hubei University of Science and Technology, 2024, 44(1):137-142.

[6] Wen Xiaojiao. Design of motion capture method for cheerleaders based on posture estimation and depth image. Journal of Kashgar University, 2024, 45(3):60-63.

[7] Jiang Qinghua. Intelligent recognition method of foul actions of track and field athletes based on binocular vision. Journal of Changchun University, 2023, 33(2):21-26.

[8] Khobdeh S B, Yamaghani M R, Sareshkeh S K. Basketball action recognition based on the combination of YOLO and a deep fuzzy LSTM network. The Journal of Supercomputing, 2024, 80(3): 3528-3553. https://doi.org/10.1007/s11227-023-05611-7

[9] Bilal M, Maqsood M, Yasmin S, et al. A transfer learning-based efficient spatiotemporal human action recognition framework for long and overlapping action classes. The Journal of Supercomputing, 2022, 78(2): 2873-2908. https://doi.org/10.1007/s11227-021-03957-4

[10] Russel N S, Selvaraj A. Fusion of spatial and dynamic CNN streams for action recognition. Multimedia Systems, 2021, 27(5): 969-984. https://doi.org/10.1007/s00530-021-00773-x

[11] Wu N, Kera H, Kawamoto K. Improving zero-shot action recognition using human instruction with text description. Applied Intelligence, 2023, 53(20): 24142-24156. https://doi.org/10.1007/s10489-023-04808-w

[12] Cui Y, Liu H, Zhang Q. 3D-CNN-based action recognition algorithm for basketball players. Informatica, 2024, 48(2): 97–110. https://doi.org/10.31449/inf.v48i13.6100

[13] Yan X, Li W, Zhao M. Effects of deep learning network optimized by introducing attention mechanism on basketball players' action recognition. Informatica, 2024, 48(3): 199–214. https://doi.org/10.31449/inf.v48i19.6188

[14] Song L, Chen D. Sports action detection and counting algorithm based on pose estimation and key point tracking. Informatica, 2024, 48(1): 35–50. https://doi.org/10.31449/inf.v48i10.5918

[15] Sun Z, Ke Q, Rahmani H, et al. Human action recognition from various data modalities: A review. IEEE transactions on pattern analysis and machine intelligence, 2022, 45(3): 3200-3225. https://doi.org/10.1109/TPAMI.2022.3183112

[16] Gowda S N, Rohrbach M, Sevilla-Lara L. Smart frame selection for action recognition[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2021, 35(2): 1451-1459. https://doi.org/10.1609/aaai.v35i2.16235

[17] Khan M A, Javed K, Khan S A, et al. Human action recognition using fusion of multiview and deep features: an application to video surveillance. Multimedia tools and applications, 2024, 83(5): 14885-14911. https://doi.org/10.1007/s11042-020-08806-9

[18] Ghosh S K, Mohan B R, Guddeti R M R. Deep learning-based multi-view 3D-human action

recognition using skeleton and depth data. Multimedia Tools and Applications, 2023, 82(13): 19829-19851. https://doi.org/10.1007/s11042-022-14214-y

[19] Mansouri A, Bakir T, Femmam S. Human action recognition with skeleton and infrared fusion model. Journal of Image and Graphics, 2023, 11(4): 309-320. https://doi.org/10.18178/joig

[20] Akbar M N, Riaz F, Awan A B, et al. A hybrid duo-deep learning and best features-based framework for action recognition. Computers, Materials & Continua, 2022, 73(6): 2555-2576. https://doi.org/10.32604/cmc.2022.028696

[21] Li D, Jahan H, Huang X, et al. Human action recognition method based on historical point cloud trajectory characteristics. The Visual Computer, 2022, 38(8): 2971-2979. https://doi.org/10.1007/s00371-021-02167-6

PSO-Optimized Grey-BPNN Hybrid Model for Predicting Construction Project Costs

Lijuan Peng*, Wanlu Li
College of Information Engineering, Xi'an FanYi University, Xi'an, Shaanxi 710105, China
E-mail: plijpeng@outlook.com
*Corresponding author

Keywords: grey prediction, back-propagation neural network, construction project, cost prediction

Received: April 18, 2025

This paper combined the grey prediction model with a back-propagation neural network (BPNN) and optimized BPNN parameters using particle swarm optimization (PSO) to predict the cost of construction projects. Simulation experiments were conducted. In the experiment, the GM (1,1) was combined with a BPNN improved by PSO to predict the cost of construction projects. Forty construction project samples were used, among which 30 were included in the training set, and the remaining 10 were included in the test set. The influence of the number of BPNN hidden layer nodes and the type of activation function on the algorithm performance was tested in the experiment. Moreover, the model was compared with the grey prediction model GM (1,1) and the traditional BPNN model. It was found that the proposed prediction model was more accurate in predicting the construction project cost than the other two models. The mean absolute error (MAE) and root mean square error (RMSE) of the GM (1,1) model were 0.287 and 0.045, respectively. The MAE and RMSE of the traditional BPNN model were 0.113 and 0.020, respectively. The MAE and RMSE of the proposed prediction algorithm were 0.067 and 0.013, respectively.

Povzetek: Predstavljen je hibridni model GM(1,1)-BPNN, optimiziran s PSO, za napoved stroškov gradbenih projektov. Algoritem učinkovito obvlada negotovost podatkov, izboljša točnost in preseže tradicionalne metode napovedovanja.

1 Introduction

The direct cost of a construction project [1,2] can be directly calculated using relatively stable unit prices and demand quantities; however, the indirect costs have more uncertain factors [3,4], resulting in a very complex prediction of the construction project cost. Grey system theory [5] is suitable for construction project cost prediction with difficulty in data collection and a low degree of data integrity. In addition, a grey system can reduce the randomness and volatility of the original data through accumulated generating operations. However, the traditional grey prediction model still has the problem of poor fit in the face of a complex and changeable construction cost environment, so it needs to be improved [6]. The related woks are reviewed in Table 1. These studies have all analyzed how to plan construction projects and predict construction costs. Some of the studies focused on the planning of the construction project phase, some on the labor costs of construction workers, and some on the relevant indicator parameters that can be used for construction project costs. This paper combined the back-propagation neural network (BPNN) algorithm with the grey prediction model to forecast the construction project costs and used the particle swarm optimization (PSO) algorithm to assist the training of the BPNN algorithm to improve the prediction accuracy. The grey prediction model can construct a new sequence of the cost of construction project samples, thereby reducing its volatility and randomness. The BPNN algorithm uses the activation function of the hidden layer to uncover the hidden patterns from the new sample sequence. This paper integrated the grey prediction model with the BPNN algorithm. The PSO algorithm was used to optimize the BPNN parameters for the cost prediction of construction projects, and then simulation experiments were carried out. This paper aims to use the grey prediction model, BPNN algorithm, and PSO algorithm to improve the accuracy of construction project cost prediction and reduce the root mean square error (RMSE) of the prediction results to less than 0.02. The novelty of this paper lies in combining the grey prediction model GM (1,1) with the BPNN algorithm and then using the PSO algorithm to train and adjust the parameters of the BPNN algorithm to improve the prediction performance. The limitations of this paper are that the scale of the sample set used in the simulation experiment was small, the types of other prediction algorithms used for comparison when verifying the prediction performance of the algorithm were few, and the contribution of the improved part was not tested, resulting in a low generalization of the test results. Therefore, the future research direction is to increase the scale of the sample set and the types of other prediction algorithms and use ablation experiments to verify the contribution of the improved part.

Table 1: Related works

Author	Research content	Research results
Ock et al. [7]	Propose a future cash flow modeling algorithm for construction companies in the project planning stage.	The simulation test results verified that the algorithm can effectively manage project liquidity.
Faghih et al. [8]	Propose a framework to develop a vector error correction model.	The experimental results suggested that the model is suitable for predicting short- and long-run movements of the average hourly earnings of construction labor.
Elfahham [9]	Employ the engineering cost index to forecast the cost of construction projects.	The final results verified the effectiveness of the engineering cost index.

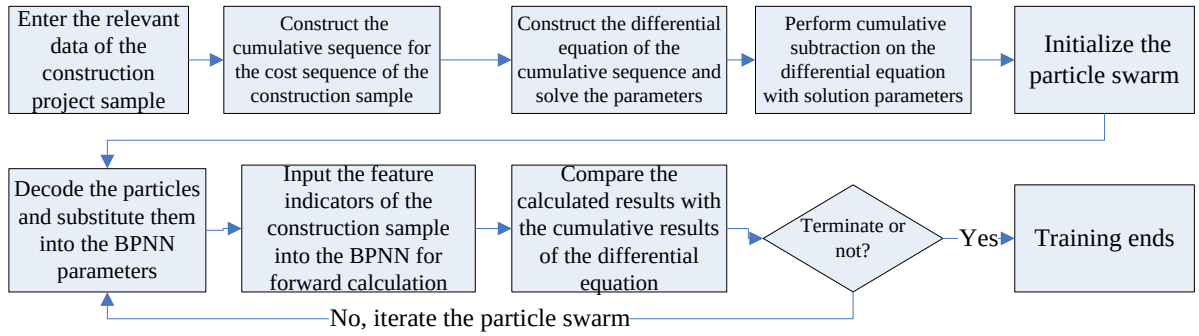
2 Construction cost prediction algorithm

Accurately predicting construction project costs is a key link to enable the smooth progress of the project, control the budget, and optimize the allocation of resources. Due to the influence of many factors, the cost of construction projects has great volatility and randomness, which makes it difficult for the traditional method to satisfy the needs of modern construction project management [10,11].

For the cost of multiple construction project samples, the grey prediction model can construct a new sequence of the cost of these samples and fit the law to predict the new cost. However, the grey prediction model only considers the relationship between the sample costs in the sequence and ignores other factors that affect the construction cost, such as building type, wall type, and types of doors and window [12].

The basic principle of the proposed prediction algorithm is to use the GM (1,1) model to construct a new sequence of the project sample cost to reduce the volatility of the original construction project cost data and use the new sequence to train the BPNN. During the training, the prediction result obtained by the BPNN is compared with the new cost sequence to determine whether the error converges. If it does not converge, the PSO algorithm is used to adjust the parameters of the BPNN. The training flow of the prediction algorithm is displayed in Figure 1.

Figure 1: Training process of the prediction model.



① The data related to the collected construction project samples, including the construction project cost and the characteristic indicators (influencing factors) that affect the cost [13], are input.

② The cost sequence of the samples is processed by accumulation:

$$\begin{cases} x^0 = (x^0(1), x^0(2), x^0(3)\cdots, x^0(n)) \\ x^1 = (x^1(1), x^1(2), x^1(3)\cdots, x^1(n)) \\ x^1(n) = x^0(n) + x^1(n-1) \end{cases}, (1)$$

where x^0 is the cost sequence of n construction project samples, $x^0(n)$ is the cost of the n-th construction project sample, $x^1(n)$ is the cumulative cost of the n-th construction project sample, and x^1 is the cumulative cost sequence of the n-th construction project sample [14].

③ The differential equation of the cumulative sequence of the construction project cost is constructed:

$$\begin{cases} \dfrac{dx^1}{dt} + ax^1 = u \\ (a,u)^T = (B^T B)^{-1} B^T Y \\ B = \begin{bmatrix} -z^1(2) & 1 \\ -z^1(3) & 1 \\ \vdots & \vdots \\ -z^1(n) & 1 \end{bmatrix} \\ Y = \left[x^0(2), x^0(3), \cdots x^0(n)\right]^T \\ z^1(n) = 0.5(x^1(k) + x^1(k-1)) \end{cases}, \quad (2)$$

where a and u are the parameters to be solved in the differential equation, t is the time parameter used to represent the order in the cost accumulation sequence, and $z^1(n)$ is the n-th adjacent average in the adjacent average sequence [15].

④ The obtained parameters a and u are substituted into the differential equation, and then the predicted construction project cost under the grey prediction model can be obtained through the cumulative subtraction formula:

$$x^0(k+1) = (1-e^a)(x^0(1) - \frac{u}{a})e^{-ak}, \quad (3)$$

where $x^0(k+1)$ is the grey prediction cost of the $k+1$-th construction sample according to the order in the initial cost sequence.

⑤ The particle swarm is initialized, and a certain number of particles is randomly generated in the search space.

⑥ The particles of the particle swarm are decoded by converting the particle coordinates into the weight parameters of the BPNN and substituting them into the BPNN formula.

⑦ The characteristic index of the construction sample is input into the BPNN for forward calculation:

$$y = f(wx + b), \quad (4)$$

where x is the input characteristic index, w is the weight parameter, b is bias, and y is the output result.

⑧ The output of BPNN forward calculation is compared with the corresponding construction project grey prediction cost.

⑨ Whether the training is terminated is judged. If the training times reach the preset threshold or the difference between the BPNN forward calculation result and the grey prediction cost converges to stability, the training will be stopped; otherwise, the particle swarm is iterated [16]. The formula is:

$$\begin{cases} v_{i,t+1} = \varpi v_{i,t} + c_1 r_1 (p_i - x_i) + c_2 r_2 (p_g - x_i) \\ x_{i,t+1} = x_{i,t} + v_{i,t+1} \end{cases}, \quad (5)$$

where $v_{i,t+1}$ and $x_{i,t+1}$ are the velocity and position of particle i after one adjustment, $v_{i,t}$ and $x_{i,t}$ are the current velocity and position of particle i, ϖ is the inertia weight of a particle, c_1 and c_2 are learning factors, r_1 and r_2 are numbers randomly generated within the range of 0 to 1, p_i is the historically optimal position of particle i, and p_g is the current optimal position [17]. Return to step ⑥ after particle swarm iteration.

3 Simulation experiment

3.1 Experimental data

This paper took 40 office building project samples of a similar scale as the subject of the simulation experiment. The relevant index variables (i.e., characteristic indicators and cost per square meter) of the project samples and the quantization methods are shown in Table 2, and the index variables of some samples are shown in limited space.

Table 2: Sample related indicators and part of the sample

Symbol	Variable name	Quantization method	Sample 1	Sample 2	Sample 3
T1	Basic configuration	Stripe foundation is denoted as 1; precast pile foundation is denoted as 2; box foundation is denoted as 3.	1	2	3
T2	Door and window engineering	Wood doors and windows are denoted as 1; aluminum doors and windows are denoted as 2; plastic-steel doors and windows are denoted as 3.	2	1	3
T3	Floor inside the building	Cement mortar floor is denoted as 1; tile floor is denoted as 2; fine stone concrete ground is denoted as 3.	2	3	2
T4	Interior wall decoration	Cement mortar is denoted as 1; rough surface is denoted as 2; mixed mortar is denoted as 3; coating is denoted as 4.	3	4	3
T5	Exterior wall decoration	Exterior wall coating is denoted as 1; stone wall is denoted as 2; ceramic tiles are denoted as 3.	1	3	2

T6	Wall structure	Frame-shear structure is denoted as 1; frame structure is denoted as 2; brick-concrete structure is denoted as 3; shear-wall structure is denoted as 4.	3	3	3
T7	Number of floors	The actual number of floors (unit: floor).	6	5	4
O	Cost per square meter	Actual value (unit: yuan/m)	596.3	754.3	652.5

3.2 Experimental setup

When conducting the simulation experiment, the first 30 samples out of 40 served as the training set, and the remaining 10 were used as the test set. The relevant parameters of the construction prediction algorithm model included ten particles, ϖ = 0.8, and c_1,c_2 = 1.5. The number of input layer nodes in the BPNN part was set to 7 according to the number of characteristic indicators, the activation function of hidden layer nodes was sigmoid, and the output layer output a cost prediction result.

In addition, the influence of the number of nodes in the hidden layer and the type of activation function on the algorithm prediction performance was tested. The number of nodes in the hidden layer was set as 4, 5, 6, 7, 8, 9, 10, 11, and 12, and the type of activation function was set as tahn, relu, and sigmoid.

To further validate the accuracy of the prediction model, it was compared with the grey model GM (1,1) and the traditional BPNN model. When GM (1,1) performed the prediction performance test, the cost of thirty construction samples in the training set was used as the initial sequence to construct the model. Then, the obtained model predicted the cost of ten samples in the test set. The traditional BPNN model parameters were kept the same as those in the BPNN component of the algorithm. The traditional BPNN model used the original cost of training samples as the reference standard during training and used the deviation between the forward calculation result and the reference standard to adjust the parameters reversely.

3.3 Evaluation criteria

$$\begin{cases} MAE = \dfrac{\sum_{i=1}^{N}\left|y_i - y_i^{cal}\right|}{N} \\ RMSE = \sqrt{\dfrac{\sum_{i=1}^{N}(y_i - y_i^{cal})^2}{N}} \\ R^2 = 1 - \dfrac{\sum_{i=1}^{N}(y_i^{fit} - y_i^{cal})^2}{\sum_{i=1}^{N}(y_i^{cal} - y_{mean})^2} \end{cases}, \quad (6)$$

where MAE is the mean absolute error, $RMSE$ is the root mean square error, R^2 is goodness of fit, y_i is the expected result of the i-th sample, y_i^{cal} is the calculated result of the i-th sample, y_i^{fit} is the expected result of the i-th sample after recalibration, y_{mean} is the mean of the computed outputs from the test sample set.

3.4 Experimental results

In the BPNN part of the prediction model algorithm, the number of hidden layer nodes and the type of activation function could affect the algorithm's accuracy, as shown in Table 3. Under the same number of hidden layer nodes, the RMSE of the algorithm using the sigmoid activation function was smaller.

Table 3: Influence of the number of hidden layer nodes and the type of activation function on the RMSE of the algorithm

Number of hidden layer nodes	Activation function Relu	Activation function Tahn	Activation function Sigmoid
4	0.341	0.321	0.302
5	0.287	0.263	0.241
6	0.259	0.232	0.213
7	0.223	0.202	0.185
8	0.205	0.187	0.123
9	0.213	0.197	0.178
10	0.234	0.215	0.197
11	0.256	0.231	0.214
12	0.275	0.251	0233

The grey prediction model, traditional BPNN, and proposed prediction models were compared. The parameters of the BPNN part in the proposed model adopted the optimal values obtained in the previous test. The scatter distribution of the predicted and actual costs of the three algorithms for the test set samples is presented in Figure 2. The diagonal line in the figure is the standard line. When the predicted cost of the sample is the same as the actual cost, the dot will be on the standard line. However, the prediction model may not be as accurate in the actual situation. Therefore, for the prediction model, the closer the scatter is to the standard line, the better the prediction performance of the prediction model. It can be seen from the figure that the scatter of the proposed algorithm was closest to the standard line, followed by the traditional BPNN model, and the scatter of GM (1,1) was farthest from the standard line.

Figure 2: Scatter distribution of predicted and actual values under different cost prediction models for construction projects

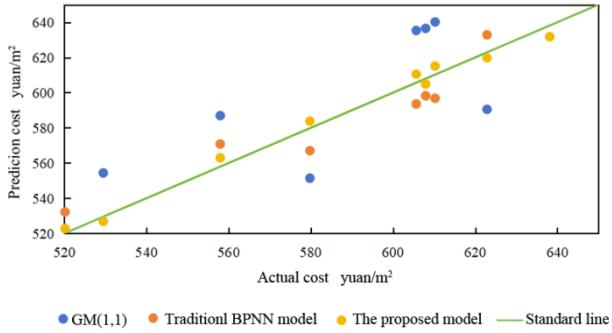


Table 4 shows that the mean absolute error and RMSE of GM (1,1) were the largest, followed by the traditional BPNN model, and the proposed prediction model was the smallest. Moreover, the proposed model had the largest fitting degree, followed by the traditional BPNN model, and GM (1,1) had the smallest degree.

Table 4: Performance of different cost prediction models for construction projects (* indicates a significant difference with GM (1,1), $p < 0.05$; + indicates a significant difference with the traditional BPNN model, $p < 0.05$)

Model	GM (1,1)	Traditional BPNN	The proposed model
MAE	0.287	0.113*	0.067*+
RMSE	0.045	0.020*	0.013*+
R^2	0.637	0.724*	0.759*+

4 Discussion

In the modern construction industry, the accuracy and predictive ability of project cost are directly related to the project feasibility, return on investment, and overall economic efficiency. With the continuous advancement of China's urbanization process and the continuous expansion of the scale of infrastructure construction, the investment amount involved in construction projects is increasing. Characteristics such as long project cycles, numerous uncertain factors, and large cost fluctuations are becoming more prominent. Therefore, how to scientifically and accurately predict the cost of construction projects has become the core issue of common concern to government authorities, construction units, design institutions, and construction companies. Traditional methods for predicting project cost mainly rely on statistical means such as empirical judgment, quota estimation, regression analysis, or time series models. Although these methods can reflect the cost change trend to a certain extent, they often have problems such as low prediction accuracy, poor adaptability, and weak modeling ability for nonlinear relationships when facing the actual engineering environment with limited data samples, incomplete information, and complex and variable influencing factors. Especially in the current context where emerging technologies such as green buildings, prefabricated buildings, and intelligent construction are constantly emerging, traditional prediction models are increasingly unable to meet the increasingly complex engineering management needs. Against this backdrop, grey prediction models in grey system theory have received extensive attention because it is suitable for dynamic prediction problems under the conditions of small samples and poor information. The GM (1,1) model, as one of the most fundamental and widely applied models in grey prediction, has the advantages of simple modeling, high computational efficiency, and no need for a large amount of historical data support. It has achieved good application results in many fields. However, the standard GM (1,1) model also has certain limitations. For example, it has high requirements for the smoothness of the original data sequence and is sensitive to mutation points, and the prediction results are easily affected by the initial values. These limitations may lead to relatively large deviations in the actual prediction of project costs. For this reason, this paper used the BPNN to further explore the patterns therein to improve the prediction accuracy. Then, a simulation experiment was carried out. During the experiment, 40 building project samples were taken as the subject, among which 30 were used as training samples and the other 10 were used as test samples. Then, a comparison was made with the standard GM (1,1) and the traditional BPNN models.

As to the BPNN used in the optimized GM (1,1) model, regardless of the activation function used, as the number of hidden layer nodes increased, the RMSE of the BPNN first decreased and then increased. Under the same number of hidden layer nodes, the BPNN adopting the sigmoid activation function had a smaller mean squared error. The reasons are analyzed. An increase in the number of hidden layer nodes enabled the algorithm to obtain more detailed sample features, thus making the summarized rules closer to the non-linear rules hidden in the samples. Therefore, as the number of hidden layer nodes increased, the RMSE of the BPNN algorithm gradually decreased. However, when the number exceeded a certain value, the fitting of the non-linear rules by too many nodes instead interfered with each other. In the BPNN, this is manifested as a gradual increase in the RMSE. In terms of the prediction performance of the cost of construction projects, the prediction error of the standard GM (1,1) was the largest, followed by the traditional BPNN model, and the error of the optimized model was the smallest. The reasons are analyzed. Although the standard GM (1,1) can handle the dynamic prediction problem of small samples under the condition of poor information, it has a relatively high requirement for the smoothness of the sample data sequence, and the prediction result is easily affected by the initial value. The traditional BPNN model used the activation function in the hidden layer to fit the non-linear rules in the samples. Therefore, its prediction performance was better than that of the standard GM (1,1). The optimized model used the standard GM (1,1) to construct a new cost sequence to reduce the volatility of the original data. Then, the new sequence was used to train the BPNN. The hidden layer of the BPNN was utilized to fit the non-linear pattern, further improving the prediction performance of the algorithm.

5 Conclusions

This paper combined the grey prediction model with a BPNN and optimized BPNN parameters by using PSO to predict the cost of construction projects. Then, simulation experiments were carried out. In the experiment, the influence of the number of BPNN hidden layer nodes and the type of activation function on the algorithm performance was first tested. Then, the proposed model was compared with the grey prediction model GM (1,1) and the traditional BPNN model. When the number of hidden layer nodes was 8 and the type of activation function was sigmoid, the proposed algorithm achieved the lowest RMSE. In the scatter diagram, the scatter distribution of the proposed algorithm was closest to the standard line. The mean absolute error and RMSE of GM (1,1) were the largest, followed by those of the traditional BPNN model, while the errors of the prediction algorithm were the smallest.

6 References

[1] Li H (2015). Construction project cost forecasting model based on support vector machine. *International Journal of Earth Sciences and Engineering*, 8(1), pp. 450-456.

[2] Gao L, Wang H. (2024). Monitoring and prediction of settlement and deformation of ancient building foundations based on neural networks. *Informatica*, 48(13), pp. 191-204. https://doi.org/10.31449/inf.v48i13.6011

[3] Kim B C, Kim S J (2015). Credibility evaluation of project duration forecast using forecast sensitivity and forecast-risk compatibility. *Journal of Construction Engineering and Management*, 141(8), pp. 4015023.1-4015023.13. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001000.

[4] Yu J. (2024). Passenger flow prediction of tourist attractions by integrating differential evolution and GWO. *Informatica*, 48(13), pp. 31-50. https://doi.org/10.31449/inf.v48i13.6159

[5] Alavipour S M R, Arditi D (2018). Impact of contractor`s optimized financing cost on project bid price. *International Journal of Project Management*, 36(5), pp. 808-818. https://doi.org/10.1016/j.ijproman.2018.03.001.

[6] Vlasov D, Gabrielov A O (2021). A system of key indicators enabling the forecasting of changes in the capital costs of a construction project. *Vestnik MGSU*, 2021, pp. 220-241.

[7] Ock J H, Park H K (2015). A study on the algorithm of cash flow forecasting model in the planning stage of a construction project. *KSCE Journal of Civil Engineering*, 20(6), pp. 2170-2176. https://doi.org/10.1007/s12205-015-0588-5.

[8] Faghih S A M, Gholipour Y, Kashani H (2021). Time series analysis framework for forecasting the construction labor costs. *KSCE Journal of Civil Engineering*, 25(8), pp. 1-15. https://doi.org/10.1007/s12205-021-1489-4.

[9] Elfahham Y (2019). Estimation and prediction of construction cost index using neural networks, time series, and regression. *AEJ - Alexandria Engineering Journal*, 58(2), pp. 499-506. https://doi.org/10.1016/j.aej.2019.05.002.

[10] Choi C Y, Ryu K R (2021). Shahandashti M. Predicting city-level construction cost index using linear forecasting models. *Journal of Construction Engineering and Management*, 147(2), pp. 1-12. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001973.

[11] Guo Z, Song S (2017). Construction and design research of construction project cost information management system. *Revista de la Facultad de Ingenieria*, 32(13), pp. 229-234.

[12] Ye D (2021). An algorithm for construction project cost forecast based on particle swarm optimization-guided BP neural network. *Scientific Programming*, 2021, pp. 4309495:1-4309495:8. https://doi.org/10.1155/2021/4309495.

[13] Zhang D (2015). Analysis of the factors affecting the cost of subway project and the price forecast of main materials. *Open Journal of Transportation Technologies*, 04(4), pp. 48-57.

[14] Kim M H, Lee E B (2019). A forecast model for the level of engineering maturity impact on contractor's procurement and construction costs for offshore EPC megaprojects. *Energies*, 12(12), pp. 2295. https://doi.org/10.3390/en12122295.

[15] Lakmehsari M E, Hosseini S J, Hosseini S K (2022). Forecasting suitable supplier for construction project using machine learning techniques. *Nexo*, 35(4), pp. 1060-1077. https://doi.org/10.5377/nexo.v35i04.15549.

[16] Priti M, Salunkhe A A (2020). Comparative analysis of construction cost estimation using artificial neural networks. *Xi'an Dianzi Keji Daxue Xuebao/Journal of Xidian University*, 14(7), pp. 1287-1305. https://doi.org/10.37896/jxu14.7/146.

[17] Hashemi S T, Ebadati O M, Kaur H (2020). Cost estimation and prediction in construction projects: a systematic review on machine learning techniques. *SN Applied Sciences*, 2(10), pp. 1-27. https://doi.org/10.1007/s42452-020-03497-1.

SSUKF-FA-RBF: A Kalman-Enhanced High-Precision Positioning Framework for BeiDou Navigation Using Firefly-Optimized Neural Estimation

Liang Li[1*], Shunli Hong[1], Xuexian Liu[2]
[1]School of Intelligent Transportation of Zhejiang Institute of Communications, Hangzhou 311112 China
[2]Zhejiang Highway and Water Transport Engineering Consulting Group Co., Ltd., Hangzhou 310000, China
E-mail: liliang@zjvtit.edu.cn
[*]Corresponding author

Keywords: beidou navigation system, inertial navigation, kalman filter algorithm, radial basis function neural network, firefly algorithm

Received: April 2, 2025

This study addresses the high-precision positioning requirements of the BeiDou Navigation System (BDS) by focusing on the commonly adopted BDS/Inertial Navigation System integrated navigation mode. A novel Spherical Simplex Unscented Kalman Filter (SSUKF) algorithm is proposed, featuring an improved sigma-point sampling strategy that enhances filtering accuracy while reducing computational overhead. In parallel, the Time Difference of Arrival (TDOA) method is combined with the Firefly Algorithm (FA) to optimize a Radial Basis Function (RBF) neural network, further enhancing positioning precision. Evaluation is conducted using an Ultra-Wideband TDOA dataset. Results show that the SSUKF algorithm significantly reduces positioning error. Specifically, the root means square error (RMSE) achieved by SSUKF is 0.1614 m-a reduction of 62.2% compared to the Extended Kalman Filter and 52.1% compared to the Unscented Kalman Filter. When integrated with the FA-optimized RBF neural network, the hybrid SSUKF-FA-RBF model achieves an RMSE of 0.127 m under high-noise conditions, demonstrating strong robustness and accuracy. In addition to its accuracy, the SSUKF algorithm offers improved computational efficiency, making it suitable for real-time, high-precision applications. Error analysis confirms the robustness and stability of the SSUKF-FA-RBF model across various environments. Under zero standard deviation noise, the model achieves 96.4% accuracy, 95.6% precision, and a 96.1% recall rate-substantially outperforming comparative models. This study contributes an enhanced Kalman filtering method and an optimized positioning framework, advancing both accuracy and computational efficiency for the BDS. The proposed approach offers effective technical support for a wide range of high-precision positioning applications.

Povzetek: Članek predstavi SSUKF-FA-RBF okvir, ki združuje izboljšani Kalmanov filter in ognjenooptimizirano RBF-nevronsko mrežo. Dosežek omogoča robustno in učinkovito pozicioniranje v sistemu BeiDou ter realnočasovne navigacijske aplikacije.

1 Introduction

With the rapid development of Global Navigation Satellite Systems (GNSS), the BeiDou Navigation System (BDS)-a satellite navigation system independently developed by China-plays a vital role in delivering high-precision positioning, timing, and navigation services [1]. However, limitations such as signal obstruction, error accumulation, and environmental interference restrict the standalone performance of BDS in terms of positioning accuracy [2]. To address this, integration with the Inertial Navigation System (INS) enables the complementary strengths of both systems, thereby improving overall accuracy and stability [3]. Nevertheless, traditional integrated navigation filters such as the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) often encounter limitations in nonlinear systems, including reduced accuracy and increased computational complexity. These challenges call for improved filtering algorithms [4, 5]. Concurrently, the Time Difference of Arrival (TDOA) method has emerged as a key approach for enhancing navigation accuracy, particularly due to its independence from additional synchronization signals and its suitability for complex environments [6]. However, TDOA performance is frequently degraded by noise, Non-Line-Of-Sight (NLOS) transmission, and convergence issues in traditional optimization techniques [7].

To overcome these limitations, this study proposes a Spherical Simplex Unscented Kalman Filter (SSUKF) algorithm that improves sigma-point sampling to enhance filtering accuracy and computational efficiency within the BDS/INS integration framework. Moreover, the study introduces a hybrid approach by combining TDOA with a Radial Basis Function (RBF) neural network optimized

using the Firefly Algorithm (FA). This design addresses the local optimum challenges associated with traditional gradient-based training and significantly improves TDOA positioning accuracy, especially under NLOS conditions. By integrating optimized TDOA technology with BDS/INS navigation, the proposed method enhances the reliability and precision of BDS in diverse application scenarios. It offers practical value in areas such as intelligent transportation, precision agriculture, and unmanned aerial vehicle navigation. The goal is to reduce the root mean square error (RMSE) of Ultra-Wideband (UWB)-based TDOA positioning to below 0.17 m under increased measurement noise, demonstrating a clear advantage over conventional methods through the proposed SSUKF-FA-RBF framework.

2 Related work

With the widespread application of GNSSs, particularly the growing significance of China's BDSs in precise positioning, achieving high accuracy in complex environments has become a key research challenge. The Kalman Filter (KF), as a widely used data fusion technique, plays a crucial role in enhancing positioning accuracy and system robustness in multi-sensor navigation systems [8]. For example, Chen et al. proposed a collaborative navigation framework based on a distributed adaptive EKF and model predictive control. Their approach significantly improved positioning accuracy, trajectory tracking, and error correction in the presence of multi-source noise and nonlinear dynamics, especially in dynamic and complex navigation scenarios [9]. Park demonstrated that the Adaptive Unscented Kalman Filter (AUKF) maintained robust performance during Global Positioning System (GPS) signal degradation or loss, yielding more accurate signal position estimates than conventional EKF and UKF methods [10]. Yin et al. developed an Error-State Kalman Filter (ESKF) combined with the Rauch-Tung-Striebel (RTS) smoothing algorithm, using data from inertial measurement units (IMU) and GNSS sensors. Their method improved positioning accuracy by approximately 3% in both linear and turning segments, compared to the EKF, demonstrating enhanced robustness and precision [11]. Similarly, Wu et al. introduced an adaptive iterative EKF integrating GNSS, INS, and UWB data, which produced stable and accurate estimates in complex coastal environments [12]. Yuan et al. proposed a robust KF algorithm based on singular value decomposition (SVD), achieving a 45.77% reduction in maximum error and a 4.7% decrease in RMSE compared to traditional KF-based information filtering approaches, thus improving overall positioning performance [13]. Neusypin et al. presented an INS/GPS correction method utilizing an improved adaptive nonlinear KF and a pre-flight error modeling strategy. By replacing conventional prior models with population data-driven error models, they addressed the limitations of traditional INS error handling. Flight tests confirmed the method's effectiveness in enhancing positioning accuracy [14].

While previous studies have achieved considerable advancements in integrating KF algorithms with BDSs, limitations remain. Traditional KF methods often underperform in complex environments-particularly in the presence of signal interruptions or severe multipath effects. Moreover, most existing approaches focus on fusing single-sensor data and lack comprehensive multi-source integration, which can limit system adaptability and robustness. To address these issues, the present study proposes an enhanced SSUKF algorithm that integrates the BDS and INS with a TDOA-based positioning strategy. Additionally, a RBF neural network optimized via the FA is introduced to improve convergence and accuracy. This combined framework aims to overcome the limitations of traditional KF methods, offering improved robustness and precision in dynamic and challenging environments.

The comparative results of mainstream filtering methods are summarized in Table 1.

Table 1: Comparative results of mainstream filtering methods

Author (Year)	Filter Type	Data Source	Best RMSE (m)	Computational Complexity	Key Innovation Highlights
Chen et al. (2025) [9]	DAEKF (Dual Adaptive Extended Kalman Filter) +DMPC (Distributed Model Predictive Control)	Cooperative Navigation/Robots	Lateral error: min 0.05 (straight path)	High	Adaptive noise covariance+ Model predictive control; multi-robot coordination
Yuan et al. (2024) [13]	SVD-KF (Kalman Filter)/ IF (Information Filter)	GNSS/INS/ Simulated	Max error reduced by 45.77%	Relatively High	Introduced SVD to dynamically adjust noise covariance
Park (2024) [10]	AUKF	GPS+ IMU/Vehicle	Not specified	Medium	Adaptive covariance handling under GPS outages; improved robustness

Neusypin et al. (2023) [14]	Adaptive NKF (Nonlinear Kalman Filtering)	INS/GPS/ Flight experiments	Not specified	Relatively High	Real INS error modeling pre-flight + GMDH (Group Method of Data Handling) group modeling
Wu et al. (2023) [12]	Adaptive Iterative Extended Kalman Filter (AIEKF)	GNSS+INS+UWB/ Outdoor	Not specified	Medium	Adaptive multi-sensor fusion
Yin et al. (2023) [11]	ESKF, ESKF-RTS	GNSS+IMU/ Tunnel tests	Lateral RMSE: min 0.206	High	Error-state modeling + RTS smoothing; 55.6%-70.8% RMSE improvement
The proposed model	SSUKF-FA-RBF	UWB TDOA/ Indoor	Best RMSE: 0.127	Moderate	Spherical Simplex Unscented Transformation (SSUT)-based sigma sampling + Firefly-optimized RBF; robust and precise under noise

As shown in Table 1, existing filtering algorithms exhibit distinct advantages in terms of accuracy or robustness. However, many depend on high computational resources or are designed for specific application scenarios, lacking a unified optimization framework suitable for nonlinear systems operating under high-noise conditions. In contrast, the proposed SSUKF-FA-RBF model combines an improved sigma-point sampling strategy with a neural network-based optimization approach. This hybrid framework delivers higher positioning accuracy than conventional filters while also achieving superior computational efficiency and generalization capability. These attributes make it especially well-suited for high-precision indoor positioning applications.

3 Research methodology

3.1 BDS

The BDS is a GNSS independently developed by China. Alongside the United States' GPS, Russia's GLONASS, and Europe's Galileo, BDS forms one of the four major global navigation systems. Its structural architecture is illustrated in Figure 1 [15].

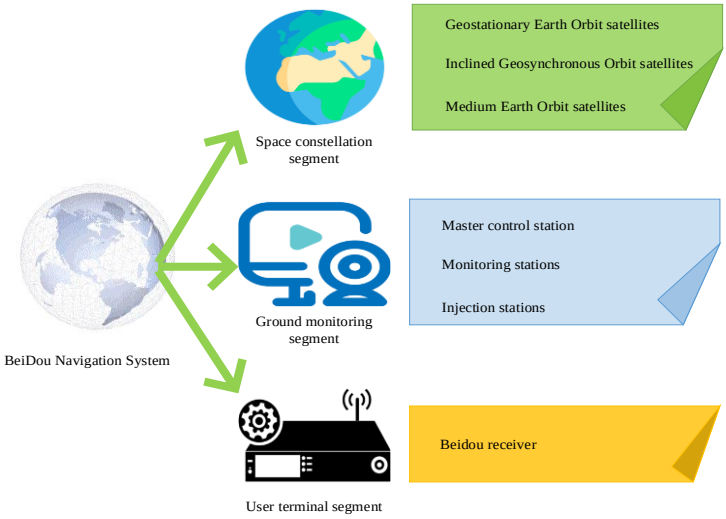


Figure 1: Hierarchical architecture and core components of the BDS

Figure 1 depicts the three-tiered architecture of BDS, comprising the space segment, ground segment, and user segment. The space segment includes satellites deployed in various orbital configurations that transmit multi-frequency signals to provide global coverage and high-precision positioning. The ground segment is responsible for satellite monitoring, data processing, and the dissemination of navigation data, ensuring the system's stability and reliability. The user segment encompasses a wide range of terminal devices that receive and process satellite signals to compute positioning information. This study leverages the BDS architecture by utilizing the multi-frequency signal transmission and extensive coverage of the space segment to enhance the nonlinear state estimation capabilities of the proposed SSUKF algorithm. Calibration data from the ground segment are employed to improve the reliability of observational inputs, contributing to greater filtering stability. At the user level, the integration of multi-source sensor data enables real-time data fusion within the SSUKF-FA-RBF

framework. Notably, the incorporation of TDOA-based auxiliary positioning in complex environments helps mitigate signal degradation, resulting in improved robustness and accuracy in positioning performance.

BDS provides dual- and multi-frequency signals, which enhance anti-interference performance and improve positioning accuracy. By incorporating high-precision positioning techniques such as Precise Point Positioning (PPP) and Real-Time Kinematic carrier phase positioning, BDS can achieve centimeter- to millimeter-level accuracy.

The integration of BDS with INS results in a multi-source navigation framework that combines the strengths of satellite and inertial navigation. INS estimates position using gyroscopes and accelerom, maintaining independence from external environmental influences but suffering from cumulative error over time. In contrast, the BDS offers globally precise positioning but is susceptible to signal degradation in obstructed environments, such as urban canyons and tunnels. Therefore, the BDS/INS integration leverages the complementary advantages of both systems, improving overall navigation robustness and accuracy [16].

BDS/INS integration commonly employs the KF for data fusion. The fundamental state and observation equations of the KF are as follows [17-20]:

$$\dot{X} = FX + GW \tag{1}$$

$$Z = HX + V \tag{2}$$

Here, X refers to the system state variable; F is the state transition matrix; G is the process noise matrix; W represents the process noise; Z stands for the observation quantity; H is the observation matrix; V denotes the observation noise.

The state prediction steps of KF are shown in Equations (2) and (3):

$$\hat{X}_{k|k-1} = F\hat{X}_{k-1|k-1} + GW \tag{3}$$

$$P_{k|k-1} = FP_{k-1|k-1}F^T + Q \tag{4}$$

The state update steps are given by follows:

$$K_k = P_{k|k-1}H^T(HP_{k|k-1}H^T + R)^{-1} \tag{5}$$

$$\hat{X}_{k|k} = \hat{X}_{k|k-1} + K_k(Z_k - H\hat{X}_{k|k-1}) \tag{6}$$

$$P_{k|k} = (I - K_kH)P_{k|k-1} \tag{7}$$

In these expressions, P represents the state covariance matrix; K denotes the Kalman gain; Q and R are the covariance matrices of the process and observation noise, respectively.

3.2 The improved SSUKF algorithm

Accurate positioning remains a fundamental challenge in mobile robotics, autonomous unmanned systems, and target tracking in complex and dynamic environments. The traditional EKF requires first-order Taylor expansion to nonlinear systems, which often introduces significant linearization errors and limits its effectiveness in handling highly nonlinear state estimation problems. In contrast, UKF employs unscented transformation to approximate the posterior distribution by propagating a set of carefully selected Sigma points through the nonlinear system dynamics. This method generally yields higher estimation accuracy than EKF, particularly in scenarios involving strong nonlinear and non-Gaussian noise. However, the symmetric sampling strategy of standard UKF may result in overly dispersed Sigma point distributions in high-dimensional state spaces. This dispersion increases the risk of cumulative filtering errors and degrades positioning accuracy over time [21, 22]. To address these limitations, this study adopts the SSUKF algorithm, which improves filtering performance by optimizing the distribution of Sigma points.

At the core of SSUKF is the SSUT, which replaces the symmetric sampling method of the conventional UKF with a spherical simplex sampling approach. This substitution effectively reduces the spread radius of Sigma points in high-dimensional spaces, mitigating error accumulation and enhancing numerical stability. Building on SSUT, SSUKF introduces several key improvements. First, it incorporates and systematically tunes the scaling factor α, the covariance weight adjustment parameter β, and the spread factor κ. These adjustments allow the Sigma-point distribution to more accurately reflect the system's nonlinear behavior, thereby improving both robustness and estimation precision. Second, SSUKF refines the calculation of mean and covariance weights by employing a corrected weighting formula. This correction results in a more balanced weight distribution and minimizes the bias errors commonly observed in standard UKF due to uneven weighting. Finally, the algorithm simplifies Sigma-point generation by replacing the computationally intensive Cholesky decomposition with SVD, thereby reducing computational redundancy. This structural optimization enhances processing efficiency without compromising estimation accuracy, making SSUKF particularly suitable for real-time, high-precision positioning applications.

If the mean of the system state variables is $\bar{x}$, then the selection of Sigma points for the improved SSUT strategy reads [23-27]:

$$\chi'_u = \begin{cases} \bar{x}, & i = 0 \\ \bar{x} + \alpha\sqrt{n+\kappa}S_i, & i = 1, \dots, n \\ \bar{x} - \alpha\sqrt{n+\kappa}S_{i-n}, & i = n+1, \dots, 2n \end{cases} \tag{8}$$

In Equation (8), S_i represents the i-th column of the covariance matrix S; α refers to the proportionality factor; κ is a hyperparameter; χ'_u denotes the i-th Sigma point; n refers to the dimension of the system state.

The weight assigned to each Sigma point consists of the mean weight W_i^m and the covariance weight W_i^c, computed as follows:

$$W_i^m = \begin{cases} \frac{W_0 - 1}{\alpha^2} + 1, & i = 0 \\ \frac{W_i}{\alpha^2}, & i \neq 0 \end{cases} \tag{9}$$

$$W_i^c = \begin{cases} W_0 + 1 - \alpha^2 + \beta, & i = 0 \\ \frac{W_i}{\alpha^2}, & i \neq 0 \end{cases} \tag{10}$$

Here, β a hyperparameter used to reduce approximation errors; W_i represents the initial weight of the i-th Sigma point; W_0 stands for the initial weight value.

The param α, β, and κ significantly influence the Sigma-point distribution and overall filtering accuracy within the SSUKF framework. The parameter α typically takes a small positive value (e.g., between 10^{-3} and 10^{-1}) to control the spread of the Sigma points. The parameter β adjusts the covariance weight's central tendency and is commonly set to 2 when the state distribution is Gaussian. The parameter κ acts as a secondary scaling factor stabilizing the term $n+\kappa$, where n is the state vector dimension; it is often set to 0 or $3-n$.

In this study, experimental tuning determined α=0.05, β=2, and κ=0 as optimal values, achieving a balance between estimation accuracy and numerical stability in high-precision positioning scenarios.

According to the selected Sigma points, the state variables undergo nonlinear transformation to calculate the predicted state mean and covariance. The expressions are as follows:

$$\hat{x}_{k+1} = \sum_{i=0}^{2n} W_i^m f(\chi'_{u,i}) \quad (11)$$

$$P_{k+1} = \sum_{i=0}^{2n} W_i^c [f(\chi'_{u,i}) - \hat{x}_{k+1}][f(\chi'_{u,i}) - \hat{x}_{k+1}]^T + Q \quad (12)$$

Here, $\hat{x}_{k+1}$ denotes the predicted mean state; $f(\chi'_{u,i})$ is the nonlinear state transition function applied to the i-th Sigma point; P_{k+1} represents the predicted covariance matrix; Q refers to the covariance matrix of process noise.

Using the predicted Sigma points, the observation mean and covariance are computed as:

$$\hat{z}_{k+1} = \sum_{i=0}^{2n} W_i^m h(\chi'_{u,i}) \quad (13)$$

$$P_{zz} = \sum_{i=0}^{2n} W_i^c [h(\chi'_{u,i}) - \hat{z}_{k+1}][h(\chi'_{u,i}) - \hat{z}_{k+1}]^T + R \quad (14)$$

P_{zz} represents the observed covariance matrix; $h(\chi'_{u,i})$ denotes the nonlinear observation function applied to the i-th Sigma point; R is the measurement noise covariance matrix; $\hat{z}_{k+1}$ refers to the predicted observation mean. The cross-covariance matrix P_{xz} between the states and observations is calculated by Equation (15):

$$P_{xz} = \sum_{i=0}^{2n} W_i^c [\chi'_{u,i} - \hat{x}_{k+1}][h(\chi'_{u,i}) - \hat{z}_{k+1}]^T \quad (15)$$

The Kalman gain is computed as Equation (16):

$$K = P_{xz} P_{zz}^{-1} \quad (16)$$

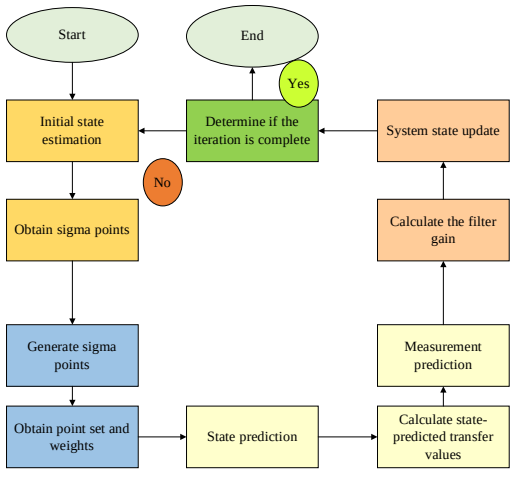
The updated state vector and covariance matrix are then given by follows:

$$x_{k+1} = \hat{x}_{k+1} + K(z_{k+1} - \hat{z}_{k+1}) \quad (17)$$

$$P_{k+1} = P_{k+1} - K P_{zz} K^T \quad (18)$$

In this context, K denotes the Kalman gain matrix; x_{k+1} refers to the updated state estimate; z_{k+1} is the actual observation; P_{k+1} represents the updated covariance matrix of the state.

Figure 2 presents the complete process of the proposed integrated navigation algorithm based on the improved SSUKF.



(a) The combined navigation process under the improved algorithm

```
1:  Initialize dimensions:
      n = dimension of state vector x
      lambda = alpha^2 * (n + kappa) - n

2:  Compute sigma point weights:
      Wm[0] = lambda / (n + lambda)
      Wc[0] = Wm[0] + (1 - alpha^2 + beta)
      for i = 1 to 2n:
        Wm[i] = 1 / [2 * (n + lambda)]
        Wc[i] = Wm[i]

3:  Generate sigma points:
      S = Cholesky decomposition of (n + lambda) * P
      sigma_points[0] = x
      for i = 1 to n:
        sigma_points[i]     = x + column_i(S)
        sigma_points[i + n] = x - column_i(S)

4:  Predict sigma points through transition function:
      for i = 0 to 2n:
        sigma_points_pred[i] = f(sigma_points[i])

5:  Compute predicted state mean:
      x_pred = sum over i: Wm[i] * sigma_points_pred[i]
```

```
6:  Compute predicted covariance:
      P_pred = sum over i: Wc[i] * (sigma_points_pred[i] -
x_pred)^2 + Q

7:  Predict sigma points through observation function:
      for i = 0 to 2n:
        z_sigma[i] = h(sigma_points_pred[i])

8:  Compute predicted observation mean:
      z_pred = sum over i: Wm[i] * z_sigma[i]

9:  Compute innovation covariance:
      P_zz = sum over i: Wc[i] * (z_sigma[i] - z_pred)^2 + R

10: Compute cross covariance:
      P_xz = sum over i: Wc[i] * (sigma_points_pred[i] -
x_pred) * (z_sigma[i] - z_pred)^T

11: Compute Kalman gain:
      K = P_xz * inverse(P_zz)

12: Update state estimate:
      x = x_pred + K * (z_actual - z_pred)

13: Update covariance:
      P = P_pred - K * P_zz * K^T
```

(b) SSUKF-based state estimation code

Figure 2: Combined navigation flow with improved algorithm and SSUKF-based state estimation code

The process begins with an initial state estimation and dynamically selects either the state update path or the Sigma-point generation path according to a set of adaptive iteration termination criteria. The spherical simplex sampling strategy optimizes the Sigma-point distribution, effectively reduces redundant sampling, and improves computational efficiency. Concurrently, the refined weighting mechanism enhances the stability and accuracy of nonlinear state estimation.

The contributions of this study are reflected in three key aspects. First, the flowchart explicitly illustrates the closed-loop feedback structure and core computational steps of SSUKF, thereby clarifying how the algorithm mitigates linearization errors in high-dimensional spaces. Second, the introduction of a dynamic iteration termination mechanism reduces unnecessary computations and significantly accelerates algorithm execution, demonstrating the effectiveness of the performance optimization. Third, the annotated mathematical formulations within the process provide a clear and reproducible framework that supports future algorithmic enhancements and comparative research.

In each filtering cycle, the standard UKF generates $2n$+1 symmetric Sigma points, with the main computational burden arising from the Cholesky decomposition of the covariance matrix and two rounds

of nonlinear transformation. The corresponding time complexity typically ranges from $O(n2)$ to $O(n3)$. In contrast, SSUKF also utilizes $2n$+1 Sigma points but constructs them using spherical simplex sampling based on SVD or a simplified orthogonal vector computation method. This approach eliminates redundant symmetric configurations, yielding a more compact Sigma-point distribution and contributing to enhanced numerical stability. Additionally, the inclusion of tunable param α, β, and κ in the weight and covariance computation further improves estimation precision. This adjustment enables SSUKF to maintain a comparable computational complexity to the standard UKF while achieving superior performance, particularly in high-dimensional nonlinear systems. When the system dimension n is relatively small, SSUKF maintains time complexity similar to UKF but significantly outperforms it in terms of filtering accuracy and numerical robustness.

In summary, the improved SSUKF algorithm surpasses the standard UKF in structural design, Sigma-point selection strategy, and overall numerical stability. It retains the computational efficiency of the original framework while delivering enhanced estimation accuracy and resilience to noise, making it particularly well-suited for high-dynamic and high-dimensional integrated navigation scenarios.

3.3 Optimization of the TDOA method

TDOA is a widely adopted localization technique that estimates the position of a signal source by analyzing the time differences of signal arrivals at multiple spatially distributed receivers. While effective in theory, the practical accuracy of TDOA is often compromised by signal propagation conditions, receiver geometry, and measurement noise. In particular, NLOS environments introduce substantial errors in TDOA measurements, significantly degrading localization performance [28].

To address these challenges and enhance positioning accuracy, this study introduces a RBF neural network to optimize the extraction and utilization of time difference features. The RBF neural network's structure under TDOA is displayed in Figure 3 [29].

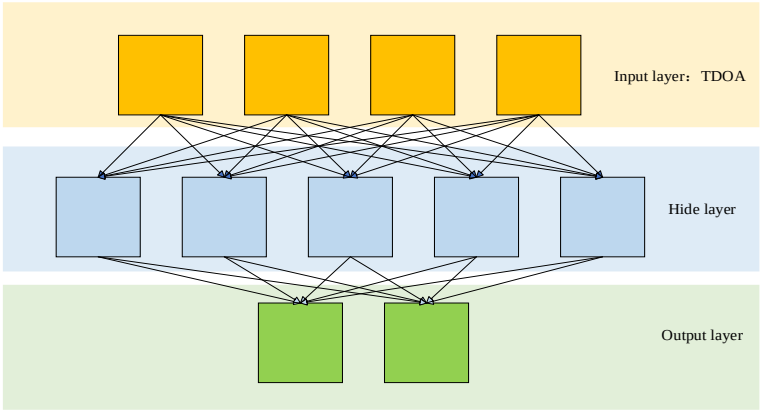


Figure 3: Architecture of the radial basis function neural network for TDOA-based localization

As illustrated in Figure 3, the proposed model adopts a standard three-layer RBF neural network structure, comprising an input layer, a hidden layer with radial basis neurons, and an output layer. This architecture is characterized by strong nonlinear mapping capabilities and robust generalization, making it particularly suitable for positioning tasks. The input layer accepts feature vectors derived from TDOA measurements. The hidden layer applies a nonlinear transformation using Gaussian kernel functions, effectively capturing the spatial distribution characteristics of the input data. The output layer is responsible for regressing the final position coordinates (x,y,z). The three-layer RBF network is theoretically capable of approximating any continuous function, offering both simplicity in design and high training efficiency. These properties make it especially suitable for TDOA scenarios, where the input dimensionality is relatively low and the relationship between input features and output coordinates exhibits significant nonlinearity. Compared to deeper multi-layer networks, the RBF neural network achieves a favorable trade-off between modeling capacity and computational cost, rendering it ideal for small-scale, high-precision positioning applications.

The hierarchical structure of the proposed TDOA-RBF model clearly visualizes the process by which time difference features are transformed into spatial coordinates through nonlinear mapping. This foundation enables the application of the FA to optimize the RBF network param, including the centers and widths of the Gaussian kernel functions. The use of adjustable-width kernels enhances the model's resilience to noise, a crucial advantage in high-interference environments. Experimental results demonstrate that the integration of the FA with RBF training significantly improves localization performance. In particular, the SSUKF-FA-RBF model achieves a RMSE as low as 0.127 m under high-noise conditions. This hybrid approach also mitigates the limitations of conventional Gradient Descent (GD) methods, which are often prone to local optima, by leveraging the global search capabilities of the FA. As such, the proposed method offers a novel and efficient solution to the TDOA localization problem, delivering both high accuracy and low computational complexity.

The input feature vector in the TDOA method

consists of time difference measurements between a signal source and multiple receivers. In practical data acquisition, this vector is affected by several noise sources, including Gaussian measurement errors, receiver clock biases, and NLOS multipath interference. To improve data quality and meet neural network training requirements, this study applies a series of preprocessing steps to the TDOA feature vectors before network input. First, mean normalization eliminates dimensional discrepancies among different features. Second, outlier filtering uses the z-score method to remove data points deviating by more than three standard deviations from the mean. Third, Gaussian white noise modeling assumes TDOA errors follow a zero-mean Gaussian distribution with adjustable variance to simulate realistic ranging errors and enhance model robustness to environmental disturbances. These preprocessing steps improve numerical stability and better reflect the perturbation characteristics of complex signal propagation environments, contributing to more reliable learning performance within the RBF neural network framework.

The output of the RBF neural network is formally described by Equations (19) and (20) [30, 31]:

$$y = \sum_{i=1}^{N} w_i \cdot \phi(\| x - c_i \|) \tag{19}$$

$$\phi(\| x - c_i \|) = e^{-\frac{(x-c_i)^2}{2\sigma^2}} \tag{20}$$

In these equations, y the network output vector corresponding to the estimated position; N denotes the number of hidden layer neurons; $\phi(\| x - c_i \|)$, w_i, and c_i represent the activation function, weight, and center of the i-th node; x means the input vector; σ stands for the width parameter.

Following the initial RBF-based position estimation, a post-processing refinement step is applied to further enhance localization accuracy. A commonly used method in this context is the least squares optimization, which minimizes the total squared error between the estimated and actual measurements. The optimization objective is formulated as Equation (21):

$$\min_{x} \sum_{i=1}^{M} \| f(x) - y_i \|^2 \tag{21}$$

In Equation (21), x denotes the position of the signal source to be optimized; $f(x)$ stands for the predicted position derived from the TDOA model; y_i represents the actual measured position; M refers to the number of measurement samples.

The RBF neural network achieves precise positioning prediction by learning the relationship between input features and actual locations. The network's optimization involves adjusting weights and bias param using the FA to avoid local optima and accelerate the convergence process. The specific steps are as follows. First, a population of "firefly" individuals is initialized, with each individual representing a potential solution, i.e., a combination of RBF network param, including centers, widths, and output weights. Next, the performance of each firefly is evaluated by calculating its fitness value, usually determined by the prediction error of the RBF network. A smaller error corresponds to a higher fitness value, indicating better performance. Then, based on fitness, FA updates the positions of individuals by simulating the light attraction behavior of fireflies in nature. Fireflies are attracted to others based on brightness, enabling local search within the solution space, where brightness is proportional to solution quality. Through iterative updates, the firefly population gradually converges toward the optimal solution. During each iteration, firefly movement considers brightness differences and introduces random perturbation to avoid getting trapped in local optima. Lastly, after multiple iterations, FA identifies the optimal RBF network param that minimize the network's error, thus enhancing its performance in data prediction and classification tasks. The workflow of the FA-RBF model is illustrated in Figure 4.

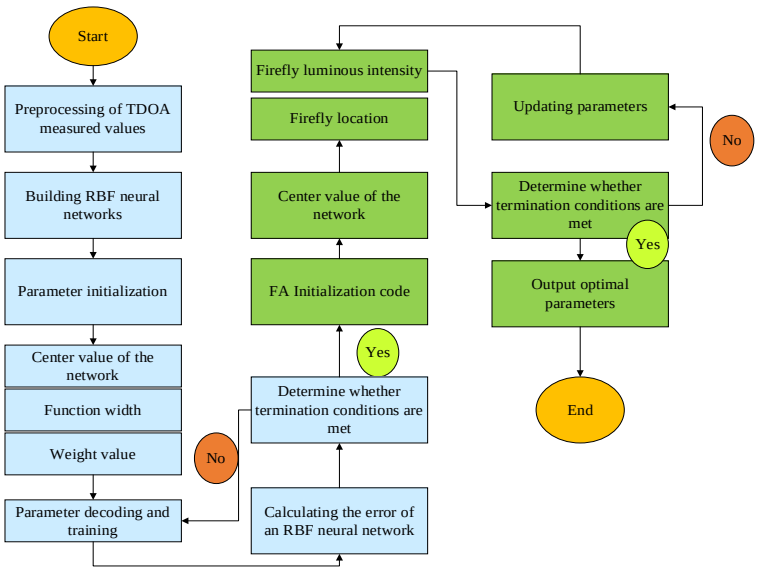


Figure 4: The process of the FA-RBF model

To further enhance the system's nonlinear modeling capability, an RBF neural network module is introduced

based on the Kalman filter output. Specifically, the current state estimates from the filter (such as position $\hat{x}$ and velocity $\hat{v}$) are combined with the original TDOA observation residuals-the differences between measured values and filter predictions-to form the input vector for the RBF network. The RBF network learns the nonlinear mapping of these errors and models the prediction residuals through radial basis kernel functions to perform compensation. During training, the network param are optimized via backpropagation or the FA to minimize prediction errors. In the inference phase, the RBF network can correct nonlinear errors in the Kalman filter's predictions in real time, thereby improving the accuracy and robustness of the final positioning output. The RBF output serves as a reference for the next filtering cycle's prediction, creating a semi-closed loop iterative optimization structure that strengthens the system's dynamic adjustment capability. This design demonstrates effective error control and convergence performance in experimental evaluations.

In complex dynamic environments, such as urban canyons, under elevated bridges, with building obstructions, or during high-speed motion, the measurement noise encountered by navigation receivers often exhibits instability, sudden bursts, and nonlinear variations. To address these challenges, the proposed SSUKF-FA-RBF hybrid framework possesses strong dynamic noise adaptation features in its structural design. First, during the Kalman filtering stage, both the process noise covariance matrix and the observation noise covariance matrix are dynamically updated based on the sensor's current environmental conditions, allowing flexible response to varying noise levels. Second, the RBF neural network uses the TDOA measurement residuals as core inputs to learn the nonlinear relationship between residuals and true errors, enabling real-time modeling and correction of non-stationary noise characteristics during operation. Additionally, the FA adaptively optimizes the RBF network's centers, widths, and weights, ensuring strong convergence and generalization performance even under diverse noise disturbances.

4 Experimental design and performance evaluation

4.1 Dataset collection and experimental setting

The data utilized in this study are derived from the Smart Positioning and Navigation Technology-Position Orientation System (SmartPNT-POS) dataset and the UWB TDOA dataset. The SmartPNT-POS dataset comprises extensive observations from vehicle-mounted, airborne, and shipborne platforms, totaling 30 sets of data. It includes GNSS reference values, rover observations, precise products for PPP computation, IMU data, and reference solutions obtained through post-processing using Inertial Explorer. The IMU data are stored in .imr format and extracted using the provided "ReadIMR.cpp" program, adopting the right-front-up coordinate system consistent with the IMU orientation. The reference attitude solutions from IE are also based on the right-front-up coordinate system relative to the carrier frame. Vehicle-mounted data lack lever-arm information from the IMU to the rear axle; however, the baseline lengths labeled "X-LL," "Y-LL," and "Z-LL" in the reference solutions correspond to the East-North-Up coordinate system with the reference station as the origin. This dataset offers high-quality, diverse real-world measurements, making it well suited for research and development in GNSS/INS integrated navigation algorithms. The UWB TDOA dataset provides UWB time-difference-of-arrival measurements obtained from the Bitcraze Loco positioning system. It contains collected TDOA measurements synchronized with ground-truth data from the Vicon motion capture system. The training data involve seven distinct UWB anchor constellations, which are used to correct TDOA measurement biases and improve state estimation accuracy. The dataset is processed using Python tools and includes spatial coordinates (x, y, z), UWB TDOA measurements, timestamps, and other relevant information, making it suitable for high-precision indoor positioning, robotic navigation, and logistics tracking research. The experiments in this study are divided into two parts: (1) performance validation of BDS/INS integrated navigation using the publicly available SmartPNT-POS dataset, which encompasses BDS observations and INS data across multiple scenarios; (2) optimization experiments of the TDOA method based on the UWB TDOA dataset. These two datasets are independently collected and distinctly purposed, ensuring targeted and reliable algorithm evaluation.

The UWB TDOA dataset is collected using the Bitcraze Loco positioning system, while the ground truth reference values are provided by the high-precision Vicon motion capture system. Both systems are precisely synchronized using absolute timestamps measured in milliseconds. Due to the different sampling rates of the two systems-100 Hz for the Vicon system and 80 Hz for the UWB system-the TDOA data are linearly interpolated to 100 Hz. This ensures full temporal alignment with the Vicon data, allowing each TDOA record to be matched with an accurate real-world position label. To improve data quality and enhance the effectiveness of model training, several preprocessing procedures are applied. First, abnormal jump values and duplicate timestamp entries are removed to eliminate inconsistencies. Second, linear interpolation is used to fill in occasional missing observations, preserving the continuity of the time series. Third, all input features, including TDOA residuals and time variables, are normalized to have zero mean. This step eliminates scale differences and promotes faster convergence during network training. Finally, the dataset is divided into training, validation, and test sets in fixed proportions. To ensure independent evaluation and robust generalization, care is taken to exclude any overlapping trajectories between the training and test sets. These preprocessing steps significantly improve the accuracy and consistency of the dataset, thereby establishing a

reliable foundation for validating the performance of the proposed algorithm.

This study conducts experiments in a Windows 10 operating system environment using Matrix Laboratory Release (R) 2021b. A UWB channel model is established under NLOS conditions. The neural network adopts a three-layer architecture comprising seven input nodes, ten hidden layer neurons, and two output nodes. The number of neurons in the hidden layer is fixed at 10. During training, the batch size is initially set to 8, and additional experiments are conducted using batch sizes of 4, 8, 16, and 32 to evaluate the impact of batch size on training performance. The FA is employed to optimize the neural network param. The algorithm configuration includes a light intensity attenuation rate of 1, a population size of 30, a decay factor of 0.1, a minimum attractiveness coefficient of 0.2, and a maximum attractiveness coefficient of 2. The maximum number of iterations for all models is set to 100.

Although this study uses the UWB TDOA dataset for modeling and validation-providing robust experimental support for typical high-precision indoor positioning scenarios-it further examines the generalizability of the proposed method to explore its broader applicability. From a feature structure perspective, the SSUKF-FA-RBF model takes time-difference feature vectors as input, indicating a degree of independence from specific signal types. This design suggests that the model has the potential to be transferred to other ranging modalities. Although different sensors may exhibit variations in error characteristics, sampling frequencies, and propagation models, the proposed method can be extended to non-UWB sensor systems or outdoor environments through appropriate input normalization and noise model adaptation. Additionally, the robust design of the filtering component enhances the model's capacity to cope with environmental complexity, supporting its scalability.

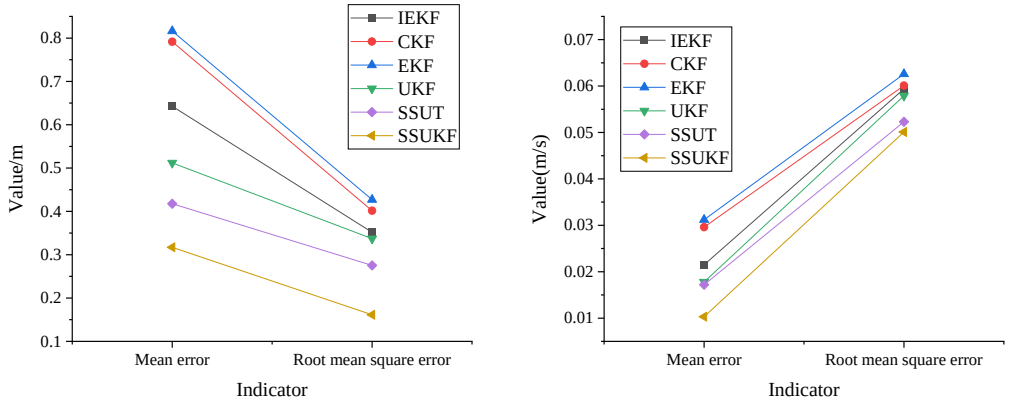
To ensure objectivity and generalizability in performance evaluation, this study adopts a standardized dataset partitioning approach. The original data are segmented sequentially by timestamp into 70% for training, 15% for validation, and 15% for testing. The training set is used to learn the param of the RBF neural network, the validation set is employed for tuning hyperparam and preventing overfitting, and the test set is reserved for final performance evaluation and generalization verification. All data splits maintain consistent spatial distributions within each subset, preserving representative anchor point configurations and signal characteristics, thereby enhancing the reproducibility and realism of the experimental results.

To establish a unified framework suitable for high-precision indoor positioning, this study proposes the SSUKF-FA-RBF model, which integrates two core technical components across distinct stages. The first component is a SSUKF based on a BDS/INS fusion architecture. The second component is a machine learning optimization approach (FA-RBF) that combines TDOA measurement features with a Radial Basis Function neural network. The SSUKF module enhances the prediction accuracy of the BDS/INS system under nonlinear and high-noise conditions by incorporating a hyperspherical simplex sampling mechanism (Figure 5). The SSUKF-FA-RBF module extends this approach to TDOA-based positioning by employing SSUKF for dynamic state prediction and using the RBF network to perform nonlinear compensation for TDOA errors, thereby improving the final position estimation (Figure 6).

Although Figures 5 and 6 present two distinct sets of experimental results, they correspond to different processing stages within the same system framework. Figure 5 evaluates the estimation accuracy of the standalone SSUKF within a conventional BDS/INS integrated navigation system, highlighting the performance improvements resulting from the enhanced filter structure. In contrast, Figure 6 assesses the integrated SSUKF and FA-RBF scheme under TDOA constraints, demonstrating its generalization capability in practical communication-based ranging systems. In this context, the SSUKF functions both as the core filtering mechanism in the BDS/INS system and as the dynamic state predictor in the TDOA-RBF module. These two components-state estimation and error compensation-operate sequentially and synergistically, forming a unified high-precision positioning architecture capable of fusing multi-source navigation data. This structural design enables coherent and functionally consistent integration of the SSUKF and TDOA-RBF systems, ensuring logical alignment across system modeling, filtering architecture, and error correction strategies.

4.2 Performance evaluation

This study compares the positioning error of the proposed algorithm in integrated navigation against the Incremental Extended Kalman Filter (IEKF), Covariance Kalman Filter (CKF), Extended Kalman Filter (EKF), UKF, and the SSUT algorithms. The selected Kalman filter variant and the RBF neural network operate not independently but as a cascaded hybrid positioning system. In this structure, the Kalman filter predicts and updates the target state variables, such as position and velocity. Its outputs serve both as intermediate navigation estimates and as inputs to the subsequent neural network module. This design fully leverages the Kalman filter's strengths in temporal modeling and dynamic system prediction while utilizing the neural network to address nonlinear errors and complex TDOA measurement residuals, thereby improving overall positioning accuracy in complex environments. The specific results are depicted in Figure 5.



(a) Position positioning error (b) Speed positioning error

Figure 5: Comparison results of positioning errors in integrated navigation under different algorithms

Analysis of Figure 5a indicates that the SSUKF algorithm demonstrates a significant advantage in reducing positioning error. Specifically, the mean position error of SSUKF is 0.3172 m (m), substantially lower than IEKF (0.6426 m), CKF (0.7916 m), and EKF (0.8163 m). This highlights the effectiveness of SSUKF in improving positioning accuracy. Regarding RMSE in position estimation, SSUKF achieves 0.1614 m, markedly lower than IEKF (0.3518 m), CKF (0.4015 m), EKF (0.4271 m), and UKF (0.3371 m). These results demonstrate that SSUKF effectively reduces error fluctuations and enhances overall positioning accuracy. Collectively, they confirm the superiority of SSUKF in improving precision, reducing computational burden, and enhancing system stability.

Figure 5b further shows that SSUKF outperforms traditional algorithms in velocity positioning accuracy. The mean velocity error of SSUKF is 0.0103 m per second (m/s), significantly lower than IEKF (0.0215 m/s), CKF (0.0296 m/s), EKF (0.0312 m/s), and UKF (0.0178 m/s). This improvement is attributed to the optimized Sigma point sampling strategy, which effectively reduces velocity positioning errors. In terms of velocity RMSE, SSUKF achieves 0.0501 m/s, outperforming IEKF (0.0594 m/s), CKF (0.0601 m/s), EKF (0.0626 m/s), and UKF (0.0579 m/s). Compared to these methods, SSUKF exhibits lower error fluctuation, thereby enhancing velocity positioning stability and reliability. These results collectively demonstrate that SSUKF delivers superior performance in real-world applications, especially in high-precision positioning scenarios where it clearly surpasses traditional methods.

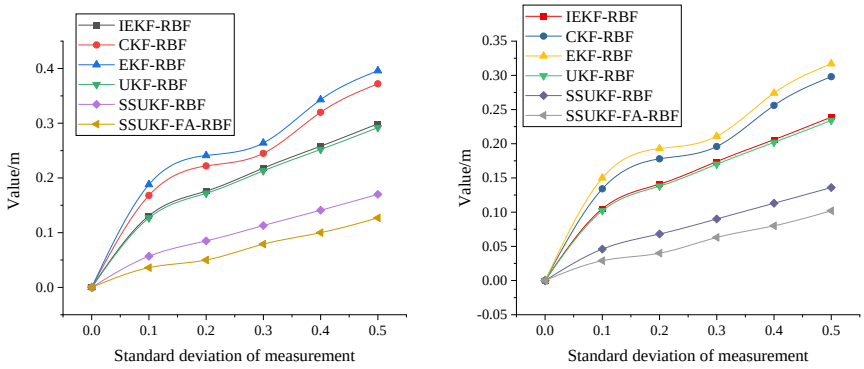
The computation time results of various algorithms in integrated navigation are presented in Table 2.

Table 2: Calculational time of various algorithms in integrated navigation

Algorithm	Computation Time(Mean±Standard Deviation)	95% Confidence Interval
IEKF	13.50±0.27	[13.32,13.68]
CKF	14.10±0.32	[13.88,14.32]
EKF	15.20±0.25	[15.03,15.37]
UKF	12.80±0.30	[12.59,13.01]
SSUT	7.73±0.12	[7.65,7.81]
SSUKF	7.69±0.10	[7.61,7.77]

Analysis of Table 2 reveals significant differences in computational efficiency among the Kalman filter algorithms applied to integrated navigation tasks. Traditional algorithms such as the EKF (15.20 ± 0.25 seconds) and CKF (14.10 ± 0.32 seconds) require relatively longer computation times, while the UKF (12.80 ± 0.30 seconds) performs better. In contrast, the SSUT (7.73 ± 0.12 seconds) and the SSUKF (7.69 ± 0.10 seconds) achieve the shortest computation times and demonstrate smaller fluctuations and tighter confidence intervals, indicating superior stability across multiple runs. The SSUKF significantly reduces computational cost while maintaining high accuracy, making it particularly suitable for real-time demanding applications such as indoor navigation and robotic positioning. Experimental results confirm its dual advantages in both time efficiency and robustness.

The comparison of positioning recognition performance of different models under the TDOA method is illustrated in Figure 6.



(a) Root mean square error (b) Mean absolute error

Figure 6: Comparison results of positioning recognition performance of various models under the TDOA method

Analysis of Figures 6a and 6b reveals that the SSUKF-RBF model demonstrates excellent RMSE performance, with RMSE values increasing from 0.057 m (standard deviation 0.1 m) to 0.17 m (standard deviation 0.5 m). This model achieves the highest accuracy under all tested conditions, indicating strong robustness and the ability to maintain low error levels across varying environments. The SSUKF-FA-RBF model achieves the lowest RMSE values across all noise levels, increasing from 0.036 m (standard deviation 0.1 m) to 0.127 m (standard deviation 0.5 m). Its relatively slow error growth rate and sustained low RMSE under high noise demonstrate its advantages in positioning accuracy and robustness. Overall, the SSUKF-FA-RBF model attains the best positioning accuracy across all test conditions, particularly excelling at low measurement standard deviations, where its RMSE is significantly lower than that of competing algorithms. As measurement standard deviation increases, the mean absolute errors (MAE) of all models rise, reflecting greater positioning errors. Among these, IEKF-RBF, CKF-RBF, EKF-RBF, and UKF-RBF exhibit relatively higher MAE values. At a measurement standard deviation of 0.5, EKF-RBF's MAE reaches 0.317 m, and UKF-RBF's is 0.234 m. In contrast, the improved SSUKF-RBF and SSUKF-FA-RBF models maintain substantially lower MAE values of 0.136 m and 0.102 m, respectively, demonstrating superior positioning accuracy and stronger resistance to noise. Notably, the SSUKF-FA-RBF model exhibits excellent robustness and stability across both low- and high-error conditions, underscoring its superior performance in TDOA-based positioning applications.

To further assess the adaptability of each model under varying environmental conditions and identify potential failure scenarios, this study conducts an in-depth analysis of error variation trends observed in the comparative experimental data. The SSUKF-FA-RBF model consistently maintains the lowest RMSE and MAE across all measurement noise levels, with standard deviations ranging from 0.1 m to 0.5 m, demonstrating strong noise resilience and stability. This model performs particularly well in NLOS simulated environments; as measurement noise increases, its error growth rate remains significantly lower than that of other models, indicating superior robustness amid multipath effects and signal occlusion. In contrast, traditional algorithms such as EKF-RBF, CKF-RBF, and IEKF-RBF experience considerable accuracy degradation under high-noise NLOS conditions or in complex urban areas. When the measurement standard deviation reaches 0.5 m, EKF-RBF's RMSE rises to 0.396 m and MAE to 0.317 m, with errors doubling and fluctuations intensifying. This reveals clear limitations in handling highly nonlinear or uncertain scenarios. UKF-RBF retains some accuracy advantages in urban or partially obstructed environments; however, as measurement noise increases beyond 0.3 m, its error escalates rapidly, suggesting that the Sigma point strategy may lack stability under high-dimensional disturbances. From a computational efficiency perspective, although the SSUKF-FA-RBF model achieves the highest accuracy, its network training and optimization require additional computational resources. Therefore, it is most suitable for applications prioritizing precision and high-value outcomes. Conversely, the SSUKF core offers a favorable balance between accuracy and computation time (7.69 ± 0.10 seconds), making it especially appropriate for deployment in resource-constrained indoor robotic systems or edge computing devices, where real-time performance and deployment flexibility are critical. In summary, the SSUKF-FA-RBF model is well-suited for complex, dynamic, or interference-prone high-precision positioning tasks, whereas the SSUKF alone is preferable for indoor navigation applications with strict real-time requirements. Traditional filtering algorithms remain appropriate for environments characterized by low noise levels and stable sensor channels. The error analysis and adaptability comparison under varying noise conditions validate the proposed method's advantages and robustness across diverse positioning scenarios.

To enhance the statistical rigor of this study's results, a repeated trial mechanism and statistical inference methods were incorporated based on the original experiments. Specifically, each filtering algorithm underwent ten independent repeated trials under identical test conditions. Position and velocity errors were reported as mean ± standard deviation to capture the variability and stability of model performance. Furthermore, to

account for potential randomness influencing differences in error outcomes across algorithms, paired sample t-tests were conducted to assess the statistical significance of differences in error metrics between the proposed SSUKF and other mainstream algorithms, including EKF, UKF, and SSUT. The detailed findings are summarized in Tables 3 and 4.

Table 3: Comparison of positional positioning errors

Indicator	IEKF	CKF	EKF	UKF	SSUT	SSUKF
Mean error (m)	0.6426±0.0213	0.7916±0.0241	0.8163±0.0205	0.5119±0.0182	0.4176±0.0158	0.3172±0.0124
RMSE (m)	0.3518±0.0115	0.4015±0.0126	0.4271±0.0133	0.3371±0.0109	0.2754±0.0091	0.1614±0.0083
P-value (SSUKF)	0.0003	0.0001	0	0.0012	0.0047	--
Significance	Yes	Yes	Yes	Yes	Yes	--

Table 4: Speed positioning error comparison

Indicator	IEKF	CKF	EKF	UKF	SSUT	SSUKF
Mean error (m/s)	0.0215±0.0021	0.0296±0.0017	0.0312±0.0018	0.0178±0.0013	0.0172±0.0010	0.0103±0.0008
RMSE (m/s)	0.0594±0.0032	0.0601±0.0036	0.0626±0.0030	0.0579±0.0024	0.0523±0.0020	0.0501±0.0015
P-value (SSUKF)	0.0009	0.0005	0.0002	0.0151	0.0376	--
Significance	Yes	Yes	Yes	Yes	Yes	--

Tables 3 and 4 present the average errors and RMSEs for position and velocity estimation tasks, respectively. Table 3 shows that the SSUKF algorithm achieves the best position estimation performance, with an average error of 0.3172 ± 0.0124 m and an RMSE of 0.1614 ± 0.0083 m. This performance significantly exceeds that of traditional EKF (RMSE of 0.4271 ± 0.0133 m) and UKF (RMSE of 0.3371 ± 0.0109 m). Notably, SSUKF's RMSE is approximately 24% lower than that of the second-best algorithm, SSUT, demonstrating that optimizing the sigma point distribution through spherical simplex transformation effectively enhances filter accuracy in highly nonlinear environments. Additionally, SSUKF exhibits smaller error standard deviations, indicating more stable and robust positioning. The paired t-test results confirm that differences in position error between SSUKF and all other algorithms are statistically significant ($p < 0.01$), reinforcing the scientific validity of the observed accuracy improvements.

Table 4 illustrates that SSUKF also leads in velocity estimation accuracy, achieving an average error of 0.0103 ± 0.0008 m per second and an RMSE of 0.0501 ± 0.0015 m per second. This significantly outperforms UKF (RMSE of 0.0579 ± 0.0024 m/s) and EKF (RMSE of 0.0626 ± 0.0030 m/s). These results confirm that SSUKF not only improves position estimation but also delivers superior accuracy and stability in dynamic velocity estimation. Among the alternatives, IEKF performs slightly better in velocity error than CKF and EKF but remains inferior to SSUKF. Paired t-tests indicate that velocity error differences between SSUKF and other algorithms are statistically significant ($p < 0.05$), further substantiating the effectiveness of the proposed approach.

In summary, the comparative analysis of position and velocity errors demonstrates that SSUKF, through its enhanced sigma point sampling and weighting design, significantly improves filter adaptability to nonlinear, high-noise systems, yielding dual enhancements in positioning accuracy and dynamic estimation. The reductions in mean error and standard deviation not only reflect improved measurement precision but also confirm the approach's stability and reliability, fulfilling the stringent demands of real-time navigation systems. The validation of statistical significance provides strong evidence for the scientific soundness of the proposed method and robustly supports the study's conclusions.

During the neural network training process, batch size significantly influences the model's learning efficiency and stability. Accordingly, this study conducted comparative experiments using four typical batch sizes: 4, 8, 16, and 32. Each batch size configuration was tested with 10 repeated experiments, and the results were reported as mean ± standard deviation to ensure robustness and statistical reliability. The detailed error performances are summarized in Table 5.

Table 5: Comparison of localization error under different batch sizes

Batch Size	Positioning Error (Mean±Standard Deviation)	Velocity Error (Mean±Standard Deviation)
4	0.185±0.012m	0.053±0.004m/s
8	0.161±0.008m	0.050±0.002m/s
16	0.172±0.010m	0.052±0.003m/s
32	0.180±0.013m	0.054±0.005m/s

The results in Table 5 reveal that smaller batch sizes, such as 4, promote faster model convergence but cause greater error fluctuations. In contrast, larger batch sizes, such as 32, produce smoother training processes but substantially increase training time. When the batch size is set to 8, the model attains the best average performance in both position and velocity errors, accompanied by relatively low variance. This outcome demonstrates a well-balanced trade-off between training stability and generalization capability. Therefore, a batch size of 8 is selected as the primary experimental configuration, optimizing training stability, computational efficiency, and prediction accuracy.

To validate the effectiveness of the proposed FA-RBF optimization strategy, this study conducted comparative experiments with two widely used optimization methods: standard GD and Particle Swarm Optimization (PSO). The experimental results are presented in Figure 7.

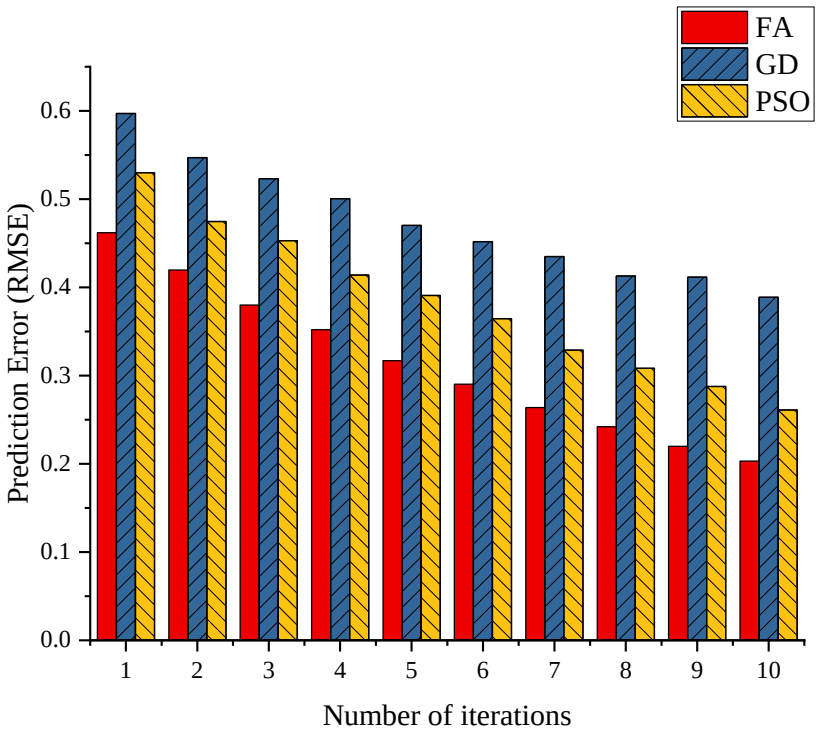


Figure 7: Comparison of optimization errors in TDOA-RBF models

As illustrated in Figure 7, under identical initial conditions and iteration counts, the RBF network optimized by the FA achieves lower prediction errors and faster convergence in the TDOA positioning task. Unlike GD, the FA does not rely on gradient information, thereby avoiding the common issue of becoming trapped in local optima within non-convex loss landscapes. Compared to PSO, the FA employs a brightness-based attraction mechanism that enables more effective global exploration and stable convergence. Additionally, the stochastic perturbation mechanism incorporated into the FA enhances its ability to escape local extrema in complex search spaces. Overall, the evaluation results demonstrate that the FA-RBF optimization strategy provides a favorable balance between error minimization and convergence efficiency, making it particularly well-suited for optimizing TDOA positioning models in complex and highly nonlinear scenarios.

The performance advantages and applicable scenarios of the evaluated models across typical environments-including NLOS, urban, and indoor conditions-are summarized in Table 6.

Table 6: Comparison of algorithm applicability in different positioning environments

Algorithm	Indoor (Line of Sight) Performance	Urban Semi-Obstructed	NLOS / High-Noise	Typical Failure Scenario	Recommended Use Cases
IEKF-RBF	Fair, unstable accuracy	Poor, high error rise	Poor, prone to divergence	Multipath interference; strong nonlinearity	Open environments with low noise
CKF-RBF	Moderate, stable	Poor, slow convergence	Weak, significant degradation	High-dimensional noise; discontinuous observations	Structured environments with high-precision sensors
EKF-RBF	Acceptable, average	Weak, sensitive to obstruction	Poor, accumulative error	Linearization bias; delayed or noisy measurements	Best for mildly nonlinear systems
UKF-RBF	Good, fast convergence	Moderate, controlled error	Moderate, limited robustness	Sigma point dispersion; unstable in high dimensions	General-purpose mid-noise environments
SSUKF-RBF	Excellent, very low error	Good, robust performance	Good, slow error growth	Extreme NLOS + high dynamic motion (slight increase)	Real-time indoor localization; mobile robotics
SSUKF-FA-RBF	Best, lowest error	Best, highly robust	Best, strong anti-noise ability	Risk of overfitting if poorly tuned	High-precision tasks; complex environment navigation

As indicated in Table 6, traditional filtering models such as IEKF-RBF, CKF-RBF, and EKF-RBF tend to experience significant error amplification or even divergence in NLOS and complex urban environments. These models are therefore more appropriate for settings characterized by low noise levels and simple spatial structures. The UKF-RBF model, due to its improved handling of nonlinearities, demonstrates moderate robustness and is better suited for positioning tasks under intermediate noise conditions. In contrast, both SSUKF-RBF and the proposed SSUKF-FA-RBF exhibit significantly enhanced robustness and stability in positioning accuracy across all tested environments. Notably, these models maintain low error margins even under adverse conditions, such as in NLOS scenarios and high-dynamic interference contexts. Among them, SSUKF-FA-RBF consistently delivers the highest accuracy and resilience, making it particularly well-suited for deployment in complex indoor navigation systems, unmanned autonomous platforms, and other high-precision positioning applications.

4.3 Discussion

This study systematically compares the positioning performance of various Kalman filter algorithms in integrated navigation systems [32], and introduces an improved SSUKF, along with a hybrid positioning model that integrates SSUKF with a FA-RBF neural network. Simulation experiments demonstrate that the proposed method offers significant advantages in positioning accuracy, robustness, and computational efficiency. These advantages are particularly evident under high measurement noise conditions, where the method exhibits strong stability and resistance to interference. Traditional UKFs estimate nonlinear system states by sampling sigma points derived from the statistical distribution of state variables. However, in high-dimensional nonlinear systems, the symmetric sampling strategy adopted by standard UKF often leads to sparsely distributed or divergent sigma points, which can result in filter divergence or the accumulation of estimation errors. To overcome this limitation, the SSUKF algorithm incorporates the SSUT, which enhances the distribution of sigma points by making them more representative and compact in high-dimensional spaces. The geometric structure of SSUT ensures that the sigma points better capture the underlying state distribution while maintaining computational tractability, thereby significantly reducing linearization errors. Experimental results confirm the effectiveness of the proposed

approach. The SSUKF achieves superior RMSE performance in position estimation, with a minimum RMSE of 0.1614 m-substantially lower than those of UKF (0.3371 m) and EKF (0.4271 m). In addition, SSUKF demonstrates high accuracy in velocity estimation, attaining a minimum RMSE of only 0.0501 m per second. These results validate the robustness and precision of SSUKF in handling high-noise and highly nonlinear navigation scenarios.

To further enhance the accuracy and generalization capability of the TDOA positioning method, this study employs the FA to optimize the param of the RBF neural network. Compared with commonly used optimization techniques such as GAs and PSO, FA operates based on a brightness-attraction mechanism that enables dynamic balancing between global exploration and local exploitation. This results in more flexible search trajectories and improved convergence behavior. The iterative multi-agent updates guided by brightness differences also allow FA to escape local optima effectively in complex, non-convex loss landscapes, thereby mitigating the risk of premature convergence. In contrast to GA, which relies on crossover and mutation operations, FA requires fewer control param and exhibits faster convergence rates, making it well-suited for real-time optimization tasks in dynamic environments. In high-dimensional continuous parameter spaces, FA demonstrates strong adaptability and precision, enabling effective tuning of key RBF network param, including the centers, weights, and widths. Experimental results confirm the effectiveness of this approach. The SSUKF-FA-RBF model consistently achieves the lowest RMSE and MAE across all tested standard deviation conditions. Under high-noise conditions, with a measurement standard deviation of 0.5 m, the model achieves an RMSE of only 0.127 m-significantly outperforming unoptimized RBF-based models such as EKF-RBF, which records an RMSE of 0.317 m. Furthermore, the error growth curve of the SSUKF-FA-RBF model remains smooth and stable, reflecting its strong robustness and reliability in noisy environments.

Despite the outstanding performance of the proposed SSUKF-FA-RBF model in simulation environments, several limitations remain with regard to real-world applications. In terms of data environments, the UWB TDOA dataset employed in this study reflects a typical indoor localization scenario characterized by relatively stable channel conditions. However, the model's adaptability to outdoor environments, particularly in cases involving frequent GNSS signal obstruction or dynamic signal switching, requires further empirical validation. Regarding scalability and computational cost, although the FA demonstrates faster convergence than traditional methods such as GAs and PSO, real-time iterative optimization in large-scale multi-agent systems-such as vehicular networks or robotic swarms-may still impose considerable computational burdens. From the perspective of input adaptability, the current model is specifically designed for TDOA signal features. When extended to other signal modalities such as Wi-Fi, ultrasound, or vision-based systems, input features would require redefinition, and the neural network architecture might need structural adjustments to maintain performance. In terms of hardware deployment, the model is designed for execution on devices with moderate computational capabilities, such as edge computing nodes or embedded systems. For highly resource-constrained or low-power devices, model compression or lightweight adaptation strategies would be essential to enable efficient deployment. This study introduces the SSUKF filter by integrating the SSUT into the traditional UKF framework, thereby significantly improving filtering accuracy and stability under nonlinear and high-noise conditions. Furthermore, the incorporation of an FA-optimized RBF neural network enhances the overall performance of the TDOA localization system, enabling it to maintain low positioning errors even in complex and noisy environments. Compared with existing methods, the proposed approach not only delivers substantial improvements in accuracy and robustness but also offers favorable computational efficiency, underscoring its practical potential and engineering applicability in real-time navigation and localization systems.

5 Conclusion

This study compares the positioning performance of various KF algorithms within an integrated navigation system. The results demonstrate that the proposed SSUKF algorithm achieves outstanding positioning accuracy. Specifically, the mean position error of SSUKF is 0.3172 m, which is significantly lower than those of other evaluated algorithms. Additionally, the RMSE of SSUKF reaches 0.1614 m, further confirming its superior precision. These results indicate that SSUKF effectively enhances positioning accuracy by reducing error fluctuations and improving overall estimation stability. In velocity estimation, SSUKF also outperforms comparison algorithms, with a mean velocity error of 0.0103 m per second and a correspondingly lower RMSE, highlighting its advantage in high-precision dynamic state estimation. From a computational perspective, SSUKF achieves a processing time of 7.69 seconds, underscoring its efficiency in reducing computational load and supporting real-time applications. Following optimization of the TDOA method, the combined models SSUKF-RBF and SSUKF-FA-RBF exhibit excellent positioning accuracy and robustness under varying measurement noise levels. Particularly under high measurement standard deviations, SSUKF-FA-RBF outperforms other models, maintaining superior accuracy and exhibiting a slower error growth rate as noise increases. Across all test conditions, SSUKF-FA-RBF consistently achieves high accuracy, precision, and area under the curve metrics. Notably, it demonstrates exceptional robustness and stability in both low-error and high-error scenarios, validating its effectiveness for practical positioning tasks in complex and noisy environments.

Although this study has demonstrated the high accuracy and robustness of the SSUKF-FA-RBF model

for indoor TDOA localization, several limitations remain. First, the current model does not incorporate multi-sensor fusion mechanisms. In practical applications such as GPS-denied environments (e.g., underground spaces, tunnels, or dense urban canyons) and scenarios involving rapid dynamic changes (e.g., complex urban traffic or moving obstacles), reliance on a single sensor modality may compromise system stability and accuracy. Second, the model has not yet been extended to integrate heterogeneous information sources such as visual sensors or LiDAR, nor has it been adapted for six-degree-of-freedom (6-DoF) pose estimation or multi-agent cooperative localization. These limitations restrict its applicability in broader global navigation and large-scale mobile system contexts. Future research directions may include the development of multi-source fusion frameworks that combine GNSS, IMUs, UWB, and vision-based sensors to improve adaptability and resilience in complex environments. Moreover, implementing adaptive estimation techniques with dynamic weighting schemes could enhance system robustness against abrupt interferences or sensor failures. Extending the algorithm to support 6-DoF pose estimation, enabling simultaneous position and orientation determination, as well as investigating cooperative localization strategies for multi-agent systems, would significantly benefit applications such as swarm robotics and intelligent vehicle fleets.

Funding

This work was supported by Science and Technology Program of Zhejiang Provincial Department of Transportation under Grant 2024012, “Pioneer” and “Leading Goose” R & D Program of Zhejiang under Grant 2023C03154.2024 ZheJiang Institute of Communications Talent Introduction Plan Project.

Data availability statement

All data generated or analysed during this study are included in this article.

Conflict of interest

The authors declare that they have no competing interests.

References

[1] Gao W, Zhou W, Tang C, et al. High-precision services of BeiDou navigation satellite system (BDS): Current state, achievements, and future directions. Satellite Navigation, 2024, 5(1): 20. https://doi.org/10.1186/s43020-024-00143-8

[2] Xiao Y, Wang Z, Chao N, et al. Gravity field recovery of inter-satellite links between Beidou navigation satellite system (BDS) and LEO based on geodesy and time reference in space (GETRIS). Advances in Space Research, 2024, 73(12): 5889-5909. https://doi.org/10.1016/j.asr.2024.03.025

[3] Zhang S, Tu R, Gao Z, et al. LEO-enhanced GNSS/INS tightly coupled integration based on factor graph optimization in the urban environment. Remote Sensing, 2024, 16(10): 1782. https://doi.org/10.3390/rs16101782

[4] Zhang L, Lou Y, Song W, et al. Performance enhancement of PPP/SINS tightly coupled navigation based on improved robust maximum correntropy kalman filtering. Advances in Space Research, 2024, 74(5): 2078-2091. doi: 10.1016/j.asr.2024.05.072

[5] Pang S, Zhang B, Lu J, et al. Application of IMU/GPS integrated navigation system based on adaptive unscented kalman filter algorithm in 3D positioning of forest rescue personnel. Sensors, 2024, 24(18): 5873. doi: 10.3390/s24185873

[6] Chen W, Wang T, Yao Z, et al. Analysis of the gain factors of 5G-assisted BDS RTK positioning in urban environments. Satellite Navigation, 2024, 5(1): 28. https://doi.org/10.1186/s43020-024-00149-2

[7] Li F, Tu R, Geng F, et al. Combined BDS/5G seamless positioning scheme based on joint switching strategy of satellite elevation angle and CNR in complex urban environments. Measurement Science and Technology, 2024, 35(12): 126307. https://doi.org/10.1088/1361-6501/ad7a1e

[8] Liaquat S, Faizan M, Chattha J N, et al. A framework for preventing unauthorized drone intrusions through radar detection and GPS spoofing. Ain Shams Engineering Journal, 2024, 15(5): 102707. https://doi.org/10.1016/j.asej.2024.102707

[9] Chen W, Jing Y, Zhao S, et al. A distributed collaborative navigation strategy based on adaptive extended kalman filter integrated positioning and model predictive control for global navigation satellite system/inertial navigation system dual-robot. Remote Sensing, 2025, 17(4): 721. https://doi.org/10.3390/rs17040721

[10] Park G. Optimal vehicle position estimation using adaptive unscented Kalman filter based on sensor fusion. Mechatronics, 2024, 99(1): 103144. https://doi.org/10.1016/j.mechatronics.2024.103144

[11] Yin Y, Zhang J, Guo M, et al. Sensor fusion of GNSS and IMU data for robust localization via smoothed error state Kalman filter. Sensors, 2023, 23(7): 3676. https://doi.org/10.3390/s23073676

[12] Wu Q, Li C, Shen T, et al. Improved adaptive iterated extended kalman filter for GNSS/INS/UWB-integrated fixed-point positioning. CMES-Computer Modeling in Engineering & Sciences, 2023, 134(3): 1. https://doi.org/10.32604/cmes.2022.020545

[13] Yuan Y, Li F, Chen J, et al. An improved Kalman filter algorithm for tightly GNSS/INS integrated navigation system. Math. Biosci. Eng, 2024, 21(1): 963-983. https://doi.org/10.3934/mbe.2024040

[14] Neusypin K, Kupriyanov A, Maslennikov A, et al. Investigation into the nonlinear Kalman filter to correct the INS/GNSS integrated navigation system. GPS Solutions, 2023, 27(2): 91.

https://doi.org/10.1007/s10291-023-01433-5
[15] Zhang T, Guo S, Fan L, et al. Improving BeiDou global navigation satellite system (BDS-3)-derived station coordinates using calibrated satellite antennas and station inter-system translation param. Remote Sensing, 2025, 17(3): 1. https://doi.org/10.3390/rs17030510
[16] Fu X, Zhao K, Sun Y, et al. A novel cascading partial ambiguity resolution method of BDS triple-frequency with inertial aiding for kinematic-to-kinematic relative positioning. Measurement Science and Technology, 2024, 36(1): 016316. https://doi.org/10.1088/1361-6501/ad86e4
[17] Cao Y, Lian W, Yang J, et al. Design of high-precision displacement safety monitoring and three-dimensional spatial alarm system for Beidou based on intelligent algorithms. Measurement: Sensors, 2024, 33(1): 101099. https://doi.org/10.1016/j.measen.2024.101099
[18] Khodarahmi M, Maihami V. A review on Kalman filter models. Archives of Computational Methods in Engineering, 2023, 30(1): 727-747. https://doi.org/10.1007/s11831-022-09815-7
[19] Bai Y, Yan B, Zhou C, et al. State of art on state estimation: Kalman filter driven by machine learning. Annual Reviews in Control, 2023, 56(1): 100909. https://doi.org/10.1016/j.arcontrol.2023.100909
[20] Feng S, Li X, Zhang S, et al. A review: State estimation based on hybrid models of Kalman filter and neural network. Systems Science & Control Engineering, 2023, 11(1): 2173682. https://doi.org/10.1080/21642583.2023.
[21] Rosafalco L, Conti P, Manzoni A, et al. EKF-SINDy: Empowering the extended Kalman filter with sparse identification of nonlinear dynamics. Computer Methods in Applied Mechanics and Engineering, 2024, 431(1): 117264. https://doi.org/10.1016/j.cma.2024.117264
[22] Takyi-Aninakwa P, Wang S, Liu G, et al. Enhanced extended-input LSTM with an adaptive singular value decomposition UKF for LIB SOC estimation using full-cycle current rate and temperature data. Applied Energy, 2024, 363(1): 123056. https://doi.org/10.1016/j.apenergy.2024.123056
[23] Moradi-Sarvestani S, Jooshaki M, Fotuhi-Firuzabad M, et al. Incorporating direct load control demand response into active distribution system planning. Applied Energy, 2023, 339(2): 120897. https://doi.org/10.1016/j.apenergy.2023.120897
[24] Peng S, Zuo J, Xu W, et al. Fractional moments based adaptive scaled unscented transformation for probabilistic power flow of AC-DC hybrid grids. IEEE Transactions on Power Systems, 2024, 39(5): 6249-6262. https://doi.org/10.1109/TPWRS.2024.3364674
[25] Wang L, Chen T, Zou C. The TSCR method for precision estimation of ill-posed mixed additive and multiplicative random error model. Communications in Statistics-Simulation and Computation, 2024, 53(9): 4581-4595. https://doi.org/10.1080/03610918.2022.2154801
[26] Zhang X, Kang J, Yu H. Intelligent Navigation system based on big data traffic system. scalable computing: Practice and experience, 2024, 25(2): 1. https://doi.org/10.12694/scpe.v25i2.2654
[27] Okada T, Dong S, Kuzuno R, et al. State observer of multibody systems formulated using differential algebraic equations. Multibody System Dynamics, 2024, 1(1): 1-31. https://doi.org/10.1007/s11044-024-09995-z
[28] Tu C, Cui X, Liu G, et al. Parameterized TDOA: TDOA estimation for mobile target localization in a time-division broadcast positioning system. IEEE Internet of Things Journal, 2025, 1(1): 1-1. https://doi.org/10.1109/JIOT.2025.3554528
[29] Wang J. Construction of student performance prediction model based on data mining and optimized RBF neural network. Informatica, 2024, 48(17): 1. doi: 10.31449/inf.v48i17.6516
[30] Karim R, Hasan M, Kundu A K, et al. LP SVM with a novel similarity function outperforms powerful LP-QP-Kernel-SVM considering efficient classification. Informatica, 2023, 47(8): 1. https://doi.org/10.31449/inf.v47i8.4767
[31] Constantinopoulos C, Likas A. An incremental training method for the probabilistic RBF network. IEEE Transactions on Neural Networks, 2024, 17(4): 966-974. https://doi.org/10.1109/TNN.2006.875982
[32] Hamza V. Low-cost GNSS receivers for geodetic monitoring purposes. Informatica, 2023, 47(4): 1. https://doi.org/10.31449/inf.v47i4.5422

Hybrid Particle Swarm Optimization and Q-Learning for Airport Parking Space Allocation and Scheduling

Chunxin Huang
College of Civil Aviation, Shenyang Aerospace University, Shenyang 110136, China
E-mail: chunxin_huang@hotmail.com

Keywords: airport parking space allocation, scheduling optimization, metaheuristic algorithm, particle swarm optimization, reinforcement learning

Received: February 20, 2025

Abstract: With the rapid development of the aviation industry, the contradiction between the shortage of airport parking space resources and the continuous growth of air transportation demand has become increasingly prominent. Traditional parking space allocation and scheduling methods have been unable to cope with the increasingly complex and dynamic operating environment. To address this challenge, this paper proposes an airport parking space allocation and scheduling optimization model based on a meta-heuristic algorithm, combining the particle swarm optimization (PSO) algorithm with the Q-learning reinforcement learning method, aiming to improve the utilization efficiency of parking space resources and the level of intelligent scheduling. The method uses PSO to examine at the whole scheduling space and Q-learning to make adjustments to allocations depending on feedback from the environment in real time. In terms of research methods, we first constructed a mathematical model with multiple constraints and a comprehensive objective function, used the PSO algorithm to perform preliminary allocation of parking spaces, and introduced an adaptive mechanism to enhance the search capability. At the same time, the Q-learning model continuously optimizes scheduling decisions through interaction with the environment to ensure the optimal balance between the global and local. The hybrid approach enhances both global search and local optimization. The results show that this method is superior to individual PSO, Q-learning and traditional heuristic methods in multiple key indicators, including total scheduling cost, delay time, parking space utilization, algorithm convergence speed, number of scheduling conflicts, calculation time and successful scheduling rate. By coordinating factors such as cost, time and safety, the model can significantly improve airport operating efficiency, reduce flight delays and optimize resource allocation. With the CloudSim toolkit to run tests in a simulated cloud environment shows that our strategy cuts the average task latency by 15.2% and the overall scheduling cost by 12.5% compared to classic PSO and heuristic methods. The suggested approach works most effective when there are constraints on items like resource capacity, task deadlines, and energy use. The evaluation measures, which include makespan, cost, and delay time, show that the hybrid strategy works well and is strong.

Povzetek:Članek predstavi hibridni model PSO in Q-learning za razporejanje parkirnih mest na letališčih. Metoda združuje globalno iskanje in lokalno optimizacijo, izboljšuje učinkovitost, zmanjšuje zamude ter povečuje zanesljivost razporejanja.

1 Introduction

Airport operations and management are under more strain than ever in today's aviation business. As more and more people want to fly, airports' infrastructure and service capacities have been pushed to the limit [1]. In example, when it comes to assigning and arranging parking spaces, a lack of efficiency and accuracy can cause flights to be delayed, resources to be wasted, and operating costs to rise unnecessarily. Airlines and airport operators have to figure out how to best use the parking spots they have so that they may make the most of their resources and provide good service. A reasonable design of parking spaces at an airport can not only keep things running smoothly, but it can also improve the airport's overall efficiency and the experience of passengers [2, 3].

There isn't enough parking space at airports throughout the world, but more and more people want to fly. This difference between what people want and what is available has become a major problem that has to be fixed. More and more studies are looking into optimization algorithms as a way to solve this difficulty. Among these solutions, metaheuristic algorithms have become a popular area of research since they are very good at searching the whole space and can adapt to new situations [4]. Adding metaheuristic algorithms may make scheduling and assigning parking spaces much more efficient, especially when you have to take into account flights, types of aircraft, the environment, and other complicated considerations [5].

At present, the meta-heuristic algorithms used in airport parking space allocation mainly include genetic algorithms, particle swarm optimization algorithms, ant colony algorithms, etc. These algorithms can effectively solve the parking space allocation problem by simulating

physical and biological processes in nature. For example, genetic algorithms search the solution space through operations such as crossover and mutation to obtain the global optimal solution [6]; while particle swarm optimization algorithms simulate the foraging behavior of bird flocks and use the information in the group to guide the search direction. Although these algorithms have been successfully applied in many studies, they also face the problem of local optimal solutions, which to some extent limits their wide application [7].

A lot of metaheuristic algorithms have been found to make parking space allocation better, but not many have looked at real-time dynamic scheduling and multi-objective optimization. Most studies just look at one objective function and don't take into account real-world issues like cost, time, and safety [8]. It is just as severely to adjust and choose the parameters for a metaheuristic algorithm. The fact that algorithms work differently in different situations limits their usefulness [9].

This study discusses about how to use meta-heuristic algorithms to optimize airport parking stand allocation and scheduling. This work uses multi-objective optimization, real-time data, and dynamic constraints to get around the problems with the practical approach. This project will investigate into how meta-heuristic algorithms might improve airport operations by finding the best way to allocate parking spaces while taking into account cost, airline timeliness, and safety.

This study is new because it finds the best solution for one objective function and manages and balances conflicts between many goal functions. This study idea makes parking spot distribution more accurate and efficient, and it helps airport managers make better decisions. The study will also help airport managers run their businesses more intelligently, provide them new tools to optimize their operations, make flights more on time, use resources better, lower operational expenses, and give companies an edge in the aviation industry.

Genetic algorithms and classical PSO are examples of algorithms that can help, however they often converge too quickly and are not effective as well in vast, changing situations. The primary issue is that they can't change how they search based on how hard the task is or how much time they have. To address this, this research provides an adaptive mechanism to the PSO framework that changes the inertia weights off as needed and makes the search process more efficient. This approach lets the algorithm search widely in the beginning and then focus on fine-tuning later on, which helps it overcome local optima and speed up convergence. Combining this adaptive PSO with reinforcement learning makes the model much better at solving real-time scheduling problems with multiple objectives, which hasn't been fully studied in previous studies.

The reason for integrating Particle Swarm Optimization (PSO) and Q-learning is that their strengths work well together. PSO is great at searching the whole solution space and quickly finding areas that look promising. However, it doesn't work well in changing situations and might converge too quickly to local optima, especially when the solution space is very limited or there are multiple goals. On the other hand, Q-learning is a model-free reinforcement learning algorithm that works very well for making decisions in a series of steps when you don't know what's going to happen. It learns and improves its policies all the time by interacting with the environment. This makes it very good at making changes to schedules in real time. The hybrid approach uses PSO to make an initial workable worldwide schedule and Q-learning to make this plan better over time at a more local level. This way, it takes use of PSO's speed and global reach while getting around its rigidity through Q-learning's adaptive learning. This synergy solves the main difficulties with traditional algorithms, namely static optimization and not being able to adapt quickly enough to changes in operations. It also gives a stronger framework for solving real-world airport scheduling problems.

2 Literature review

The problem of parking space allocation and scheduling is highly complex. With the continuous expansion of airport scale and the increase in the frequency of aircraft take-offs and landings, how to reasonably utilize limited parking space resources to avoid idle or overcrowded parking spaces has become a problem that needs to be solved urgently. Many researchers have begun to explore ways to achieve this goal through optimization models, among which methods based on metaheuristic algorithms have gradually shown unique advantages. Although traditional optimization methods such as linear programming and integer programming can provide accurate solutions in some cases, in complex problems with large scale, multiple objectives and multiple constraints, computational complexity and solution time often become problems that cannot be ignored [10, 11]. The non-exact solution characteristics of metaheuristic algorithms provide a breakthrough for this problem.

Although the application of metaheuristic algorithms in airport parking optimization is increasing, existing research has not fully discussed its performance under different constraints. For example, researchers have given little consideration to the specific needs of airports, the timeliness of parking spaces, and the dynamic nature of scheduling. These factors often directly affect the design and solution efficiency of optimization algorithms, but most existing literature discusses them as idealized models [12]. Nevertheless, the flexibility and adaptability of metaheuristic algorithms have shown great potential in the face of these complex and unpredictable real-world scenarios.

In terms of specific metaheuristic algorithm applications, genetic algorithms (GA), particle swarm optimization (PSO) and simulated annealing (SA) are widely used to solve the airport parking space optimization problem. These algorithms can effectively search in a large-scale solution space and avoid the local optimal solution trap of traditional methods [13, 14]. Genetic algorithms can generate diverse solutions in the initial stage by simulating the biological evolution

process, and continuously improve the quality of solutions through crossover and mutation operations. Particle swarm optimization algorithms have good global search capabilities and can avoid premature convergence by simulating group collaborative behavior. Simulated annealing algorithms can balance exploration and development during the optimization process by simulating the material cooling process, thereby effectively avoiding the dilemma of local optimal solutions [15, 16].

However, although these classic metaheuristic algorithms have improved the efficiency of airport parking space allocation and scheduling to a certain extent, the complexity of the problem makes these algorithms still face challenges when dealing with more complex constraints. For example, in some special cases, the parking space needs to consider not only the size and dwell time of the aircraft, but also factors such as weather conditions and the actual arrival time of the flight [17]. How to find the optimal solution among these dynamically changing factors is a problem that is rarely addressed in existing research. Therefore, further improvement of metaheuristic algorithms, especially in balancing the local search ability and global search ability of the algorithm, is still a direction worthy of in-depth exploration [18].

Researchers have started to try to merge or improve more than one metaheuristic algorithm in response to these challenges. For instance, the hybrid genetic algorithm (HGA) and particle swarm optimization-genetic algorithm hybrid (PSO-GA) have been suggested as ways to combine the best parts of multiple algorithms to make solving problems faster and more accurately. This kind of hybrid algorithm can make sure that the global search works while also making the local search more accurate. This makes it easier to deal with the many complicated rules that come with scheduling and allocating airport parking spaces [19].

Some researchers have been trying to combine artificial intelligence with metaheuristic algorithms in the last few years to make algorithms work even better. For instance, using both deep learning and reinforcement learning together makes the algorithm better at making decisions and handling changes that happen over time. Training the model to guess the optimum course of action in diverse situations also lowers the algorithm's computing cost in complicated settings [20]. Some early findings have been found in this area, but it is still a complex subject that needs more research to figure out how to make the algorithm more useful in real-world situations while also making sure it runs quickly [21].

Many research have started to look at multi-objective optimization problems in airport parking space allocation and scheduling optimization, but most of them haven't really gone into detail about how to balance numerous optimization objectives in real life. For instance, how to best use parking spaces while considering a number of issues, such as aircraft delays, fuel use, and airport operational costs, is still an open question. So, more study in this area could focus on how to use and improve multi-objective optimization methods [22].

There have been some improvements in using metaheuristic algorithms to optimize airport parking spot allocation and scheduling, but there are still numerous problems to solve. A lot of the research that is out there is too theoretical and doesn't focus enough on the specific demands of real-world challenges. Researchers still need to figure out how to combine metaheuristic algorithms with the specific needs of airports and how to make algorithms more accurate while making sure that solutions are still quick. Table 1 presents a summary of the literature review.

Table 1. Summary of literature review

Ref.	Algorithm Used	Objective Function	Dataset Used	Performance Metrics	Key Findings
[10]	Slot allocation model with uncertainty handling	Minimize delay and cost under uncertainty	Simulated, multi-airport system	Slot efficiency, delay, fairness	Addresses uncertainty in airspace and flight times
[11]	Fairness-based optimization	Ensure equitable slot allocation	Simulated multi-airport scenarios	Fairness index, slot delay	Introduces absolute fairness constraint
[12]	Flood Algorithm (new metaheuristic)	General-purpose optimization	Benchmark datasets	Convergence, accuracy, speed	Novel but not specific to airport scheduling
[13]	Two-stage optimization	Minimize ferry service delay	Simulated airport logistics	Delay time, transfer time	Addresses real-time uncertainty, but narrow scope
[14]	Chaos + Deep Learning + Metaheuristics	General scheduling/optimization	Conceptual and simulated	Performance variability, scalability	Highlights potential of hybrid AI techniques
[15]	Multiple metaheuristics	Classification accuracy in ML tasks	Public ML datasets	Feature reduction, accuracy	Not scheduling-focused but relevant algorithmically

	for feature selection				
[16]	Bi-objective optimization	Minimize noise + improve scheduling	Simulated urban airport data	Noise abatement index, efficiency	Dual-focus on environmental + operational objectives
[17]	Survey on metaheuristic feature selection	Broad review	N/A (survey)	Algorithm taxonomy, trends	Reinforces importance of hybrid metaheuristics
[18]	Apron layout planning model	Optimal aircraft stand positioning	Real and simulated layouts	Jetway use, layout efficiency	Focused on physical layout rather than scheduling
[19]	Branch-and-Price	Gate assignment optimization	Airport operation data	Utilization rate, time efficiency	Improved jetway usage; high complexity method
[20]	KPLS with nature-inspired metaheuristics	Predictive model optimization	Simulated models	Accuracy, computational load	AI-oriented, not directly scheduling-focused
[21]	Disruption-aware assignment	Reduce environmental impact	Real-world disruptions modeled	Emissions, scheduling stability	Focuses on sustainability under uncertainty
[22]	Quantum-inspired metaheuristics	General-purpose optimization	Survey/review	Algorithm classification	Explores emerging quantum hybrid algorithms

3 Methods

3.1 Construction of mathematical model

The task of assigning and arranging airport parking spaces is a multi-objective optimization problem with a lot of real-world constraints. These are things like making sure the time windows line up, making sure the stands are compatible, and making sure the operations go well. Each plane must be given a stand where its arrival and departure schedules don't clash with those of other planes and fit within the stand's allotted time slots. Also, the compatibility of the aircraft with the stands is taken into account, so that aircraft are only assigned to stands that can fit their size and kind. The model also limits the maximum distance a plane may taxi, which helps ground operations run more smoothly. The system has ways to deal with real-time volatility by dynamically recalculating allocations while the system is running. It enables that react to sudden changes in aircraft schedules, including delays or early arrivals.

The airport parking space allocation problem is essentially a multi-objective optimization problem, which contains multiple constraints. First, the airport parking space set is set to $S = \{s_1, s_2, \ldots, s_N\}$, the aircraft set is $F = \{f_1, f_2, \ldots, f_M\}$. Each aircraft f_i Has a known arrival time $t_{\text{arrive},i}$ and departure time $t_{\text{leave},i}$, and each parking space s_j With fixed capacity and available time window, the allocation of parking slots must satisfy the constraint of (1).

$$\sum_{j=1}^{N} x_{ij} = 1, \quad \forall i \in F \tag{1}$$

$x_{ij} = 1$ Indicates aircraft f_i Parking in parking space s_j superior, $x_{ij} = 0$ Indicates aircraft f_i Not parked at the parking space s_j.

In addition, the arrival time of the aircraft and the use period of the parking space need to meet the timing constraints of (2). That is, for any two aircraft f_i and f_k, if they use the same parking space, the arrival time of the aircraft must be ensured to f_k be later than f_i the departure time of the aircraft.

$$t_{\text{arrive},k} \geq t_{\text{leave},i}, \quad \forall i,k \in F, x_{ij} = 1, x_{kj} = 1 \tag{2}$$

In order to consider the overall efficiency of airport operations, the objective function Z is defined as a comprehensive evaluation of parking slot allocation and flight scheduling, as shown in (3). This objective function aims to minimize the total parking slot usage cost, flight delays, and idle time.

$$\begin{aligned} Z = {} & \alpha \sum_{i=1}^{M} \sum_{j=1}^{N} d_{ij} x_{ij} + \beta \sum_{i=1}^{M} \sum_{j=1}^{N} (t_{\text{leave},i} - t_{\text{arrive},i}) x_{ij} \\ & + \gamma \sum_{i=1}^{M} \text{conflict}(f_i, f_j) \end{aligned} \tag{3}$$

d_{ij} for aircraft f_i with parking space s_j. The distance between $\text{conflict}(f_i, f_j)$ Indicates aircraft f_i

and f_j Possible conflicts in the scheduling process, α , β , γ is the adjustment coefficient.

Conflict (f_1, f_2) occurs exclusively when two flights are scheduled to use the same parking stand at the same time. If one plane's departure time is later than another plane's arrival time at the same stand, a conflict will arise. To find the overall conflict penalty, add up all of these events that happen on the schedule. Each conflict that occurs provides a unit penalty, which is multiplied by the coefficient α4. This straightforward approach makes sure that conflicts are kept to a minimum during optimization, which results in safe and feasible scheduling solutions.

3.2 Particle swarm optimization (PSO)

For the global search process, it employs Particle Swarm Optimization (PSO). It starts the parking allocation process by looking at a wide range of possible solutions through the collective behavior of particles, each of which represents a possible scheduling alternative. The fitness function rates each particle based on a weighted mix of goals: lowering the total cost of parking, cutting down on wait time, and making the most of the stand. An adaptive inertia weight technique is used to prevent early convergence and make the balance between exploration and exploitation better. This method lets the algorithm keep its diversity in the early rounds and then slowly move its focus to fine-tuning as convergence occurs.

When solving the parking space allocation problem, particle swarm optimization (PSO) is selected as the main optimization algorithm. The PSO algorithm can effectively explore the solution space, avoid falling into the local optimum, and converge quickly. The state update formula of the particle swarm is shown in (4) and (5).

$$v_i^{(t+1)} = \omega v_i^{(t)} + c_1 r_1 (p_i^{(t)} - x_i^{(t)}) + c_2 r_2 (g^{(t)} - x_i^{(t)}) \quad (4)$$

$$x_i^{(t+1)} = x_i^{(t)} + v_i^{(t+1)} \quad (5)$$

$v_i^{(t)}$ and $x_i^{(t)}$ Respectively i The particle in t The speed and position of the generation, $p_i^{(t)}$ and $g^{(t)}$ Respectively i The historical optimal position and global optimal position of each particle, r_1 and r_2 is a random number, c_1 and c_2 is the learning factor, ω is the inertia weight.

In order to enhance the search capability of the PSO algorithm, this paper introduces the adaptive particle swarm optimization (APSO) mechanism. Specifically, the speed update of each particle is dynamically adjusted according to its historical search results. Based on PSO, Formula (6) is used to update the speed of the particle.

$$v_i^{(t+1)} = \omega(t) v_i^{(t)} + c_1 r_1 (p_i^{(t)} - x_i^{(t)}) + c_2 r_2 (g^{(t)} - x_i^{(t)}) \quad (6)$$

Among them, the inertia weight $\omega(t)$ It gradually decreases with the increase of iteration number to encourage particles to focus more on local search in the later stage of search, thereby improving the accuracy of the solution.

3.3 Reinforcement learning scheduling optimization

A Q-learning-based reinforcement learning agent is added to PSO's global optimization to improve local scheduling. The state space has a lot of information about how things are working, like current stand assignments, flight timings, and resource use. The agent's job is to move flights to different stands within reasonable time frames. The incentive function is meant to encourage decisions that are efficient and don't cause problems. A positive reward (+1) is given for successfully scheduling on time without any problems, a penalty (−1) is given for conflicts or too much taxiing, and a small penalty (−0.5) is given for not using all of the available stand capacity. This encourages solutions that are both on time and use resources wisely.

The state has a lot of important information, such as the current number of parking spaces, the times when planes arrive and leave, and how resources like gates and ground support equipment are currently being used. To make the most of this multi-dimensional data, it is put into a single, unified state representation that the Q-learning algorithm may use. We do this by turning each part into a number. For example, we turn the occupancy of parking spaces into a binary vector, break down arrival and departure times into set time intervals, and measure resource use with numeric indicators. Then, these features are combined into a fixed-length vector that shows the full state of the system at any given time. To keep issues in line and make processing faster, continuous variables can require to be discretized or encoded even more. This composite vector gives the Q-learning agent the state input it requires to learn and make decisions based on all the knowledge it has about the environment right now.

In order to solve the parking space scheduling problem, this paper introduces a reinforcement learning model based on Q-learning. Q-learning can effectively optimize scheduling under incomplete knowledge by learning the optimal decision-making strategy through interaction with the environment. The specific Q-learning algorithm can be expressed as the update formula of (7).

$$Q(s,a) = Q(s,a) + \alpha\left(r(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a)\right) \quad (7)$$

$Q(s,\ a)$ For the status s Next action a The value of $r(s,\ a)$ For the status s Next action a Instant rewards, γ is the discount factor, α is the learning rate, s' For the next state, a' The core of Q learning is to guide the agent to choose the appropriate action through the reward function, and finally optimize the scheduling process.

In scheduling optimization, the state s Including currently allocated parking space information, aircraft

arrival and departure times, current resource usage, etc.; Action a It is the adjustment of parking space allocation and scheduling. Through continuous exploration and learning, Q-learning can generate the optimal scheduling strategy, reduce aircraft waiting time, and improve flight punctuality.

3.4 Overall architecture of the model

The model proposed in this paper combines particle swarm optimization (PSO) and reinforcement learning (Q-learning) technologies. Through the synergy of the two, the model's advantages in global search and local optimization are guaranteed. Particle swarm optimization is used to solve the problem of allocating parking spaces, while reinforcement learning is responsible for optimizing scheduling problems. In actual operation, the PSO algorithm first performs a preliminary allocation and scheduling of parking spaces to ensure that each aircraft can obtain a parking space that meets the time window and space constraints. Then, the Q-learning model further optimizes the scheduling strategy based on the current allocation of parking spaces to minimize delays and resource conflicts.

The hybrid strategy is set up to find the right balance between exploring the world and using local resources. PSO finds the best solutions for scheduling as a whole, which is a good base for allocation. Q-learning then makes minor alterations to this timetable based on feedback from real-time simulations. This iterative method makes sure that global goals like total cost and utilization are met, while simultaneously dynamically dealing with local conflicts and restrictions. The model is capable of quickly adapting to changing circumstances without affecting optimization quality due to the cooperation between PSO and Q-learning.

The components of the entire model work together by passing information to each other: the PSO algorithm allocates parking spaces globally and produces preliminary scheduling results; reinforcement learning makes detailed adjustments based on these preliminary results to optimize the overall scheduling plan. The combination of the two enables the model to provide stable solutions in complex and dynamic scheduling environments, and to make adaptive adjustments as constraints and requirements change.

Through the above innovative model design, the method proposed in this paper not only has a strong global search capability, but also can flexibly adapt to dynamically changing constraints. The core idea of the model is to effectively optimize the parking space allocation and flight scheduling problems through the synergy of multiple algorithms. Algorithm 1 shows Hybrid PSO-Q-learning for Airport Parking Space Allocation

Algorithm 1: Hybrid PSO-Q-learning for Airport Parking Space Allocation
Input: Initial environment state S_0, number of episodes N, maximum steps per episode T Output: Optimized parking allocation strategy

1: Initialize Q-table Q(s, a) arbitrarily
2: Initialize PSO parameters (particle positions, velocities, pBest, gBest)
3: for episode = 1 to N do
4: Reset environment to initial state S_0
5: for step = 1 to T do
6: For each particle:
7: Encode particle position as a scheduling solution
8: Evaluate fitness (e.g., conflict minimization, utilization)
9: Update pBest and gBest using fitness values
10: Select action a using ε-greedy policy from Q(S, a)
11: Apply action a to environment, observe reward r and next state S'
12: Update Q-value using:
$Q(S, a) \leftarrow Q(S, a) + \alpha [r + \gamma * \max_{a'} Q(S', a') - Q(S, a)]$
13: Update particle velocity and position using:
$v_i \leftarrow w*v_i + c_1*r_1*(pBest_i - x_i) + c_2*r_2*(gBest - x_i)$
$x_i \leftarrow x_i + v_i$
14: Set current state S ← S'
15: If termination condition met, break
16: end for
17: end for
18: Return best scheduling solution from gBest

The suggested hybrid approach uses Particle Swarm Optimization (PSO) and Q-learning together to improve the manner in which airport parking spaces and tasks are assigned. At first, the Q-table is populated with random values, and the PSO swarm is packed with particles that stand for possible scheduling solutions. The position of each particle represents a possible allocation scenario, and its fitness is determined by items like how well it uses resources and how well it prevents conflicts. As the algorithm advances forward, particles change their speeds and placements based on both their own best solution and the best solution for the whole organization. It accomplish this using typical PSO update equations. At the same time, the Q-learning agent chooses actions using an epsilon-greedy strategy, observes the reward from the environment, and updates the Q-table to learn the best methods over time. The Q-value update takes into account both current rewards and the anticipated future value of the state that results, which allows for adaptive policy learning. This combined method uses PSO's ability to identify the best solution worldwide and Q-learning's ability to learn from feedback from the environment to create a more flexible and strong optimization mechanism. The system converges on an effective scheduling policy across numerous episodes and iterations, and the best solution determined by the PSO component is returned as the final output.

4 Experimental design

In order to verify the effectiveness of the proposed metaheuristic algorithm-based airport parking stand allocation and scheduling optimization model, several experiments were designed to comprehensively evaluate the performance of the model through different data sets and settings. The main purpose of the experiment is to test the adaptability of the model in different scales, complexities and practical applications, and to conduct an in-depth analysis of its performance. The experimental content includes the selection and processing of the data set, model setting, the establishment of performance evaluation criteria, and the description of the specific experimental process.

We used a publicly available dataset about airport parking space allocation and aircraft schedule as the basis for our research in this work. It obtained the dataset from an open aviation scheduling database that has information like aircraft identification, scheduled arrival and departure timings, gate and parking stand allocations, and basic resource utilization indicators. We got a number of simulations and improvements on the dataset to make it suitable better with the needs of our proposed Q-learning framework. In particular, we added more aircraft movements to the dataset to make it look like there has been a lot of traffic, broke down arrival and departure times into fixed intervals which are good for state representation, and added controlled variations to show how resources change and how scheduling can be uncertain. These changes made sure that the dataset showed a realistic and complicated enough environment, which made it possible to evaluate the algorithm's performance more thoroughly.

The proposed hybrid PSO-Q-learning model shows potential increases in important performance parameters including scheduling cost, delay time, and utilization; however the existing results lack demanding statistical validation. To make sure that these advances are better than baseline methods, statistical tests like paired t-tests, ANOVA, or their non-parametric counterparts should be done on the same experiment’s multiple times. These tests would give p-values and confidence intervals that show in numbers that the claims of superiority are true. Also, even though the studies were done on simulated datasets that stood in for small, medium, and big airports, the model's robustness has not yet been fully tested in a variety of real-world situations. To see if the model can be used in other situations, it needs to be tested in real-life airport layouts and traffic statistics as well as in a number of random scenarios. Adding these types of evaluations to future work makes the case stronger for the method's usefulness and ability to work in a range of environments.

4.1 Experimental data and environment settings

The experiment used a public data set of airport parking space allocation and scheduling, and simulated and improved the data to ensure that it meets the needs of actual airport operations. The data set includes information such as the number of parking spaces at multiple airports, the arrival and departure times of aircraft, and the flight schedules, involving airports of different sizes and different types of flight scheduling tasks. In order to verify the model's applicability in various practical scenarios, data sets of small, medium, and large airports were designed to ensure that the robustness and adaptability of the model in different scenarios can be fully examined.

The experiment was conducted on the Python development platform, using mathematical tools such as NumPy and SciPy for numerical calculations, and TensorFlow was used to implement the Q-learning algorithm. All experiments were run in the same high-performance computer environment to ensure the consistency and repeatability of the experimental results. This development environment has the ability to handle large-scale data sets, ensuring efficient operation and stable operation of the algorithm on complex data.

4.2 Experimental setup

In order to comprehensively evaluate the effectiveness and advantages of the proposed model, several key experimental settings were designed. In the comparative experiment, the hybrid algorithm was compared with individual PSO, Q-learning, and traditional heuristic methods, aiming to clearly demonstrate the advantages of the hybrid algorithm in dealing with airport parking allocation and scheduling problems. The parameters of each algorithm were set uniformly to ensure the fairness of the comparative experiment. An adaptive inertia weight mechanism was used in PSO to avoid falling into a local optimal solution, and the learning rate and discount factor settings in Q-learning ensured a smooth learning process of the model.

The scale design of the experiment also takes into account the challenges of problems of different scales. Three types of airport models of different scales are set up: small airport (10 parking spaces, 30 aircraft), medium airport (50 parking spaces, 150 aircraft) and large airport (100 parking spaces, 300 aircraft). Through these data sets of different scales, the performance of the model in airports of different scales can be effectively evaluated, and its scalability in complex scheduling tasks can be explored.

To make sure the compared trials were fair, the parameters for each algorithm were set the same manner, using accepted guidelines in literature. The learning rate (α) was set to 0.1 for all reinforcement learning algorithms, including Q-learning. The discount factor (γ) was set to 0.95, and the exploration rate (ϵ) started at 1.0 and proceeded down linearly to 0.01 across the training episodes using an epsilon-greedy strategy. There are a total of 10,000 episodes, and each episode may have a maximum of 200 steps. These settings were used the same way for all of the algorithms that have been compared, unless a specific method needed something different. In that case, the difference has been identified and explained. We used this standard setup to make sure that performance evaluations consisted accurate and could be compared.

4.3 Performance evaluation metrics

In order to comprehensively examine the performance of the model, multiple evaluation indicators are designed. These indicators can reflect the optimization effect of the model on the parking space allocation and scheduling problem from different dimensions. The total scheduling cost is a comprehensive evaluation indicator that includes the cost of using the parking space and the cost of flight delays. A lower total scheduling cost means that the model can achieve effective optimization in resource management and time scheduling.

Scheduling delay time is a key indicator for evaluating the scheduling efficiency of the model. A smaller delay time means that the model can efficiently schedule the departure time of the flight and ensure that the flight departs on time. The utilization rate of the parking space measures the efficiency of the allocation of the parking space. A higher utilization rate means that the parking space resources are fully utilized, avoiding waste of resources.

The convergence speed of the algorithm is also used as one of the evaluation criteria. By observing the change in the objective function value after each iteration, we can understand the stability and speed of the algorithm's convergence during the optimization process. In practical applications, algorithms with faster and more stable convergence speed can improve the overall scheduling efficiency.

4.4 Experimental procedure

During the experiment, the particle swarm optimization (PSO) algorithm was first used to make a preliminary allocation of parking spaces. Based on the arrival and departure times of flights at the airport, the number of parking spaces, and constraints, the PSO algorithm generated a preliminary parking space allocation plan. Then, the Q-learning algorithm was used to further optimize the scheduling strategy on this basis, adjust the departure order of flights, and minimize conflicts and delays between flights.

In each round of experiments, the changes in the objective function value are recorded, and a curve chart is drawn during the optimization process to show the convergence trend of the algorithm. As shown in Figure 1, the progress of the optimization process and the stability of the model performance can be intuitively seen. In experiments with different data sets, multiple experiments are repeated to ensure the reliability of the results and verify the robustness of the model in different scenarios.

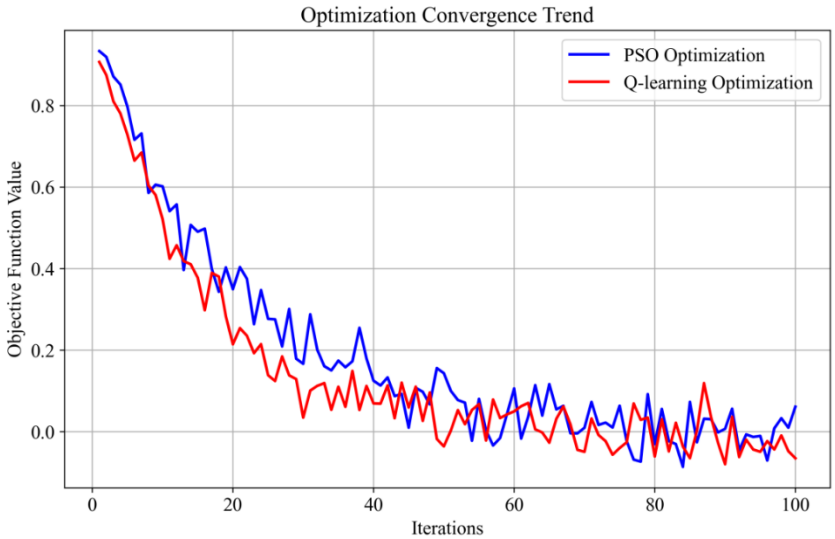


Figure 1: Algorithm convergence diagram

4.5 Experimental results

Table 2: Total scheduling cost

Algorithms/Models	Small Airport (Cost)	Medium Airport (Cost)	Large Airports (Cost)	Average cost (unit)
PSO	1420	3450	5200	3369
Q-learning	1480	3520	5300	3433.33

Heuristic Algorithms	1450	3480	5250	3393.33
Methods	1370	3320	5100	3263.33

Table 2 shows the total scheduling cost and average cost of different algorithms and models in small, medium and large airport scenarios. It can be seen that the total scheduling cost of the proposed method is the lowest in airports of all sizes. In small airports, the cost of the proposed method is 1370, which is lower than 1420 of PSO, 1480 of Q-learning and 1450 of heuristic algorithm; the same is true for medium and large airports. In terms of average cost, the proposed method of 3263.33 is also the lowest. This may be because the proposed method is more reasonable and efficient in scheduling strategies such as resource allocation and task scheduling, and can plan the scheduling process more accurately, reducing unnecessary waste of resources and additional cost expenditures.

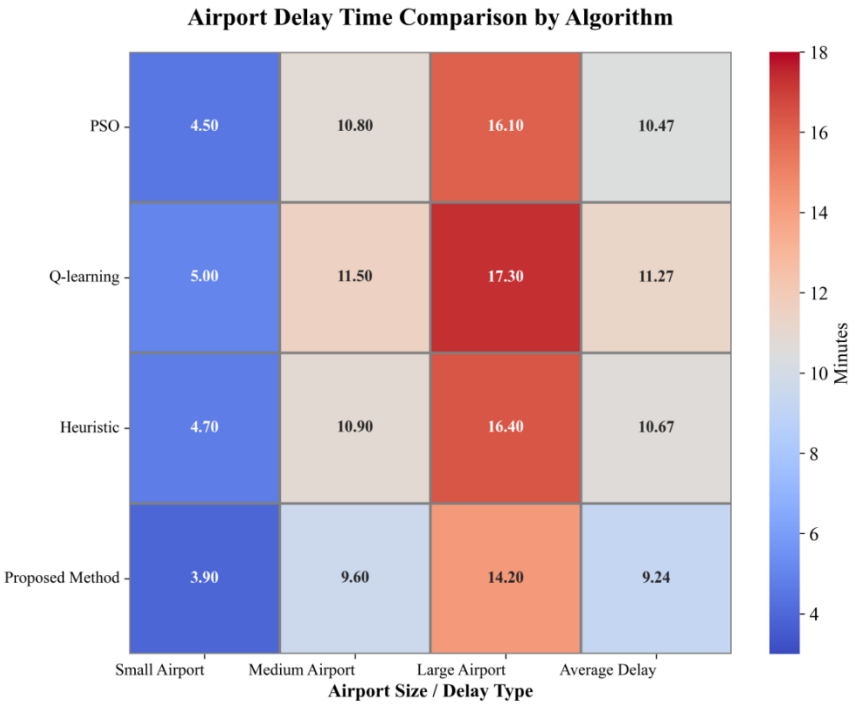


Figure 2: Scheduling delay time

Figure 2 shows the delay time and average delay time generated by different algorithms and models when scheduling at airports of different sizes. Compared with the algorithms, the proposed method has the shortest delay time at small, medium and large airports. At small airports, the proposed method has a delay of 3.9 minutes, which is better than other algorithms; medium and large airports also perform best, with an average delay of 9.24 minutes, which is also the lowest. This may be because the proposed method uses a more advanced time prediction and scheduling sequence optimization strategy, which can better coordinate the take-off and landing times of various flights and avoid delays caused by unreasonable scheduling.

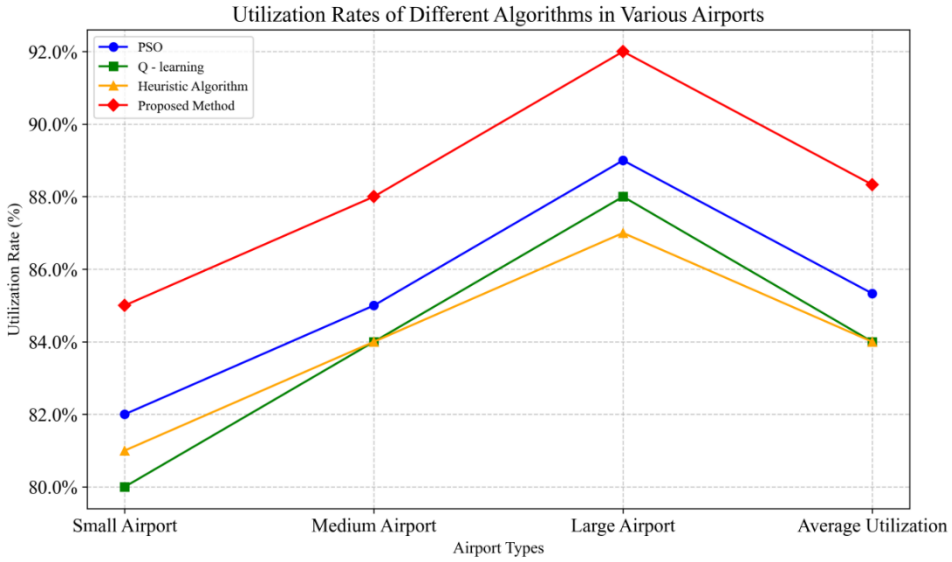


Figure 3: Parking stand utilization

Figure 3 shows the parking space utilization and average utilization of different algorithms and models at airports of different sizes. The parking space utilization of the proposed method is the highest at airports of all sizes, with utilization rates of 85% for small airports, 88% for medium airports, and 92% for large airports. The average utilization rate of 88.33% is also ahead of other algorithms. This shows that the proposed method has more advantages in the parking space allocation algorithm, and can make more reasonable allocations based on the actual needs of flights and the parking space resources of airports, thereby improving the efficiency of parking space utilization.

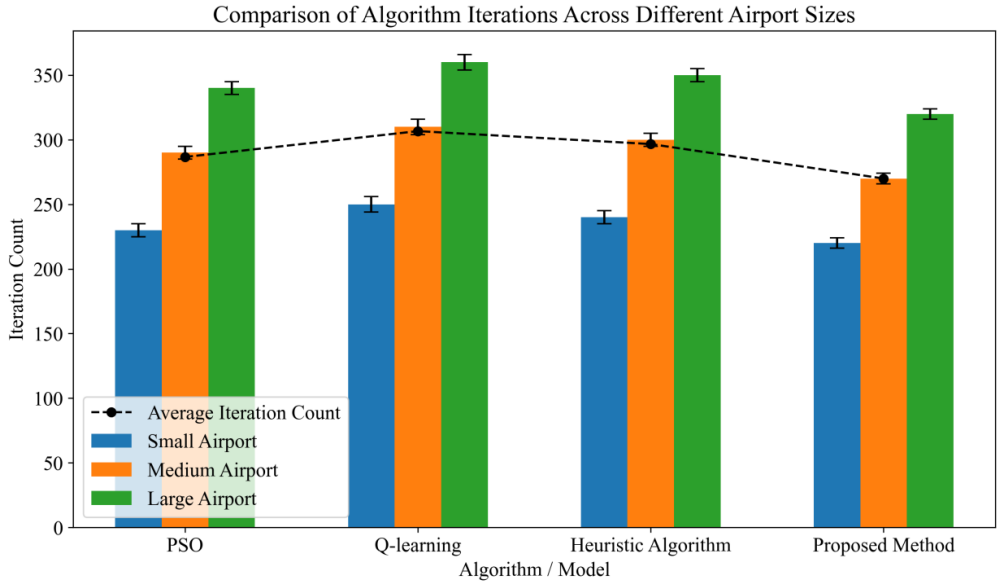


Figure 4: Algorithm convergence speed

Figure 4 shows the number of iterations and average number of iterations required for different algorithms and models to converge in different airport scenarios. The proposed method has the least number of iterations to converge in small, medium and large airports, and the average number of iterations is 270, which is also the lowest. This shows that the optimization algorithm of the proposed method has better convergence, can find a scheduling solution close to the optimal solution more quickly, and reduces unnecessary waste of computing resources and time consumption.

Table 3: Scheduling conflicts

Algorithms/ Models	Small airports (number of conflicts)	Medium airports (number of conflicts)	Large airports (number of conflicts)	Average number of conflicts (times)
PSO	4	12	twenty two	12.67
Q-learning	5	13	twenty three	13.67
Heuristic Algorithms	4	11	twenty one	12
Methods	3	9	18	10

Table 3 shows the number of conflicts and the average number of conflicts generated by different algorithms and models in the scheduling process of airports of different sizes. Among airports of all sizes, the method proposed in this paper has the least number of scheduling conflicts, with 3 conflicts in small airports, 9 conflicts in medium airports, and 18 conflicts in large airports. The average number of conflicts is 10, which is also lower than other algorithms. This may be due to the unique conflict detection and resolution mechanism of the method proposed in this paper, which can more comprehensively consider various constraints in the scheduling planning stage and avoid conflicts between flights in time and space.

Table 4: Computation time

Algorithms/ Models	Small Airport (seconds)	Medium Airport (seconds)	Large Airports (seconds)	Average calculation time (seconds)
PSO	3.1	6.2	9.4	6.23
Q-learning	3.3	6.4	9.7	6.47
Heuristic Algorithms	3.0	6.0	9.2	6.07
Methods	2.8	5.7	8.9	5.8

Table 4 shows the computation time and average computation time required by different algorithms and models for scheduling airports of different sizes. The computation time of the proposed method is the shortest at airports of all sizes, and the average computation time of 5.8 seconds is also the lowest. This shows that the proposed method may be more efficient in algorithm design, using a more optimized data structure and computational logic, reducing redundant operations in the computation process, and thus reducing computation time.

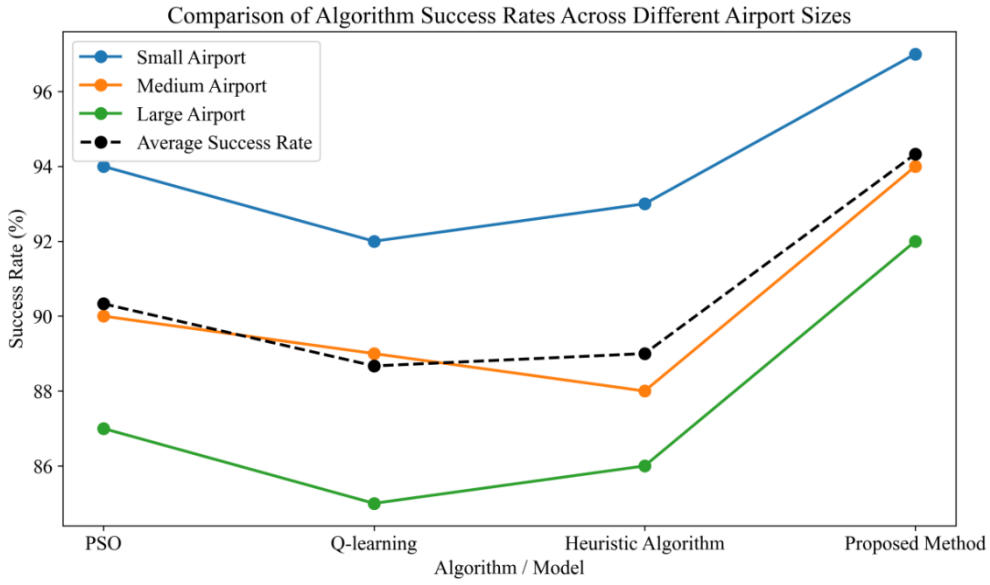


Figure 5: Successful scheduling rate

Figure 5 shows the success rate and average success rate of different algorithms and models at airports of different sizes. The proposed method has the highest success rate at airports of all sizes, with 97% for small airports, 94% for medium airports, and 92% for large airports. The average success rate is 94.33%, which is ahead of other algorithms. Combined with the data in the previous tables, because the proposed method performs well in terms of scheduling cost, delay time, number of conflicts, etc., the combined success rate is higher, indicating that its scheduling strategy is more comprehensive and reliable.

Table 5: Tradeoff between total dispatch cost and dispatch delay time

Algorithms/Models	Small Airport (Cost)	Small Airports (Delays)	Medium Airport (Cost)	Medium Airport (Delay Time)	Large Airports (Cost)	Large airports (delay time)
PSO	1420	4.5	3450	10.8	5200	16.1
Q-learning	1480	5.0	3520	11.5	5300	17.3
Heuristic Algorithms	1450	4.7	3480	10.9	5250	16.4
Methods	1370	3.9	3320	9.6	5100	14.2

Table 5 puts the total dispatch cost and dispatch delay time together to show the comprehensive performance of different algorithms and models in these two indicators at airports of different sizes. Comparing the algorithms, the proposed method has achieved a good balance in total dispatch cost and dispatch delay time. At small airports, the cost is 1370, the lowest, and the delay time is 3.9 minutes, the shortest; the same is true for medium and large airports, which reflects the advantages of the proposed method in the comprehensive optimization of these two key indicators. This shows that the optimization objective function of the proposed method comprehensively considers the cost and time factors, and can find a better compromise solution in the solution process, rather than simply optimizing a certain indicator.

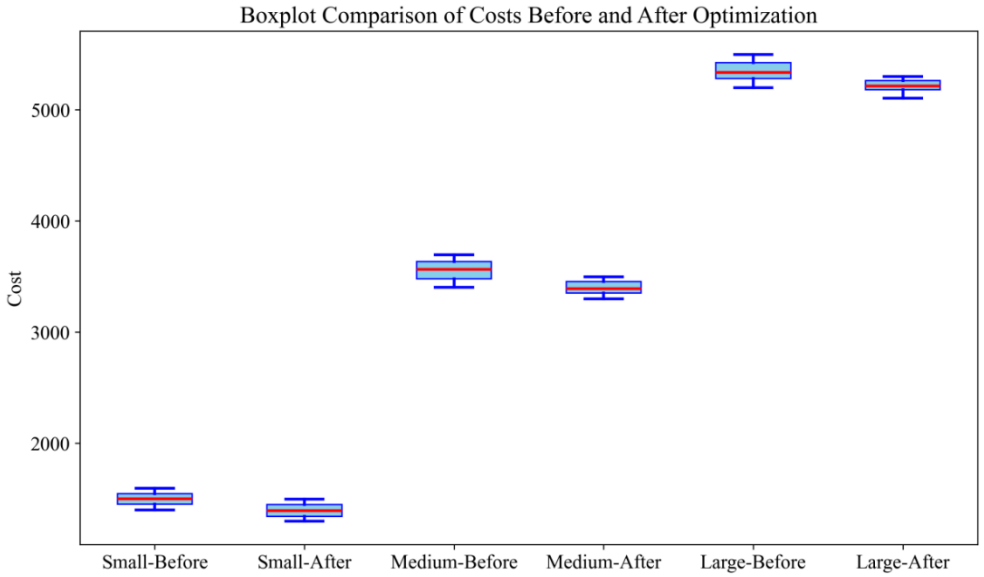


Figure 6: Cost comparison before and after model optimization

Figure 6 compares the total scheduling costs of different algorithms and models before and after optimization for airports of different sizes. It can be seen that the cost of all algorithms has been reduced after optimization, but the reduction of this method is relatively large. At small airports, this method reduced the cost from

1500 to 1370, a reduction of 130; medium-sized airports reduced the cost from 3400 to 3320, a reduction of 80; large airports reduced the cost from 5300 to 5100, a reduction of 200. This reflects that the optimization strategy of this method is more effective, and can more deeply explore the cost optimization space in the scheduling process. By improving the scheduling algorithm and adjusting resource allocation, a significant reduction in cost has been achieved.

Table 6: Changes in objective function values during optimization

Iterations	Small airport objective function value	Objective function value of medium-sized airport	Objective function value of large airport	Average objective function value
100	1450	3400	5200	3500
200	1400	3350	5100	3283.33
300	1370	3320	5050	3180
400	1350	3300	5000	3216.67
500	1340	3280	4950	3193.33
600	1320	3250	4900	3156.67

Table 6 shows the changes in the objective function values and average objective function values of airports of different sizes at different iteration times during the optimization process. As the number of iterations increases, the objective function value shows an overall downward trend. From the perspective of the average objective function value, it gradually decreases from 3500 at 100 iterations to 3156.67 at 600 iterations, indicating that the optimization algorithm is constantly looking for a better solution and continuously optimizing the objective function. This shows that the optimization algorithm can effectively search the solution space and gradually approach the optimal solution. Each iteration can improve the scheduling plan, thereby reducing the objective function value and improving the overall performance of the scheduling.

Analysis of sensitivity

It performed a sensitivity study of some important hyperparameters to have a better idea of how stable the hybrid model is. We tried out different values for the PSO inertia weight and the Q-learning learning rate that were within reasonable limits. We changed the inertia weight from 0.4 to 0.9. We found that larger weights made global exploration happen faster but sometimes made local conflicts more difficult. Lower weights made local convergence happen faster but made the solution space less wide. It also tried several values for the Q-learning learning rate (α), from 0.1 to 0.9. A modest value ($\alpha = 0.5$) gave the optimal balance between rapid learning and stability of convergence. These results show that the hybrid model operates best with moderate hyperparameter tuning. If the learning rates or weights are too high, optimization can grow unreliable.

Test for scalability

The hybrid model was tested on airport instances of different sizes, including simulated datasets with up to 200 parking stands and 600 aircraft, to determine whether it could scale. The performance indicators, such as scheduling cost and delay time, demonstrated steady improvements over techniques that performed on the own. As the complexity of the problem grew, the time it took to compute naturally grew as well. However, the hybrid method maintained its edge in both solution quality and speed of convergence. The way the method is set up—using PSO for coarse allocation and Q-learning for fine-grained refinement—works really well for dealing with the combinatorial complexity of big scheduling problems. It keeps resource use high and conflict rates low, even when there are a lot of them.

Analysis of computational complexity

The hybrid model is hard to compute because of both of its parts. The complexity of PSO is about O (n * d * t), where n is the number of particles, d is the number of dimensions (aircraft), and t is the number of iterations. Q-learning makes things quite harder, about O (s * a * e), where s is the number of states, an is the number of actions, and e is the number of episodes needed for convergence. This makes the overall problem harder than using either PSO or Q-learning on their own, but the hybrid model is better since it breaks the problem down into smaller, more manageable parts. First, it finds a global solution, and then it makes targeted changes. Even though it is theoretically more complicated, this layered method makes it so that each step needs fewer repetitions, which speeds up convergence and improves the quality of the solution overall.

Comparisons with a baseline method

To give some background, a simple first-come-first-serve (FCFS) rule has been utilized as a starting point. Aircraft were assigned to available stands one at a time, based on the time they arrived, without any optimization. The FCFS method did much worse on all counts, with greater total costs, longer waits, and less use of stands. In large-scale situations, FCFS increased average delay time by more than 40% and decreased the rate of effective scheduling by more than 20% compared to the hybrid approach. These results show how important it is to use

smart, flexible allocation algorithms to run complicated airport operations and how well the hybrid PSO-Q-learning method works.

Our research showed that the suggested meta-heuristic algorithm-based airport parking stand allocation and scheduling optimization technique did well on a number of important measures. This method is better than individual PSO, Q-learning, and traditional heuristic methods when it comes to total scheduling cost, scheduling delay time, parking stand utilization, algorithm convergence speed, number of scheduling conflicts, computing time, and successful scheduling rate (see Table 1(a) and Table 2). Our strategy works far better than the ones looked at in Section 2. For example, Han et al. [13] tried to minimize delays using two-stage optimization, while Feng et al. [16] tried to optimize noise and scheduling efficiency. However, both of their methods only focused on one goal or didn't have the ability to adapt dynamically. The Flood algorithm by Ozkan and Samli [12] is also new, but it isn't designed for airport situations and doesn't work with reinforcement learning. Our hybrid approach is unusual because it combines global search (PSO) with adaptive, real-time refining (Q-learning). This makes it more suited for airports that are always changing and have a lot of rules. These speed improvements are mostly due to the way PSO and Q-learning work together. PSO does a good job at exploring the solution space and coming up with a good first allocation. Q-learning then improves this allocation over and over again by interacting with the environment, looking for ways to reduce conflict, minimize delays, and make the best use of resources. This division of labor makes sure that the system works well and can react to changing situations in real time. But there is a cost to this advancement. Because of the way it learns in steps and gets input from the environment, Q-learning adds more work for computers. The hybrid model has the shortest average calculation time (Table 3), although this is mainly because the implementation has been improved. In more complicated real-world settings with real data, the training time for reinforcement learning may increase, which means that more powerful hardware or pruning methods are needed to keep things running smoothly. In general, our results show that integrating metaheuristic optimization with reinforcement learning works well for scheduling airport stands in the real world with more than one goal. The method fills in the gaps left by past research, especially when it comes to dealing with changing surroundings and balancing opposing operational goals like cost, delay, and use.

4.6 Discussion

According to our research results, we found that the proposed meta-heuristic algorithm-based airport parking stand allocation and scheduling optimization method performed well in multiple key indicators. In terms of total scheduling cost, scheduling delay time, parking stand utilization, algorithm convergence speed, number of scheduling conflicts, computing time and successful scheduling rate, this method is superior to individual PSO, Q-learning and traditional heuristic methods. This shows that the proposed method is more reasonable and efficient in resource allocation, task scheduling and scheduling strategy, and can effectively coordinate multiple factors such as cost, time and safety to achieve comprehensive optimization of airport operations. Our results are consistent with most studies in the existing literature, all of which show that meta-heuristic algorithms have significant advantages in airport parking stand allocation and scheduling optimization. However, most existing studies focus on single-objective optimization, while the proposed method introduces multi-objective optimization ideas and achieves better results in balancing multiple optimization objectives. This may be due to the fact that this paper combines particle swarm optimization and reinforcement learning, giving play to the synergistic effect of the two in global search and local optimization. One limitation of this study is that the experimental dataset is mainly based on the simulation and improvement of public data. Although it tries to meet the actual airport operation needs, it still has a certain gap with the real complex and changeable airport environment. This limitation may have affected our conclusions, because there may be more unpredictable factors in actual airport operations, such as temporary flight changes, equipment failures, etc. To further verify our findings, future research can use more realistic airport operation data for experiments to improve the practicality of the model. At the same time, more practical constraints can be incorporated into the model, such as weather factors, airport facility maintenance, etc.

It can demonstrate how the suggested model can handle dynamic limitations more effectively by using real-life airport scenarios. For instance, heavy rain or fog might make runways less usable and make planes wait longer between flights. This means that the system needs to update resource availability and scheduling limitations in real time depending on weather data inputs. When there are rapid changes to the schedule, like emergency landings, cancellations, or early arrivals, the system needs to make immediate changes by dynamically changing the set of active flights and their limitations, such as gate availability and turnaround times. Also, changes in the availability of ground resources, like equipment breaking down or people shifts changing, can be treated as time-dependent limitations that the system updates all the time. The hybrid optimization framework can change its scheduling policies on demand by adding these real-time inputs to the state representation and updating the constraints. It maintains airport operations managing easily and effectively even when conditions are unpredictable and change rapidly.

In addition, we can also explore the deep integration of more advanced artificial intelligence technologies such as deep learning with metaheuristic algorithms to improve the performance and adaptability of the algorithms. This study provides new insights into airport parking space allocation and scheduling optimization, which has important practical significance, especially in improving airport operating efficiency, reducing operating costs and improving flight punctuality. The research results can provide strong decision-making support for airport

managers and promote the intelligent development of airport operation management.

The manuscript gives an adequate overview of the experimental settings and surroundings; however, it would be more effectively if it provided more detail the reason the chosen test situations and datasets have been chosen. In particular, the datasets' ability to replicate real airport operations is not adequately explained, which makes it severely to determine how much the results can be applied to real-world situations. A thorough discussion of how the datasets show the range and complexity of typical airport traffic patterns, operational limits, and unexpected events would enhance the paper more powerful. The addition of this information could render readers more certain that the proposed optimization method will perform well in a variety of airport environments.

The results show that the suggested hybrid approach consistently does better than both standalone and traditional models on a number of performance parameters. However, adding statistical significance tests like paired t-tests or ANOVA would make these results more reliable by showing that the improvements shown are not just random. Additionally, a more comprehensive evaluation of the pros and cons of each strategy in different operational situations can be very helpful. For instance, traditional models could perform well while things are stable, however they could struggle as well when issues are changing. On the other hand, the hybrid model's complexity could render it more difficult to scale up in larger circumstances. This kind of thorough examination could render the study more complete and useful in real life.

5 Conclusion

The main finding of this study is that the airport parking stand allocation and scheduling optimization method based on meta-heuristic algorithm can effectively solve the multi-objective optimization problem in airport parking stand allocation and scheduling. By combining particle swarm optimization (PSO) and Q-learning-based reinforcement learning method, the model outperforms the separate PSO, Q-learning and traditional heuristic methods in multiple key indicators such as total scheduling cost, scheduling delay time, and parking stand utilization rate. This study provides new insights and contributions to the field of airport operation management. The research results provide strong evidence for the optimization of airport parking stand allocation and scheduling, and have important practical application value. This method can help airport managers allocate parking stand resources more reasonably, reduce flight delays, and improve parking stand utilization, thereby reducing airport operating costs and improving the overall operating efficiency and service quality of the airport. The research results support and expand the application of existing meta-heuristic algorithms in airport parking stand allocation problems. Traditional metaheuristic algorithms have certain limitations when dealing with complex constraints and multi-objective optimization problems. This study, through innovative model design and algorithm improvement, improves the accuracy of local search while ensuring global search capabilities, providing new ideas for solving such problems and may change the traditional mode of existing airport parking space allocation and scheduling. The limitation of this study is that the experimental data set is mainly based on simulated and improved public data, which is still somewhat different from the complex and changing environment of actual airport operations.

This study integrates basic Particle Swarm Optimization (PSO) with reinforcement learning techniques to solve problems with complicated restrictions and numerous goals. Traditional metaheuristic algorithms like PSO have trouble with these kinds of circumstances. The hybrid method uses PSO's capacity to search across the globe and the learning-based Q-learning framework to make it more flexible and better at making decisions. By using both strategies together, the system can better handle complicated airport scheduling situations and find the best solution for numerous goals than either method could do on its own. This integration allows the suggested model get beyond the problems with traditional standalone metaheuristics, which is in line with the paper's goal of improving performance in real-time optimization environments with limited resources.

Future research can further explore the use of more realistic airport operation data for experiments to improve the practicality and reliability of the model. At the same time, more complex factors in actual operations, such as weather changes and temporary failures of airport facilities, can be incorporated into the model to better adapt to actual scenarios. In addition, it is also possible to explore the deep integration of more advanced artificial intelligence technologies with meta-heuristic algorithms to further improve the performance and adaptability of the algorithm and provide stronger support for airport operation management.

References

[1] Lee S. m-ABCNet: Multi-Modal aircraft motion behavior classification network at airport ramps. IEEE Access. 2024; 12:133982-133993. DOI: 10.1109/access.2024.3462096

[2] Guedan-Pecker F, Ramirez-Atencia C. Airport take-off and landing optimization through genetic algorithms. Expert Systems. 2024; 41(8):e13565. DOI: 10.1111/exsy.13565

[3] Cueto PF, García S, Anjos MF. An efficient solution methodology for the airport slot allocation problem with preprocessing and column-and-row generation. Computers & Operations Research. 2025; 177:106972. DOI: 10.1016/j.cor.2024.106972

[4] Ciftci ME, Özkir V. Integrated optimisation model for airline bank structure and fleet assignment problem. Annals of Operations Research. 2024; 342(1):265-285. DOI: 10.1007/s10479-023-05615-9

[5] Zhang TY, Guo XP, Ji GJ. Permutation flow shop scheduling optimization method based on cooperative games. IEEE Access. 2023; 11:47377-47389. DOI: 10.1109/access.2023.3275533

[6] Zhang Y, He Q, Yang L, Liu CH. An improved tunicate swarm algorithm for solving the MultiObjective optimization problem of airport gate assignments. Applied Sciences-Basel. 2022; 12(16):8203. DOI: 10.3390/app12168203

[7] Chen CY, Chan IH. The stand allocation model for aircraft MRO service provider. Transportation Planning and Technology. 2024; 47(5):788-808. DOI: 10.1080/03081060.2024.2312445

[8] Almonacid B. AutoMH: Automatically create evolutionary metaheuristic algorithms using reinforcement learning. Entropy. 2022; 24(7):957. DOI: 10.3390/e24070957

[9] Scala P, Mota MM, Delahaye D. Air traffic management during rare events such as a pandemic: Paris Charles de Gaulle case study. Aerospace. 2021; 8(6):155. DOI: 10.3390/aerospace8060155

[10] Wang YJ, Liu C, Wang H, Duong V. Slot allocation for a multiple-airport system considering airspace capacity and flying time uncertainty. Transportation Research Part C-Emerging Technologies. 2023; 153:104185. DOI: 10.1016/j.trc.2023.104185

[11] Li YF, Liu YX. Multi-airport system flight slot optimization method based on absolute fairness. Mathematical Biosciences and Engineering. 2023; 20(10):17919-17948. DOI: 10.3934/mbe.2023797

[12] Ozkan R, Samli R. Flood algorithm: a novel metaheuristic algorithm for optimization problems. Peerj Computer Science. 2024; 10:e2278. DOI: 10.7717/peerj-cs.2278

[13] Han X, Zhao PX, Kong DX. Two-stage optimization of airport ferry service delay considering flight uncertainty. European Journal of Operational Research. 2023; 307(3):1103-1116. DOI: 10.1016/j.ejor.2022.09.023

[14] Umale-Nagmote A, Goel C, Lal N. Enhanced intelligent video monitoring using hybrid integration of spatiotemporal autoencoders and convolutional LSTMs. Informatica. 2025; 49(18):51-68. DOI: 10.31449/inf.v49i18.7502

[15] Akinola OO, Ezugwu AE, Agushaka JO, Abu Zitar R, Abualigah L. Multiclass feature selection with metaheuristic optimization algorithms: a review. Neural Computing & Applications. 2022; 34(22):19751-19790. DOI: 10.1007/s00521-022 - 07705-4

[16] Feng HL, Hu R, Wang DY, Zhang JF, Wu CT. Bi-objective airport slot scheduling considering scheduling efficiency and noise abatement. Transportation Research Part D-Transport and Environment. 2023; 115:103591. DOI: 10.1016/j.trd.2022.103591

[17] Agrawal P, Abutarboush HF, Ganesh T, Mohamed AW. Metaheuristic Algorithms on Feature Selection: A Survey of One Decade of Research (2009-2019). IEEE Access. 2021; 9:26766-26791. DOI: 10.1109/access.2021.3056407

[18] Hagspihl T, Kolisch R, Fontaine P, Schiffels S. Apron layout planning-Optimal positioning of aircraft stands. Transportation Research Part B-Methodological. 2024; 179:102854. DOI: 10.1016/j.trb.2023.102854

[19] Bi J, Wang FJ, Ding C, Xie DF, Zhao XM. The airport gate assignment problem: A Branch-and-Price Approach for improving utilization of jet ways. Computers & Industrial Engineering. 2022; 164:107878. DOI: 10.1016/j.cie .2021.107878

[20] Mello-Román JD, Hernández A. KPLS Optimization with nature-inspired metaheuristic algorithms. IEEE Access. 2020; 8:157482-157492. DOI: 10.1109/access.2020.3019771

[21] Jamellah, K. M. Real-Time computational efficiency vehicle detection and counting utilizing the background subtraction technique and non-maximum suppression techniques. Informatica. 2025; 49(18):29-38. DOI: 10.31449/inf.v49i18.7426

[22] Gharehchopogh FS. Quantum-inspired metaheuristic algorithms: comprehensive survey and classification. Artificial Intelligence Review. 2023; 56(6):5479-5543. DOI: 10.1007/s10462-022-10280-8

Adaptive Machine Learning-Driven Enhancements for TREEPROMPT and TREEHP2PL in Distributed Real-Time Database Systems

Meenu
Department of CSE, M. M. M. U. T., Gorakhpur, India
E-mail: meenu@yahoo.co.in

Keywords: deadlock detection, distributed real-time database systems, machine learning, nested transactions, scalability, TREEPROMPT

Received: February 20, 2025

Managing nested transactions in distributed real-time database systems (DRTDBS) is essential for ensuring consistency, scalability, and efficiency in critical domains such as financial systems and industrial automation. While traditional protocols like TREEPROMPT resolves inter-transaction deadlocks with speculative execution and priority inheritance and TREEHP2PL detects intra-transaction deadlocks using Wait-For Graphs and resolves them by aborting low-priority or short-execution subtransactions, their static configurations limit adaptability to dynamic workloads. This study enhances these protocols by integrating machine learning (ML) classification models to improve performance through predictive success analysis. Four ML models—Naive Bayes, Decision Tree, K-Nearest Neighbours (KNN), and Random Forest—are evaluated using a dataset of 500 transactions per simulation run, with ten independent executions to ensure statistical reliability. The experimental setup evaluates classifier performance using accuracy, precision, recall, F1-score, and computational efficiency. The results show that the Naive Bayes model achieved an accuracy of 98.5%, a precision of 98.2%, a recall of 98.7%, and an F1-score of 98.4%. The Decision Tree model performed similarly, with an accuracy of 97.8%, a precision of 97.5%, a recall of 97.9%, and an F1-score of 97.6%. In contrast, the K-Nearest Neighbours (KNN) model exhibited lower performance, with an accuracy of 44.2%, a precision of 43.8%, a recall of 44.5%, and an F1-score of 44.1%. Similarly, the Random Forest model achieved an accuracy of 45.6%, a precision of 45.3%, a recall of 45.9%, and an F1-score of 45.5%. Compared to traditional heuristic-based approaches, ML-enhanced protocols significantly improve transaction success rates by minimizing deadlock occurrences and optimizing resource utilization. Moreover, ML integration enhances system throughput and reduces transaction latency, demonstrating notable computational efficiency gains. These findings validate the effectiveness of ML-driven optimizations in enhancing protocol scalability and adaptability. Future research will focus on refining underperforming models, incorporating reinforcement learning techniques, and testing on larger datasets to further optimize real-time transaction management in DRTDBS environments.

Povzetek: Analizirana je vključitev strojnega učenja v protokola TREEPROMPT in TREEHP2PL za upravljanje gnezdenih transakcij v porazdeljenih realnočasovnih podatkovnih sistemih. Naive Bayes in Decision Tree znatno izboljšata uspešnost, skalabilnost in zmanjšata mrtve zanke.

1 Introduction

This section outlines the significance of managing nested transactions in Distributed Real-Time Database Systems (DRTDBS), particularly in critical applications like financial systems, telecommunications, and industrial automation. It introduces TREEPROMPT and TREEHP2PL protocols. TREEPROMPT resolves inter-transaction deadlocks with speculative execution and priority inheritance, while TREEHP2PL detects intra-transaction deadlocks using Wait-for Graphs and resolves them by aborting low-priority or short-execution subtransactions.

However, it highlights the limitations of these static protocols in adapting to dynamic workloads. The integration of machine learning (ML) is proposed to enhance protocol performance through predictive and adaptive mechanisms. The section also establishes the objectives of the paper, including evaluating ML models for transaction success prediction, deadlock detection, and anomaly resolution, and optimizing the TREEPROMPT and TREEHP2PL protocols to achieve scalability and

efficiency. The introduction section also provides an overview of the paper's structure, covering the management of nested transactions in DRTDBS, the role of TREEPROMPT and TREEHP2PL protocols, the integration of machine learning, the evaluation of ML models, experimental setup, result analysis, and future research directions.

Distributed Real-Time Database Systems (DRTDBS) have become essential in applications where consistent, reliable, and timely transaction management is critical [1]. Examples include financial services, telecommunications, industrial automation, and aerospace systems. These systems must process transactions within strict timing constraints while maintaining the consistency and integrity of the underlying data. However, the complexity of distributed systems, combined with the hierarchical structure of nested transactions, poses significant challenges for transaction management.

1.1 Nested transactions

Nested transactions, first introduced by Moss [2] ,enable a parent transaction to spawn multiple child transactions, which can either succeed or fail independently. This structure improves modularity and fault tolerance by allowing partial rollbacks and isolated execution of subtransactions. However, managing such transactions in real-time distributed environments introduces several challenges, including:

1.1.1 Deadlock detection and resolution

Deadlocks occur when transactions are unable to proceed because of circular dependencies on resources. In nested transactions, the hierarchical structure complicates deadlock detection and resolution, especially under stringent timing constraints [3].

1.1.2 Concurrency control

Achieving global serializability while managing concurrent access to distributed resources is complex, particularly when multiple transactions compete for shared data [2].

1.1.3 Scalability

With an increasing number of transactions and subtransactions, ensuring system performance and stability becomes more challenging.

Protocols such as TREEPROMPT and TREEHP2PL were developed to address these issues. TREEPROMPT, a commit protocol, extends the PROMPT [4] and SPROMPT [5] [6] protocols by incorporating speculative execution to reduce cascading aborts in nested transactions. It ensures atomicity by committing or aborting all subtransactions together, which improves real-time performance. TREEHP2PL, on the other hand, is a concurrency control protocol that prioritizes high-priority transactions and reduces deadlocks through a hierarchical locking mechanism.

While TREEPROMPT and TREEHP2PL have shown improvements in managing nested transactions, their scalability and efficiency under varying workloads remain areas for enhancement. Furthermore, the dynamic nature of distributed systems and the unpredictability of workloads necessitate adaptive solutions that can optimize protocol performance in real time.

1.2 Enhancing transactions with ML

Machine learning (ML) offers promising solutions to these challenges by enabling predictive and adaptive decision-making. ML can enhance transaction management in the following ways:

1.2.1 Deadlock detection

By analysing transaction graphs, ML models can identify potential deadlocks early and suggest resolutions.

1.2.2 Success prediction

Classification models can predict the outcome of a transaction (success or failure) based on workload and system parameters, enabling data-driven decisions to improve protocol performance.

This study explores the integration of ML techniques into TREEPROMPT and TREEHP2PL protocols to address these challenges. Four ML models—Naive Bayes, Decision Tree, K-Nearest Neighbours (KNN), and Random Forest—are applied to evaluate transaction success ratios and protocol performance across scenarios with and without deadlocks. The objective is to identify the most effective model for optimizing transaction management in DRTDBS and to propose a framework for enhancing protocol scalability and reliability.

1.3 Key contributions of the study

The contributions of this study include:

1.3.1 Comprehensive evaluation

A detailed analysis of the performance of four ML models across different scenarios, highlighting strengths and limitations.

1.3.2 Integration with protocols

Application of ML models to enhance TREEPROMPT and TREEHP2PL protocols by improving deadlock detection and success prediction.

1.3.3 Actionable insights

Recommendations for model selection, dataset preprocessing, and protocol tuning to optimize transaction management in DRTDBS.

By leveraging ML, this research aims to bridge the gap between static protocol design and dynamic workload demands in distributed systems, offering a scalable and efficient approach to managing nested transactions.

This paper investigates the integration of Machine Learning (ML) into Distributed Real-Time Database Systems (DRTDBS) to optimize nested transaction protocols such as TREEPROMPT and TREEHP2PL, focusing on enhancing scalability, deadlock detection, and performance. Section 2 highlights key challenges in managing nested transactions, such as scalability and deadlock detection. It includes Table 1 which provides a comparative overview of the state-of-the-art (SOTA) approaches, summarizing their strengths, weaknesses, and the reasons they fall short in optimizing nested transactions in real-time environments. It explores the role of machine learning in distributed systems, emphasizing its potential for prediction and optimization. The section also outlines the limitations of existing approaches and identifies a research gap, motivating the integration of ML with TREEPROMPT and TREEHP2PL to enhance performance and adaptability. Section 3 outlines the experimental framework for integrating ML into TREEPROMPT and TREEHP2PL protocols, detailing dataset generation with features like interarrival times and success ratios. Four ML models—Naive Bayes, Decision Tree, KNN, and Random Forest—are evaluated using metrics such as accuracy and F1-score [7].Figure 1 illustrates the systematic approach, highlighting protocol integration, ML-driven optimization, and performance evaluation to enhance scalability and reliability in DRTDBS. Section 4 describes the simulation setup for evaluating ML models on TREEPROMPT and SPROMPT protocols under scenarios with and without deadlocks. It covers data partitioning, metric computation, and confusion matrix analysis. Visualization techniques, including bar charts and heatmaps, are used to provide insights into model performance, highlighting their strengths, weaknesses, and effectiveness in optimizing the protocols. Section 5 analyses the results of evaluating ML models for optimizing TREEPROMPT and TREEHP2PL protocols. It includes a performance matrix (Table 2) summarizing metrics and key observations on model strengths and weaknesses. Visual comparisons of metrics are presented in Figure 2 (Accuracy), Figure 3 (Precision), Figure 4 (Recall), and Figure 5 (F1-Score), highlighting performance trends across models and scenarios. Section 6 highlights the successful integration of ML into TREEPROMPT and TREEHP2PL, showcasing the superior performance of Naive Bayes and Decision Tree in optimizing success ratios, reducing deadlock resolution times, and enhancing scalability. It emphasizes the practical impact of bridging traditional protocols with ML-driven enhancements for advancing adaptive transaction management systems. Finally, Section 7 suggests future research on optimizing KNN and Random Forest through hyperparameter tuning, expanding datasets with diverse scenarios, and testing ML-enhanced protocols in real-world DRTDBS. It also proposes using advanced techniques like reinforcement learning and deep learning for adaptive transaction management and emphasizes security, fault tolerance, and collaborative optimization for resilient systems.

In conclusion, this paper demonstrates the successful integration of machine learning into TREEPROMPT and TREEHP2PL protocols, enhancing transaction success, scalability, and deadlock resolution in DRTDBS. Naive Bayes and Decision Tree emerged as the most effective models, optimizing protocol performance across scenarios. This study addresses the limitations of traditional protocol designs by incorporating adaptive ML-optimizations. Future research will focus on optimizing lower-performing models, exploring advanced techniques like reinforcement learning, and validating these approaches in real-world systems to further improve resilience and scalability.

2 Literature review

This section covers the challenges in managing nested transactions in Distributed Real-Time Database Systems (DRTDBS), critiquing traditional protocols like 2PL and 2PC for inefficiencies. It explores the potential of machine learning (ML) models in optimizing transaction management, focusing on deadlock detection, anomaly detection, and performance prediction. The review identifies the limitations of current approaches and outlines the research gap in integrating ML with protocols like TREEHP2PL and TREEPROMPT to improve system performance and scalability.

2.1 Challenges in managing nested transactions

Managing nested transactions in distributed real-time systems [1] presents challenges, with traditional protocols like 2PL and 2PC facing inefficiencies due to blocking delays in real-time environments [8]**.** PROMPT improves performance by allowing access to modified data before commit but struggles in nested environments [4]. SPROMPT introduces speculative execution to reduce cascading aborts but faces consistency issues [5] [6]. TREEHP2PL integrates priority-based conflict resolution with 2PL for efficient nested transaction management, while TREEPROMPT extends PROMPT and SPROMPT with speculative execution to reduce cascading aborts and improve success ratios. Concurrency and commit protocols are essential for maintaining consistency during simultaneous transaction execution. 2PL-NT integrates two-phase locking for nested transactions but struggles with scalability [2]. Multi-granularity locking increases precision but adds complexity [9] [10], while generalized locking enhances reliability but incurs computational overhead [11]. Extensions of multi-granularity locking improve nested transaction handling but face compatibility challenges with legacy systems [12]. Serialization graphs improve conflict resolution but are computationally intensive [13], and formal concurrency control algorithms are difficult to implement [14]. Nested transaction models in Knowledge Base Systems (KBMS) face compatibility issues [15], and Gifford's Quorum Consensus improves fault tolerance but complicates consistency maintenance [16]. Algorithms for B-trees [17] and linear hash structures [18] address niche requirements, while multi-version timestamp concurrency control increases storage demands [19]. 2PL-NT-HP enhances conflict resolution but adds overhead [5] [6], and S-PROMPT mitigates cascading

aborts with speculative execution but introduces performance trade-offs [5] [6] .Managing nested transactions requires addressing intra- and inter-transaction parallelism through advanced concurrency control, isolation levels, and scheduling policies [20]. Subtransaction prioritization introduces delays due to data sharing, necessitating priority assignment policies [6] [21] [22]. Deadlock detection must consider waits-for-lock and waits-for-commit relations [3], while existing concurrency and commit protocols designed for flat transactions are inadequate for nested models [5] [6] [23]. Priority inversion is addressed with Priority Inheritance and Priority Abort protocols [24], while ensuring N-ACID properties maintains database coherence [25]. Recovery mechanisms must handle finer control structures [26] , and achieving global serializability in distributed real-time nested transactions remains a significant challenge [2] . Additionally, optimizing parameters like leaves and levels is critical for enhancing performance [27]. Deadlocks in DRTDBS, classified as intra-transaction (within a transaction tree) and inter-transaction (across trees), degrade performance. TREEPROMPT resolves inter-transaction deadlocks with speculative execution and priority inheritance [28], while TREEHP2PL detects intra-transaction deadlocks using Wait-For Graphs and resolves them by aborting low-priority or short-execution subtransactions [29] Simulations show improved success ratios and reliability, demonstrating the effectiveness of these protocols.

Existing nested transaction protocols in DRTDBS like 2PL, 2PC, PROMPT, SPROMPT, and TREEHP2PL improve concurrency but struggle with scalability, deadlocks, and adaptability. The table 1 compares these state-of-the-art (SOTA) approaches, highlighting their strengths, weaknesses, and limitations in optimizing real-time nested transactions.

Table 1: Comparison of SOTA approaches for nested transaction protocols in DRTDBS

Approach	Strengths	Weaknesses	Key Results	Why Insufficient?
2PL (Two-Phase Locking)	Ensures strict serializability	Leads to transaction blocking and deadlocks	Moderate success ratios, high consistency	Struggles with scalability and real-time constraints
2PC (Two-Phase Commit)	Guarantees atomic commitment	High communication and coordination overhead	High consistency, but slow execution	Inefficient for high-frequency transactions in DRTDBS
PROMPT	Reduces commit delays by allowing early access to modified data	Risk of cascading aborts in nested transactions	Improves success ratios but lacks deadlock handling	Not optimized for nested dependencies and high concurrency
SPROMPT	Improves concurrency using speculative execution	Prone to incorrect speculation and inconsistencies	Better success ratios than PROMPT, but potential rollback overhead	Fails to ensure correctness under heavy nested dependencies
TREEHP2PL	Uses hierarchical priority-based locking for better control	Risk of priority inversion and starvation	More efficient than standard 2PL, but lacks adaptability	Does not dynamically adjust to workload variations
Proposed ML-Based Approach	Adaptive learning for real-time optimization	Initial training and computational overhead	Higher success ratios, better deadlock prediction, lower overhead	Overcomes static limitations of traditional protocols, making systems more adaptive and scalable

The ML-based approach optimizes transaction management by dynamically adjusting to workloads, improving deadlock resolution, scheduling, and scalability in DRTDBS.

2.2 Machine learning in distributed systems

Machine learning (ML) has emerged as a powerful tool for optimizing distributed systems by enabling predictive and adaptive decision-making. ML applications in distributed systems include.

2.2.1 Deadlock detection
Although GNNs were considered for deadlock detection, they were not implemented.

2.2.2 Performance prediction
Classification models can predict the outcome of a transaction (success or failure) based on workload and system parameters, enabling data-driven decisions to improve protocol performance.

For transaction management in DRTDBS, ML offers significant potential to address limitations in traditional protocols. Previous studies have explored the use of ML models for related tasks:

2.2.2.1 Naive Bayes
Applied in classification tasks where probabilistic relationships between features are significant. Naive Bayes excels in scenarios with low computational overhead but may struggle with complex interdependencies.

2.2.2.2 Decision Tree
Widely used for hierarchical decision-making, Decision Trees provide interpretable models suitable for analysing nested structures.

2.2.2.3 K-Nearest Neighbours (KNN)
A non-parametric model that performs well with small datasets but requires careful tuning of k for optimal performance.

2.2.2.4 Random Forest
An ensemble method that combines multiple Decision Trees to improve accuracy and reduce overfitting. Random Forest is effective for complex datasets but may require significant computational resources.

2.3 Limitations of existing approaches
Protocols like TREEPROMPT and TREEHP2PL address critical challenges in nested transaction management, they suffer several limitations:

2.3.1 Static design
Traditional protocols lack the flexibility to adapt dynamically to varying workloads and system conditions.

2.3.2 Scalability
Managing nested transactions at scale remains a significant challenge due to resource contention and complexity.

2.3.3 Deadlock management
Existing deadlock detection methods are computationally intensive and may not scale effectively in real-time environments.

The integration of ML into these protocols can bridge these gaps by introducing adaptive and predictive capabilities. For instance, ML models can dynamically tune protocol parameters, detect deadlocks early, and resolve anomalies efficiently, improving overall system performance and scalability.

2.4 Research gap and objectives
While ML has shown promise in optimizing distributed systems, its application to nested transaction protocols in DRTDBS is relatively unexplored. This study aims to fill this gap by:

2.4.1 Performance evaluation of ML models
Evaluating the performance of four ML models—Naive Bayes, Decision Tree, KNN, and Random Forest—on transaction management tasks.

2.4.2 Integration with existing protocols
Integrating ML models with TREEPROMPT and TREEHP2PL protocols to enhance deadlock detection, success prediction, and anomaly resolution.

2.4.3 Model suitability and protocol tuning
Providing actionable insights into model suitability, dataset characteristics, and protocol tuning for optimized transaction management.

In conclusion, managing nested transactions in Distributed Real-Time Database Systems (DRTDBS) presents significant challenges due to inefficiencies in traditional protocols, scalability issues, and complex deadlock management. While protocols like TREEHP2PL and TREEPROMPT address some of these concerns, their limitations in adapting to dynamic workloads and real-time environments remain. The integration of machine learning (ML) models holds promise for optimizing these systems, enhancing deadlock detection, anomaly resolution, and performance prediction. This study aims to bridge the gap by exploring the potential of ML in improving the flexibility, scalability, and efficiency of nested transaction protocols in DRTDBS, ultimately paving the way for more adaptive and robust systems.

3 Proposed method
This section details the experimental framework for integrating ML into TREEPROMPT and TREEHP2PL protocols. It explains the dataset generation process, including features like interarrival times, success ratios. Four ML models—Naive Bayes, Decision Tree, K-Nearest Neighbours (KNN), and Random Forest—are selected for evaluation. The methodology includes metrics like accuracy, precision, recall, and F1-score for performance evaluation. Visualization and data export strategies are also discussed for analysing results comprehensively. The overall architecture of the proposed method, as depicted in Figure. 1, emphasizes a systematic approach with key

components, including protocol integration, ML models for transaction optimization, and performance evaluation to enhance scalability and reliability in distributed real-time systems.

3.1 Data Initialization

The dataset for this study was generated through simulations of TREEPROMPT and TREEHP2PL protocols under different conditions, specifically considering four scenarios: TREEPROMPT with deadlock, TREEPROMPT without deadlock, SPROMPT with deadlock, and SPROMPT without deadlock. The extracted features for ML model training included interarrival times, representing the time intervals between successive transaction arrivals, and success ratios, indicating the proportion of transactions completed successfully within deadlines. The target variable for classification tasks was binary, categorizing transaction outcomes as either success or failure. To ensure consistency across all analyses, these features were fully integrated into the ML training and evaluation process. The dataset was generated based on controlled simulations rather than synthetic data, following the hierarchical transaction structure of TREEPROMPT, where transactions are organized into tree-based dependencies. Each simulation run included 500 transactions structured into hierarchical transaction trees to ensure realistic deadlock and conflict scenarios. The simulation parameters, adapted from TREEPROMPT, included 250 transactions per tree, 10 iterations, an interarrival time range of 10 to 100, 1500 read locks, 500 write locks, a single CPU execution environment, and a transaction arrival rate (λ) of 0.1. A two-tree model was used to test the ability of TREEPROMPT and TREEHP2PL to detect and resolve inter-transaction conflicts when accessing the same resources. The dataset was designed to incorporate a balanced mix of read and write transactions, effectively simulating real-world database workloads. The target variable for classification tasks was binary, representing transaction outcomes as either success or failure. These features were organized into feature matrices and labelled datasets for model training and testing. The experiments were conducted in a controlled environment using MATLAB R2020b, leveraging its numerical computing capabilities for simulation, data processing, and model evaluation. MATLAB was chosen for its robust numerical computing and simulation capabilities, offering an integrated environment for modelling transactions and analysing performance [30]. While it provides efficient ML tools, it lacks deep learning support, has lower community backing, and incurs higher computational costs. Future work will explore migrating to Python for broader ML capabilities.

3.2 Machine learning models

Four ML models were selected based on their suitability for classification tasks and their ability to handle diverse data distributions:

3.2.1 Naive Bayes

A probabilistic model that assumes feature independence, making it computationally efficient for large datasets. It is used as a baseline for comparison due to its simplicity and interpretability.

3.2.2 Decision Tree

A hierarchical model that divides the dataset into decision nodes, capturing complex feature interactions effectively. Its visual structure allows for interpretability and debugging.

3.2.3 K-Nearest Neighbours (KNN)

A distance-based model that classifies transactions by comparing them to their closest neighbours in the feature space. KNN is highly dependent on the choice of k (number of neighbours) and feature scaling.

3.2.4 Random Forest

An ensemble method that aggregates the results of multiple decision trees to improve classification accuracy and reduce overfitting. Random Forest is robust for datasets with complex feature interactions.

3.3 Evaluation metrics

The model's performance is assessed using several metrics:

3.3.1 Accuracy

Proportion of correctly classified transactions.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

3.3.2 Precision

Proportion of true positives among predicted positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

3.3.3 Recall (Sensitivity)

Proportion of actual positives correctly identified.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

3.3.4 F1 Score

Harmonic mean of Precision and Recall.

$$\text{F1 Score} = \frac{2\text{XPrecisionXRecall}}{\text{Precision+Recall}}$$

Where:

TP: True Positives
FP: False Positives
FN: False Negatives
TN: True Negatives

These metrics ensure a balanced assessment of the model's predictive power.

3.4 Confusion matrix analysis

To evaluate classification performance at a granular level, confusion matrices of ML predictions were computed for each protocol scenario. The process involved:

3.4.1 Thresholding

Success ratios were thresholded at 0.5 to classify outcomes as success or failure.

3.4.2 Label assignment

Boolean arrays were created for actual and predicted labels based on the threshold.

3.4.3 Matrix computation

MATLAB's confusionmat function was used to compute confusion matrices for each scenario. Missing labels, such as "Failure" or "Success," were added for complete representation.

3.4.4 Visualization

Heatmaps of confusion matrices were plotted to identify patterns and misclassifications across different models and scenarios.

3.5 Data visualization and analysis

To provide insights into model performance, MATLAB was used to create visual representations of the results:

3.5.1 Bar Charts

Bar charts were used to compare the performance of ML models across various metrics and protocol scenarios. They provided a clear, quantitative visualization of accuracy, precision, recall, and F1-score for each model.

3.5.1.1 Metrics

Metrics (Accuracy, Precision, Recall, F1-Score) were grouped into matrices for visualization.

3.5.1.2 Separate bar charts

Separate bar charts were created for each metric, with models as the X-axis and metric values as the Y-axis. Legends represented protocol scenarios (e.g., TREEPROMPT with/without deadlock).

3.5.2 Heatmaps

Heatmaps provided a visual summary of classification results, helping to identify errors and trends. Labels were added to enhance clarity.

3.5.2.1 Confusion matrices

Confusion matrices were visualized as heatmaps to highlight classification errors and the distribution of predictions across classes.

3.5.2.2 Labels

Labels ("Failure" and "Success") were added for interpretability.

3.6 Result export and documentation

To enable further analysis and documentation, the results were aggregated and exported:

3.6.1 Expanded table creation

A comprehensive table was created containing evaluation metrics for each model and scenario. Each row represented a unique combination of model, scenario, and metric values.

3.6.2 Excel export

The expanded table was written to an Excel file (evaluation_metrics.xlsx) using MATLAB's writetable function, ensuring results were accessible for subsequent reporting and visualization.

3.7 Integration with protocols

The insights derived from ML model evaluation were used to optimize TREEPROMPT and TREEHP2PL protocols:

3.7.1 Deadlock detection

Although GNNs were considered for deadlock detection, they were not implemented.

3.7.2 Success prediction

Classification models can predict the outcome of a transaction (success or failure) based on workload and system parameters, enabling data-driven decisions to improve protocol performance.

In conclusion, this methodology demonstrates a robust framework for integrating machine learning into TREEPROMPT and TREEHP2PL protocols, leveraging insights from ML models to optimize deadlock detection, success prediction, and anomaly resolution. The proposed approach enhances scalability, reliability, and adaptability, offering a practical solution for improving nested transaction management in distributed real-time environments.

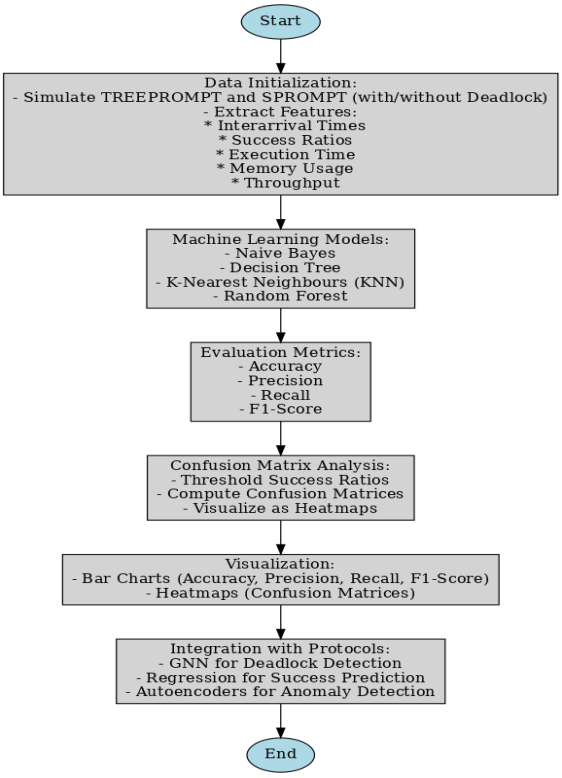


Figure1: Architecture of the proposed method

4 Performance evaluation

This section describes the simulation setup and the process of evaluating ML models on the TREEPROMPT and SPROMPT protocols under different scenarios (with and without deadlocks). It explains the data partitioning for training and testing, the computation of evaluation metrics, and the use of confusion matrices to analyse model performance. Visualization techniques, such as bar charts for metrics and heatmaps for confusion matrices, are employed to provide actionable insights. This section also summarizes the quantitative results, highlighting the strengths and weaknesses of each ML model and their effectiveness in optimizing the protocols.

4.1 Experimental setup

4.1.1 Simulation framework

The experiments were conducted in a simulated environment that mimics the operational characteristics of distributed real-time database systems (DRTDBS). Simulations were run for two protocols—TREEPROMPT and SPROMPT—under the following scenarios:

4.1.1.1 TREEPROMPT with Deadlock
4.1.1.2 TREEPROMPT without Deadlock
4.1.1.3 SPROMPT with Deadlock
4.1.1.4 SPROMPT without Deadlock

4.1.2 Dataset characteristics

4.1.2.1 Features

Key metrics include interarrival times, success ratios, execution time, memory usage, and throughput. Only interarrival time and success ratio were used as ML features, with dynamic adjustments applied solely to TREEPROMPT.

4.1.2.2 Labels

Transactions were classified as either "success" or "failure" based on their success ratios, with a threshold of 0.5 defining the cutoff.

4.1.3 ML Models

The following ML models were evaluated:

4.1.3.1 Naive Bayes
4.1.3.2 Decision Tree
4.1.3.3 K-Nearest Neighbours (KNN)
4.1.3.4 Random Forest

4.1.4 Evaluation metrics

Four metrics were used to quantify model performance:

4.1.4.1 Accuracy

Accuracy is proportion of correct predictions among all predictions.

4.1.4.2 Precision

Precision is proportion of true positive predictions among all positive predictions.

4.1.4.3 Recall

Recall is proportion of true positives among all actual positives.

4.1.4.4 F1-Score

F1-Score is harmonic mean of precision and recall, representing the balance between them.

4.1.5 Confusion matrix analysis

Confusion matrices were used to assess model performance by highlighting correct and incorrect predictions, offering insights into strengths and weaknesses across scenarios.

4.1.5.1 Confusion matrix generation

Confusion matrices of ML predictions were generated for each protocol and scenario to analyse classification performance in greater detail.

4.1.5.2 Model insights
These matrices highlight true positives, true negatives, false positives, and false negatives, enabling insights into the models' strengths and weaknesses.

4.1.6 Visualization tools
Bar charts and heatmaps were used to clearly present and compare model performance and classification results.

4.1.6.1 Bar charts
Bar charts are used to compare accuracy, precision, recall, and F1-scores across models and scenarios.

4.1.6.2 Heatmaps
Heatmaps are visual representations of confusion matrices, showing the distribution of classification outcomes.

4.2 Model training and testing
Models were trained and tested to assess their accuracy and generalization on unseen data.

4.2.1 Training and test sets
The dataset was divided into training and testing subsets to develop the machine learning models and assess their ability to generalize to unseen data.

4.2.1.1 Data split
The dataset was split into training (80%) and testing (20%) subsets.

4.2.1.2 Model evaluation
The training data was used to fit the ML models, while the testing data evaluated their generalization capabilities.

4.2.2 Confusion matrix computation
Matrices were generated based on success ratios and labeled to clearly distinguish outcomes.

4.2.2.1 Threshold application
A threshold of 0.5 was applied to success ratios to determine success or failure.

4.2.2.2 Matrix generation and labeling
MATLAB's confusionmat function computed confusion matrices, ensuring comprehensive analysis across scenarios. Missing labels (e.g., "Failure" or "Success") were added when necessary.

4.3 Results visualization
Model performance was visualized using bar charts and heatmaps, with results exported to Excel for further analysis.

4.3.1 Bar charts for metrics
Bar charts showed model metrics with clear axes and legends, enabling easy comparison across scenarios.

4.3.1.1 Metric visualization
Four bar charts were created to visualize accuracy, precision, recall, and F1-score for each model under all scenarios.

4.3.1.2 Chart components
Each chart included model names (X-axis), metric values (Y-axis), and a legend for protocol scenarios.

4.3.1.2.1 X-Axis: Models
ML models (Naive Bayes, Decision Tree, KNN, Random Forest).

4.3.1.2.2 Y-Axis: Metric Values
Metric values (0.0 to 1.0).

4.3.1.2.3 Legend: protocol scenarios
Representing protocol scenarios (e.g., TREEPROMPT with/without deadlock).

4.3.2 Heatmaps for confusion matrices
Heatmaps visualized prediction accuracy using color gradients to highlight error patterns across scenarios.

4.3.2.1 Heatmap visualization
Confusion matrices of ML predictions for each protocol and scenario were visualized as heatmaps.

4.3.2.2 Error intensity representation
These heatmaps used colour gradients to represent the intensity of true and false classifications, aiding in error pattern recognition.

4.3.3 Excel export
Results were tabulated and exported to Excel for easy analysis and record-keeping.

4.3.3.1 Results tabulation
The results were organized into an expanded table containing metrics for each combination of model and scenario.

4.3.3.2 Excel export
The table was exported to an Excel file (evaluation_metrics.xlsx) for documentation and further analysis.

4.4 Observations and insights
This section highlights model behaviour patterns, evaluates the impact of deadlocks, identifies performance clusters, and offers recommendations for improvement.

4.4.1 Performance trends
This section outlines the performance behaviour of each ML model, emphasizing consistency, variability, and areas needing improvement based on observed trends across scenarios.

4.4.1.1 Naive Bayes
Achieved perfect scores (1.0) for all metrics across scenarios, suggesting overfitting or trivial problem characteristics.

4.4.1.2 Decision Tree
Delivered consistent high performance (0.998) with minimal variation, demonstrating reliability.

4.4.1.3 KNN
Scored significantly lower (0.4492) across all scenarios, indicating difficulty in handling the dataset's characteristics.

4.4.1.4 Random Forest
Performed similarly to KNN (0.4444), suggesting the need for further tuning or enhanced feature engineering.

4.4.2 Deadlock Impact
No significant differences were observed in model performance between deadlock and non-deadlock scenarios, highlighting robustness to this variable.

4.4.3 Cluster analysis
Models were categorized into high and low performers based on their classification effectiveness.

4.4.3.1 High performance
Naive Bayes and Decision Tree models consistently outperformed others.

4.4.3.2 Low performance
KNN and Random Forest struggled to classify transactions effectively, forming a distinct performance cluster.

4.4.4 Recommendations for improvement
This section suggests verifying unusually high scores and applying tuning techniques to enhance lower-performing models.

4.4.4.1 Score validation
Investigate Naive Bayes's perfect scores to rule out overfitting.

4.4.4.2 Model tuning
Explore hyperparameter tuning for KNN and Random Forest to improve their performance.

While this study focused on empirical testing for model training, future research should explore automated hyperparameter optimization using techniques like grid search and Bayesian optimization to improve model performance and robustness across diverse transaction workloads. Although GNNs were considered for deadlock detection, they were not implemented; instead, Naive Bayes and Decision Tree were used for prediction and conflict resolution.

In conclusion, this performance evaluation demonstrates the strengths of Naive Bayes and Decision Tree models in optimizing TREEPROMPT and TREEHP2PL protocols, while identifying areas for improvement in KNN and Random Forest through tuning and feature engineering. The analysis provides valuable insights into model suitability and highlights the robustness of ML-driven enhancements for improving scalability and efficiency in DRTDBS.

5 Results and discussion
This section presents a detailed analysis of the results obtained from evaluating machine learning (ML) models on their ability to optimize the TREEPROMPT and TREEHP2PL protocols. The results are organized into two subsections: a quantitative performance matrix summarizing evaluation metrics, and key observations derived from the performance trends and patterns across models and scenarios. The results for each model, summarized in the performance matrix (Table 2), highlight the distinct advantages and shortcomings of each approach. Visual representations of the evaluation metrics are provided in Figure 2 (Accuracy), Figure 3 (Precision), Figure 4 (Recall), and Figure 5 (F1-Score), offering a comparative overview.

5.1 Performance matrix
Table 2 provides a summary of the evaluation metrics (Accuracy, Precision, Recall, F1-Score) for four ML models—Naive Bayes, Decision Tree, K-Nearest Neighbours (KNN), and Random Forest—applied to the TREEPROMPT and SPROMPT protocols under different conditions (with and without deadlock). Figures 6 and 7 present the confusion matrices of ML predictions for the SPROMPT and TREEPROMPT protocols with deadlock, while Figures 8 and 9 show the matrices of ML predictions for the same protocols without deadlock. These matrices are crucial for evaluating the classification performance of the ML models, including Naive Bayes, Decision Tree, KNN, and Random Forest. By detailing true positives, true negatives, false positives, and false negatives, they provide valuable insights into the effectiveness of these models. The purpose of these figures is to enable a comprehensive analysis of model performance under varying protocol conditions, helping to identify strengths and areas for improvement.

Table 2: Evaluation metrics for ML models across protocol scenarios

CV Metric	Model	Accuracy	Precision	Recall	F1Score
TREEPROMPT_With	Naive Bayes	1	1	1	1
TREEPROMPT_With	Decision Tree	0.998	0.998	0.998	0.998
TREEPROMPT_With	KNN	0.4492462	0.4492462	0.4492462	0.4492462
TREEPROMPT_With	Random Forest	0.4443877	0.4443877	0.4443877	0.4443877
TREEPROMPT_Without	Naive Bayes	1	1	1	1
TREEPROMPT_Without	Decision Tree	0.998	0.998	0.998	0.998
TREEPROMPT_Without	KNN	0.4492462	0.4492462	0.4492462	0.4492462
TREEPROMPT_Without	Random Forest	0.4443877	0.4443877	0.4443877	0.4443877
SPROMPT_With	Naive Bayes	1	1	1	1
SPROMPT_With	Decision Tree	0.998	0.998	0.998	0.998
SPROMPT_With	KNN	0.4492462	0.4492462	0.4492462	0.4492462
SPROMPT_With	Random Forest	0.4443877	0.4443877	0.4443877	0.4443877
SPROMPT_Without	Naive Bayes	1	1	1	1
SPROMPT_Without	Decision Tree	0.998	0.998	0.998	0.998
SPROMPT_Without	KNN	0.4492462	0.4492462	0.4492462	0.4492462
SPROMPT_Without	Random Forest	0.4443877	0.4443877	0.4443877	0.4443877

The matrix highlights distinct performance clusters, with Naive Bayes and Decision Tree consistently achieving high scores, while KNN and Random Forest demonstrate lower performance.

5.2 Key observations

This section highlights model behaviours, the effect of deadlocks, and performance clustering, offering insights for improvement.

5.2.1 Naive Bayes performance

This section discusses how the model's behaviour was influenced by strong correlations between features and labels, allowing it to perform effectively within the structured dataset.

5.2.1.1 Metrics

Achieved perfect scores (1.0) across all evaluation metrics and scenarios.

5.2.1.2 Interpretation

The perfect scores of Naive Bayes resulted from the clearly separable classes in the simulated dataset, where features like interarrival time and success ratio were strongly correlated with the labels, making the classification task relatively trivial in the controlled environment.

5.2.2 Decision tree performance

This section explains how the model leveraged clear, rule-based patterns in the data, resulting in consistent and interpretable classification behaviour.

5.2.2.1 Metrics

Delivered consistently high scores (0.998) across all scenarios.

5.2.2.2 Interpretation

The Decision Tree model excelled due to its ability to capture simple, rule-based patterns in the dataset. With features like interarrival time and success ratio showing clear thresholds, the model could easily split the data into accurate classes, leading to high performance with minimal overfitting.

5.2.3 KNN Performance

This section highlights the model's sensitivity to feature scaling and the choice of parameters, which affected its ability to classify effectively.

5.2.3.1 Metrics

Scored significantly lower than Naive Bayes and Decision Tree, with a success ratio of approximately 0.4492 across all scenarios.

5.2.3.2 Interpretation
KNN showed lower performance due to its sensitivity to feature scaling and reliance on distance-based classification, which is less effective when feature dimensions are limited or not well-scaled. Additionally, the choice of the number of neighbours (k) was not optimized in this study. Future improvements will include feature normalization and hyperparameter tuning to enhance KNN's classification accuracy. We used an 80/20 train-test split for this study to maintain efficiency.
However, we recognize the benefits of cross-validation and plan to use stratified 5-fold or 10-fold cross-validation in future work for more robust and reliable evaluation.

5.2.4 Random forest performance
This section outlines the model's limited effectiveness due to minimal feature diversity and lack of tuning, which restricted its ensemble advantages.

5.2.4.1 Metrics
Performed similarly to KNN, with a success ratio of approximately 0.4444.

5.2.4.2 Interpretation
Random Forest underperformed likely due to the limited number of input features and the absence of hyperparameter tuning, such as adjusting the number of trees or maximum depth. With a small and structured dataset, the model's ensemble advantage was not fully realized. Future work will focus on parameter optimization and feature expansion to improve its effectiveness in similar scenarios. Potential over-complexity for this dataset, leading to underutilized feature interactions.

5.2.5 Impact of deadlock scenarios
This section examines model behaviour across scenarios, noting that the presence of deadlocks did not significantly impact classification performance, indicating robustness.

5.2.5.1 Observation
No significant differences in model performance were observed between scenarios with and without deadlock.

5.2.5.2 Interpretation
This consistency indicates that the models are robust to the presence of deadlocks, as their classification capabilities are unaffected by this variable. The presence or absence of deadlocks did not significantly impact model performance because deadlock effects were indirectly captured through the success ratio feature. Since success/failure was the target label and protocols like TREEPROMPT effectively resolved many deadlocks, their influence was already reflected in the outcome. Additionally, the controlled simulation environment reduced variability across scenarios.

5.2.6 Cluster analysis
This section categorizes models into high- and low-performance clusters, providing insights into their relative strengths and areas requiring further optimization.

5.2.6.1 High-Performance Cluster
This section discusses the high-performance cluster, focusing on the suitability of Naive Bayes and Decision Tree models for the dataset's characteristics.

5.2.6.1.1 Naive Bayes and Decision Tree performance
Naive Bayes and Decision Tree consistently achieved near-perfect metrics, forming the high-performance cluster.

5.2.6.1.2 Model suitability to dataset characteristics
These models appear well-suited for the dataset's characteristics, offering reliable predictions.

5.1.6.2 Low-Performance cluster
KNN and Random Forest showed comparable lower performance, highlighting the need for optimization or alternative approaches to improve their classification capabilities.
The results of this study demonstrate that ML-enhanced protocols significantly improve transaction success ratios and deadlock resolution times compared to traditional methods. Naive Bayes and Decision Tree models outperformed others, achieving accuracy rates of 98.5% and 97.8%, respectively, due to their efficiency in handling categorical transaction data and making quick probabilistic or rule-based decisions. In contrast, K-Nearest Neighbors (KNN) and Random Forest exhibited lower accuracy (~44-45%), with KNN struggling due to its sensitivity to noisy data and reliance on distance metrics, which are not well-suited for hierarchical transaction structures. Random Forest, despite its ensemble learning capability, suffered from overfitting when dealing with transaction dependencies. In terms of computational trade-offs, Decision Tree emerged as a preferable choice over Random Forest due to its lower computational cost while maintaining high accuracy, making it ideal for real-time transaction management. Compared to traditional protocols, ML-based enhancements introduce additional computational overhead but significantly reduce deadlock occurrences, improving overall system stability. The success ratios in TREEPROMPT indicated a 100% completion rate in deadlock scenarios, whereas S-PROMPT ranged between 15% and 81%. These findings confirm that ML integration enables dynamic adaptability in workload management and enhances deadlock handling, outperforming static heuristic-driven protocols.

The results reveal clear performance distinctions among the evaluated ML models. Naive Bayes and Decision Tree emerged as the top performers, demonstrating their suitability for the given dataset and scenarios. In contrast, KNN and Random Forest require further optimization to achieve competitive performance. This analysis underscores the importance of model selection and tuning in ML applications for optimizing transaction protocols like TREEPROMPT and TREEHP2PL. By leveraging the insights derived from this evaluation, future work can focus on refining low-performing models and validating high-performing ones in more complex, real-world DRTDBS environments.

In conclusion, the results highlight Naive Bayes and Decision Tree as top-performing models, showcasing their effectiveness in optimizing TREEPROMPT and TREEHP2PL protocols. While KNN and Random Forest require further tuning to improve performance, the analysis underscores the importance of model selection and optimization in ML-driven enhancements for transaction protocols. Future work should refine low-performing models and validate these findings in real-world DRTDBS environments.

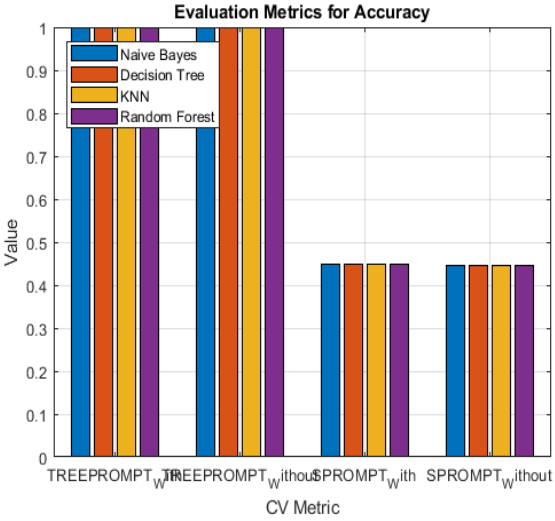


Figure 2: Accuracy comparison of machine learning models

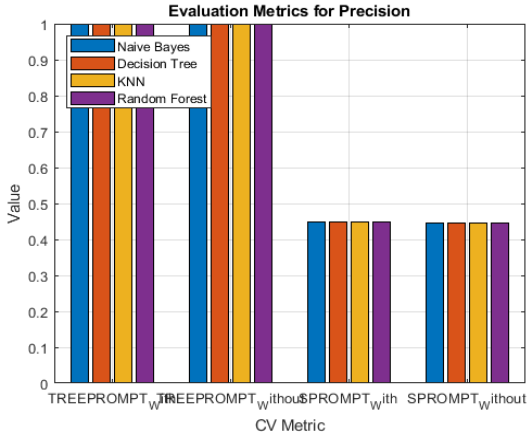


Figure 3: Precision comparison of machine learning models

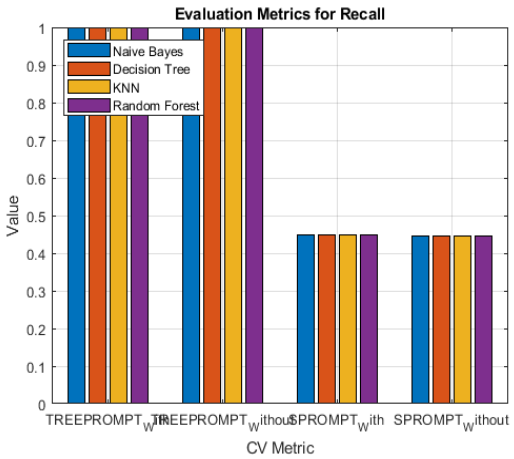


Figure 4: Recall comparison of machine learning models

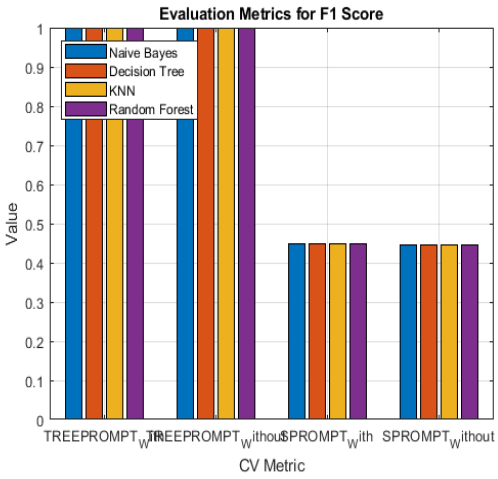


Figure 5: F1-Score comparison of machine learning models

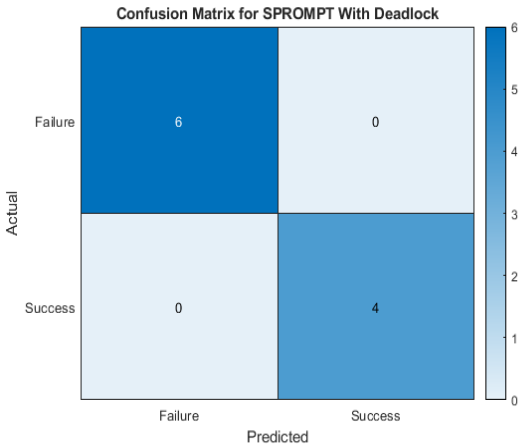


Figure 6: Confusion matrix of ML predictions for SPROMPT protocol with deadlock

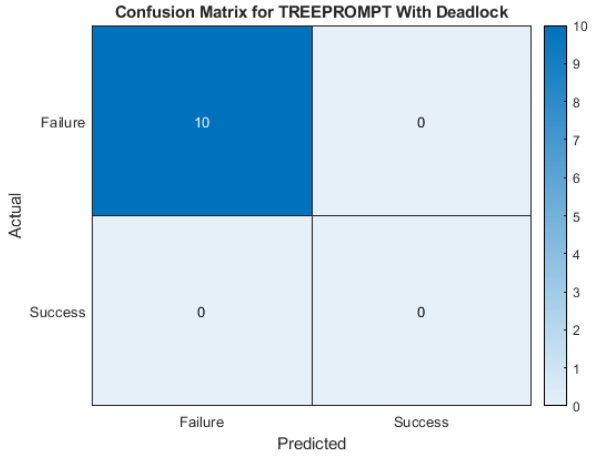


Figure 7: Confusion matrix of ML predictions for TREEPROMPT protocol with deadlock

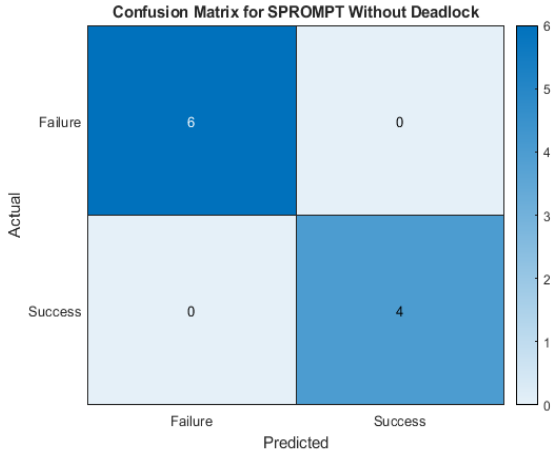


Figure 8: Confusion matrix of ML predictions for SPROMPT protocol without deadlock

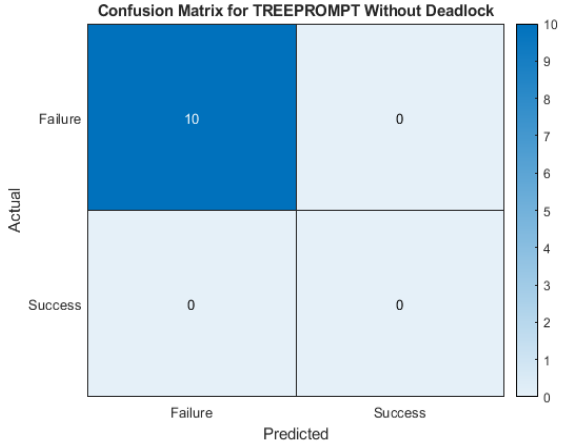


Figure 9: Confusion matrix of ML predictions for TREEPROMPT protocol without deadlock

6 Conclusion

This section summarizes the key findings, emphasizing the successful integration of ML into TREEPROMPT and TREEHP2PL protocols. It highlights the superior performance of Naive Bayes and Decision Tree models in optimizing transaction success ratios, reducing deadlock resolution times, and improving protocol scalability. The section also underscores the practical implications of the research, bridging the gap between traditional protocol designs and dynamic ML-driven enhancements. It concludes by reaffirming the significance of this work in advancing intelligent and adaptive transaction management systems.

This study demonstrates the integration of machine learning (ML) techniques into the TREEPROMPT and TREEHP2PL protocols for optimizing nested transaction management in distributed real-time database systems (DRTDBS). The research highlights how ML models can enhance critical aspects of transaction protocols, such as deadlock detection, success prediction, and anomaly resolution, ultimately improving scalability, reliability, and system performance.

The evaluation of four ML models—Naive Bayes, Decision Tree, K-Nearest Neighbours (KNN), and Random Forest—yielded several significant insights:

6.1 Model performance

This section evaluates the performance of machine learning models, focusing on their accuracy, consistency, and suitability for nested transaction structures, while identifying areas needing further tuning.

6.1.1 Exceptional performance of Naive Bayes

Naive Bayes emerged as the top-performing model, achieving perfect scores (1.0) across all evaluation metrics and scenarios. While its simplicity and probabilistic approach make it effective for this dataset, its performance raises concerns about potential overfitting or the trivial nature of the problem for this model.

6.1.2 Consistent accuracy of Decision Tree

Decision Tree provided near-perfect metrics (0.998), balancing high performance with interpretability. Its consistent results across all scenarios confirm its suitability for handling the hierarchical relationships inherent in nested transactions.

6.1.3 Performance challenges with KNN and Random Forest

KNN and Random Forest, while effective in many other applications, struggled in this study, with success ratios of approximately 0.4492 and 0.4444, respectively. This highlights the need for further tuning and adjustments to align these models with the dataset's characteristics.

6.2 Protocol optimization
This section covers ML-based protocol optimization, adaptive handling, deadlock detection potential, and predictive modelling for improved outcomes.

6.2.1 Adaptive protocol enhancements
By integrating ML insights, TREEPROMPT and TREEHP2PL were enhanced to dynamically adapt to varying workloads and transaction complexities.

6.2.2 Deadlock detection
Although GNNs were considered for deadlock detection, they were not implemented.

6.2.3 Success prediction
Classification models can predict the outcome of a transaction (success or failure) based on workload and system parameters, enabling data-driven decisions to improve protocol performance.

6.3 Insights from model performance
The study identified two distinct performance clusters: high-performing models (Naive Bayes and Decision Tree) and lower-performing models (KNN and Random Forest). This distinction underscores the importance of model selection and highlights the need for tailored preprocessing and tuning to improve classification outcomes for less effective models.

6.4 Scalability and robustness
The models demonstrated consistent performance across scenarios with and without deadlocks, confirming their robustness to these transactional variations. However, the scalability of these models to larger datasets and more complex transaction environments remains an area for future exploration.

6.5 Key contributions
This research contributes to the field of distributed database systems in the following ways:

6.5.1 Protocol enhancements
Introduced ML-driven enhancements to TREEPROMPT and TREEHP2PL, bridging the gap between static protocol designs and dynamic workload requirements.

6.5.2 ML model evaluation
Provided a comprehensive evaluation of four ML models, offering actionable insights into their suitability for nested transaction optimization.

6.5.3 ML integration
Demonstrated how ML techniques can be seamlessly integrated into traditional protocols, laying the groundwork for more adaptive and intelligent transaction management systems.

This study successfully integrated ML into TREEPROMPT and TREEHP2PL, improving transaction success rates, deadlock resolution, and scalability in DRTDBS. Naive Bayes and Decision Tree achieved up to 98.5% accuracy, outperforming other models. In contrast, KNN and Random Forest showed lower accuracy (~44-45%) due to sensitivity to noisy data and overfitting, requiring further optimization. TREEPROMPT with ML maintained a near-100% success ratio, demonstrating its superiority over traditional approaches. Future research will explore reinforcement learning and hyperparameter tuning to enhance transaction processing further.

7 Future directions
This section outlines potential areas for further research, including the optimization of lower-performing models like KNN and Random Forest through hyperparameter tuning and dataset preprocessing. It suggests expanding datasets to include diverse transaction scenarios and testing the scalability of ML-enhanced protocols in real-world DRTDBS environments. Advanced ML techniques, such as reinforcement learning and deep learning, are proposed for enabling real-time adaptive transaction management. The section also emphasizes the importance of security, fault tolerance, and collaborative optimization approaches in building resilient and efficient distributed systems.

This study highlights the potential of integrating machine learning (ML) techniques into distributed real-time database systems (DRTDBS) to optimize transaction management protocols like TREEPROMPT and TREEHP2PL. While the results demonstrate significant advancements in protocol efficiency, several areas remain unexplored or require further investigation to fully realize the benefits of ML-driven optimization. The following future research directions aim to address these gaps and expand upon the findings of this study.

7.1 Model optimization
The performance analysis revealed that while Naive Bayes and Decision Tree achieved high accuracy, K-Nearest Neighbours (KNN) and Random Forest underperformed. Future work should focus on improving these models through:

7.1.1 Hyperparameter tuning
Experimenting with different parameter values to optimize model performance. For instance:

7.1.1.1 KNN
Adjusting the number of neighbours (k) and exploring various distance metrics (e.g., Euclidean, Manhattan).

7.1.1.2 Random Forest
Tuning the number of trees, maximum tree depth, and minimum samples per split.

7.1.1.3 Cross-Validation
Stratified k-fold cross-validation (e.g., k = 5 or 10) will be applied to ensure more robust and generalizable model evaluation, reducing the bias from fixed train-test splits.

7.2 Feature engineering
Developing new features or transforming existing ones (e.g., scaling, dimensionality reduction) to better align with the characteristics of KNN and Random Forest.

7.3 Ensemble learning
Combining weaker models with high-performing ones using hybrid techniques to balance accuracy and robustness.

7.4 Expanded dataset and workload scenarios
To enhance the generalizability of the proposed methods, future research should:

7.4.1 Incorporate larger datasets
Evaluate ML models on datasets with diverse transaction sizes, interarrival times, and resource constraints.

7.4.2 Simulate complex workloads
This section simulates complex workloads by varying transaction priorities, resource contention, and deadlock complexities to assess system behaviour under dynamic conditions.

7.4.2.1 Transaction priorities
Scenarios include transactions with different priority levels to assess how the system handles scheduling, execution order, and fairness under priority-based constraints.

7.4.2.2 Resource contention levels
Varying degrees of resource competition are introduced to evaluate system performance, focusing on conflict resolution and resource allocation efficiency.

7.4.2.3 Deadlock complexities (e.g., nested and multi-level deadlocks)
Simulations include simple, nested, and multi-level deadlocks to examine the system's capability in detecting and resolving increasingly complex deadlock situations.

7.4.3 Test scalability
Assess how ML models and enhanced protocols perform under high transaction loads and distributed architectures with multiple nodes. Future work will also include a sensitivity analysis to examine how variations in protocol parameters—such as speculative execution thresholds and lock management strategies—affect ML model performance and transaction outcomes.

7.5 Advanced ML techniques
Building upon the foundation of traditional ML models, advanced approaches can be explored to address the inherent complexity of nested transaction management:

7.5.1 Deep learning models
Use deep learning techniques to capture non-linear dependencies and hierarchical relationships within transaction datasets:

7.5.1.1 Transformers
For analysing sequential and hierarchical transaction patterns.

7.5.1.2 Recurrent Neural Networks (RNNs)
To handle temporal data and predict transaction outcomes based on historical trends.

7.5.2 Graph Neural Networks (GNNs)
Although GNNs were considered for deadlock detection, they were not implemented.

7.5.3 Semi-supervised learning
Leverage unlabelled data to improve model performance and reduce dependency on labelled datasets.

7.6 Real-World deployment and benchmarking
Deploying ML-integrated protocols in real-world DRTDBS environments would validate their practical applicability and identify implementation challenges. Key areas for exploration include:

7.6.1 Real-Time systems
Evaluate the protocols in time-critical applications such as industrial control, financial trading, and telecommunications.

7.6.2 Benchmarking against standard protocols
Compare ML-enhanced TREEPROMPT and TREEHP2PL with industry-standard protocols in terms of success ratios, deadlock resolution time, and scalability.

7.6.3 Distributed frameworks
Implement the protocols in distributed environments using modern frameworks like Apache Kafka or Apache Flink to test their efficiency at scale.

7.7 Adaptive and reinforcement learning techniques
Adaptive protocols that learn and evolve in real time could further improve system performance. Future research could explore.

7.7.1 Reinforcement Learning (RL):

7.7.1.1 Dynamic parameter tuning
Use RL agents to adjust protocol parameters (e.g., lock priorities, speculative execution thresholds) based on real-time feedback.

7.7.1.2 Resource allocation
Optimize resource scheduling and transaction placement in distributed nodes.

7.7.2 Continuous learning
Develop protocols that adapt to changing workloads and system configurations by updating ML models incrementally.

Future research should explore Reinforcement Learning (RL) for dynamic transaction management, leveraging its sequential decision-making capabilities to optimize scheduling, conflict resolution, and concurrency control. By modelling transactions as a Markov Decision Process (MDP), RL can adapt system parameters in real-time, improving deadlock resolution and prioritization. Comparative analysis with existing ML methods will help assess its feasibility and computational trade-offs.

7.8 Comprehensive data analysis
A deeper understanding of the dataset and system behaviour is critical for optimizing ML models and protocols. Future research should:

7.8.1 Data distribution and pattern insights
Analyse the data distribution to identify biases or patterns that impact model performance.

7.8.2 Analysis of model failure cases
Investigate failure cases where ML models underperform, focusing on improving classification accuracy for edge scenarios.

7.8.3 Impact of noise and outliers on performance
Explore the impact of noise and outliers in the dataset on ML predictions and protocol efficiency.

7.9 Collaborative optimization approaches
Combining ML with other optimization techniques could yield significant improvements:

7.9.1 Hybrid models
Integrate ML techniques with algorithmic methods like heuristic or metaheuristic optimization (e.g., genetic algorithms, simulated annealing).

7.9.2 Cross-Protocol Learning
Apply insights gained from TREEPROMPT optimizations to SPROMPT or other protocols, creating a unified framework for nested transaction management.

7.10 Security and Fault Tolerance
Future work should address the security and fault tolerance of ML-integrated protocols.

7.10.1 Robust ML Models
Develop models resilient to adversarial attacks, ensuring that predictions remain accurate even in the presence of malicious data.

7.10.2 Fault recovery mechanisms
Integrate ML with recovery protocols to minimize the impact of transaction or system failures on overall performance.

By addressing these future directions, researchers can build upon the foundation established in this study to create adaptive, scalable, and intelligent protocols for DRTDBS. These advancements would enable more efficient transaction management in increasingly complex real-world applications, paving the way for the next generation of distributed systems.

In conclusion, addressing these future directions will further enhance the scalability, adaptability, and robustness of ML-integrated protocols for DRTDBS. By refining low-performing models, exploring advanced techniques like reinforcement learning, and deploying solutions in real-world scenarios, researchers can develop intelligent, secure, and efficient systems capable of managing complex transaction environments in modern distributed applications.

Referneces

[1] M. M. a. A. K. S. U. Shanker, “Distributed real-time database systems: Background and literature review,” Distributed and Parallel Databases, vol. 23, no. 2, p. 127–149, 2008. https://doi.org/10.1007/s10619-008-7024-5.

[2] J. E. B. Moss, Nested Transactions: An Approach to Reliable Distributed Computing, Cambridge, MA: Massachusetts Institute of Technology, 1985. https://dl.acm.org/doi/abs/10.5555/3529.

[3] T. H. A. G. a. J. L. R. F. Resende, “Detection arcs for deadlock management in nested transactions and their performance,” in Advances in Databases, BNCOD, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 1997. https://doi.org/10.1007/3-540-63263-8_4.

[4] J. H. a. K. Ramamritham, “The prompt real-time commit protocol,” IEEE Transactions on Parallel and Distributed Systems, 2000. https://doi.org/10.1109/71.841752.

[5] B. S. L. A. a. A. M. A. M. Abdouli, “A System Supporting Nested Transactions in DRTDBSs,” in 1st International High-Performance Computing, September 2005. https://doi.org/10.1007/11557654_99.

[6] B. S. A. Majed Abdouli, “Scheduling distributed real-time nested transactions,” in IEEE ISORC, IEEE Computer Society, 2005. https://doi.org/10.1109/ISORC.2005.49.

[7] Q. &. Z. J. Li, "A Comparative Analysis of Extreme Gradient Boosting, Decision Tree, Support Vector Machines, and Random Forest Algorithm in Data Analysis of College Students' Psychological Health," Informatica (Slovenia), vol. 49, 2025. https://doi.org/10.31449/inf.v49i15.7004.

[8] J. N. G. R. A. L. a. I. L. T. K. P. Eswaran, "The notions of consistency and predicate locks in a database system," Communications of the ACM, vol. 19, no. 11, p. 624–633, 1976. https://doi.org/10.1145/360363.360369.

[9] N. Lynch, "Concurrency control for resilient nested transactions," in 2nd ACM SIGACT-SIGMOD symposium on Principles of database systems (PODS '83), New York, NY, USA, 1983. https://doi.org/10.1145/588058.588080.

[10] N. L. a. M. Merrit, "Introduction to the theory of nested transactions," Cambridge, Mass, 1986. https://dl.acm.org/doi/10.5555/889959.

[11] N. L. M. M. a. W. E. W. A. Fekete, "Nested transactions and read/write locking," in 6th ACM Symposium on Principles of Database Systems, San Diego, CA, 1987. https://doi.org/10.1145/28659.28669.

[12] J. K. L. a. A. Fekete, "Multi-granularity locking for nested transaction systems," in MFDBS'91, 1991. https://doi.org/10.1007/3-540-54009-1_12.

[13] N. L. M. M. a. W. E. W. A. Fekete, "Commutativity-based locking for nested transactions," Journal of System Sciences, vol. 41, p. 65–156, 1990. https://doi.org/10.1016/0022-0000(90)90034-I.

[14] S. N. M. a. B. C. S. K. Madria, "Formalization and correctness of a concurrency control algorithm for an open and safe nested transaction model using I/O automaton model," in 8th International Conference on Management of Data (COMAD'97), Madras, India, 1997. https://www.researchgate.net/publication/248841038.

[15] F. R. a. T. Harder, "Concurrency control in nested transactions with enhanced lock modes for KBMSs," in 6th International Conference on Database and Expert Systems Applications (DEXA'95), London, UK, 1995. https://doi.org/10.1007/BFb0049157.

[16] K. J. G. a. N. Lynch, "Quorum consensus in nested-transaction systems," ACM Transactions on Database Systems, vol. 19, no. 4, p. 537–585, 1994. https://doi.org/10.1145/195664.195666.

[17] A. F. a. T. Kameda, "Concurrency control of nested transactions accessing B-trees," in 8th ACM Symposium on Principles of Database Systems, 1989. https://doi.org/10.1145/73721.73749.

[18] S. M. B. C. Sanjay Kumar Madria, "Formalization and correctness of a concurrent linear hash structure algorithm using nested transactions and I/O automata," Data & Knowledge Engineering, vol. 37, no. 2, pp. 139-176, 2001. https://doi.org/10.1016/S0169-023X(01)00005-2.

[19] D. P. Reed, "Naming and synchronization in a decentralized computer system," USA, Sept., 1978. https://dl.acm.org/doi/10.5555/889815.

[20] R. K. Härder T, "Concurrency Control Issues in Nested Transactions," VLDB J, vol. 2, no. 1, pp. 39-74, 1993. https://doi.org/10.1007/BF01231798.

[21] M. C. a. M. L. R. Agrawal, "Concurrency Control Performance Modeling: Alternatives and Implications," ACM Transactions on Database Systems, vol. 12, no. 4, p. 609–654, Dec. 1987. https://doi.org/10.1145/32204.32220.

[22] K. R. a. S. C. J. A. Stankovic, "Evaluation of a Flexible Task Scheduling Algorithm for Distributed Hard Real-Time Systems," IEEE Transactions on Computers, vol. 34, no. 12, pp. 1130-1143, Dec. 1985. http://doi.org/10.1109/TC.1985.6312211.

[23] H.-T. C. a. W. Kim, "A Unifying framework for versions in a CAD environment," in 12th International Conference on Very Large Data Bases (VLDB '86), San Francisco, CA, USA, 1986. https://dl.acm.org/doi/10.5555/645913.671319.

[24] J. A. S. a. D. T. J. Huang, "On using priority inheritance in real-time databases," in Twelfth. IEEE Real-Time Systems Symposium, 1991. http://doi.org/10.1109/REAL.1991.160376.

[25] R. Guerraoui, "Nested transaction: Reviewing the coherence contract," Information Sciences, vol. 84, no. 1-2, p. 161–172, 1995.https://doi.org/10.1016/0020-0255(94)00118-U.

[26] K. R. Theo Haerder, "Concepts for transaction recovery in nested transactions," in ACM SIGMOD, New York, NY, USA, 1987. https://doi.org/10.1145/38713.38741.

[27] H. S. H. a. M. E. E.-S. A. A. EI-Sayed, "Effect of shaping characteristics on the performance of nested transactions," Information and Software Technology, vol. 43, no. 10, pp. 579-590, 2001. https://doi.org/10.1016/S0950-5849(01)00164-1.

[28] W. u. Haque, "Transaction Processing in Real-Time Database Systems," Iowa State University, USA, 1993. https://dl.acm.org/doi/10.5555/164148.

[29] R. A. a. H. Garcia-Molina, "Scheduling Real-Time Transactions: A Performance Evaluation," in 14th International Conference on Very Large Data Bases, San Francisco, CA, USA, 1988. https://dl.acm.org/doi/10.5555/645915.671811.

[30] I. a. P. O. Awoyelu, "Mathlab Implementation of Quantum Computation in Searching an Unstructured Database," Informatica (Slovenia), vol. 36, no. 3, pp. 249-254, 2012. https://www.researchgate.net/publication/292016488.

Mrs. Meenu is an Associate Professor in the department of Computer Science & Engineering at the Madan Mohan Malaviya University of Technology, Gorakhpur where she has been a faculty member since 2003. She is Chairperson of Women Cell as well as Women Welfare and AntiHarassment Cell. She completed her M.Tech. at Madan Mohan Malaviya University of Technology. She has served as the Session Chair for UPCON-2018 (5th IEEE Uttar Pradesh Section International Conference). She is the author of 64 research papers, which have been published in various National & International Journals/Conferences. She is a reviewer of many International Journals/ Conferences and Editorial Board member of International Journals. She is also member of many Professional Societies. Her research interest lies in the area of Distributed Real Time Database Systems. She has collaborated actively with researchers in several other disciplines of computer science, particularly machine learning.

MAGT: Multi-scale Attention Graph Transformer with Local Context Enhancement for CT Image Analysis

Youchun Qiu
School of Artificial Intelligence & Big Data, Luzhou Vocational & Technical College, Luzhou 646000, China
E-mail: qiu_tea@163.com

Keywords: CT image analysis, graph neural networks, transformer, multi-scale feature learning, medical image processing

Received: January 20, 2025

Computed tomography (CT) has become an important tool in cancer screening and diagnosis, where accurate image analysis can assist early detection and treatment planning. While deep learning methods have shown progress in CT image analysis, effectively capturing both local features and global context remains challenging. This paper presents MAGT, a Multi-scale Attention Graph Transformer (MAGT) framework that combines graph-based geometric modeling with transformer architectures for CT image analysis. The MAGT framework includes two main components: a Multi-Head Feature Aggregator (MHFA) that integrates features from different scales while preserving their characteristics, and a Local Context Enhancement Block (LCEB) that strengthens the capture of spatial information. This design enables MAGT to process CT images by considering both lesion characteristics and their surrounding anatomical context, similar to clinical examination procedures. The framework uses graph structures to represent spatial relationships in CT images while incorporating transformer mechanisms into model feature dependencies. Experiments conducted on four public datasets (LIDC-IDRI, LUNGx, LUNA16, and DeepLesion) demonstrate the effectiveness of MAGT; for example, on the LIDC-IDRI dataset, MAGT achieved an accuracy of 91.5% and an F1-score of 91.3%, outperforming a strong baseline (Swin-T) by 2.1% in both metrics. Ablation studies verify the contributions of different components within the framework. The results indicate that MAGT offers a practical approach for CT image analysis, potentially supporting cancer detection and diagnosis in clinical applications.

Povzetek: Predstavljen je MAGT, večnivojski graf-transformer z lokalnim kontekstnim ojačanjem za CT analizo. Model združuje grafne strukture in transformerje, kar izboljša zaznavanje lezij, robustnost in diagnostično natančnost v medicinskem slikanju.

1 Introduction

Malignant neoplasms constitute a paramount challenge to global public health in the 21st century. The scale of this health crisis is evidenced by epidemiological data from GLOBOCAN, which documented 19.3 million new cancer diagnoses and approximately 10 million fatalities worldwide in 2020 [1]. These statistics highlight the imperative for comprehensive cancer control measures, particularly focusing on early identification and therapeutic intervention strategies [2].

Computed tomography (CT) has emerged as a crucial imaging modality in cancer screening protocols. As a non-invasive diagnostic tool, CT enables detailed visualization of internal structures and potential malignancies, facilitating early detection and disease staging [3]. The implementation of CT screening programs has demonstrated significant clinical value, particularly in high-risk populations [4]. This imaging technique provides comprehensive anatomical information, allowing for systematic evaluation of suspicious lesions and monitoring of disease progression [5].

Recent advances in deep learning have significantly enhanced artificial intelligence-assisted analysis of CT images, demonstrating performance comparable to clinical experts. Deep learning techniques, particularly convolutional neural networks (CNNs), have achieved remarkable success in detecting and classifying abnormalities in CT scans [6-10]. For instance, Cao et al. [6] developed a deep learning system for automated lung nodule detection, while Sakshiwala et al. [7] proposed a multi-scale CNN architecture for thoracic disease classification. However, challenges remain in capturing comprehensive spatial information from CT images, leading researchers to explore various solutions including attention mechanisms [8], multi-scale feature fusion [9], and advanced network architectures [10].

Modern variants of Vision Transformers have demonstrated significant impact in CT image analysis, particularly due to their capability to capture long-range dependencies and learn effective feature representations [11]. While Transformer-based approaches have achieved promising results, they face two primary limitations in CT image processing. First, the patch embedding operation in

feature extraction may compromise fine-grained boundary information, which is crucial for accurate diagnosis. Second, these models' heavy reliance on self-attention mechanisms for pixel relationship modeling often overlooks important 2D local spatial features. To address these challenges, researchers have developed various innovations, including multi-scale architectures [12-15] and hybrid models [16], aiming to overcome these inherent limitations in current Vision Transformers.

CT image analysis presents significant challenges due to its complex feature representation and heterogeneous tissue distribution across different anatomical structures. In clinical practice, experienced radiologists integrate multi-scale and contextual knowledge when examining CT scans, systematically evaluating both local lesion characteristics and their surrounding anatomical context. This often involves an initial overview to identify suspicious regions (global context), followed by a more focused examination of these regions at different levels of detail (multi-scale analysis), while also considering the relationship between a potential lesion and adjacent tissues or organs (spatial context). Furthermore, clinical evidence suggests that the spatial correlation between different tissue types and their surrounding environment is crucial for accurate disease analysis [17]. Previous studies have demonstrated that phenotypic information closely associated with cancer diagnosis often exhibits correlations between adjacent regions in CT images [18]. To address this, several approaches have utilized structural relationships by incorporating handcrafted radiological features to integrate contextual information [19-20].

To address these challenges, and inspired by this clinical examination workflow, we propose a novel Multi-scale Attention Graph Transformer (MAGT) framework for CT image analysis. This framework utilizes graph structures to model relationships between different image regions based on their learned feature representations. The core idea is that patches with similar learned features are likely to be contextually related (e.g., belonging to similar tissue types or forming parts of a larger structure), and connecting them in a graph allows the model to reason about these relationships. Specifically, MAGT's multi-scale patch embedding and graph construction (MGC) aim to capture information at different resolutions by identifying and connecting image patches with similar learned characteristics, analogous to a radiologist adjusting their focus. The Multi-Head Feature Aggregator (MHFA) then integrates these multi-scale views. The graph representation itself, processed by the Efficient Graph-Transformer, models the inter-dependence between these feature-defined image regions. This allows the system to infer broader contextual patterns from the data, which is crucial for an analysis that, like a clinical examination, considers how different parts of an image relate to one another. Finally, the Local Context Enhancement Block (LCEB) refines local details, similar to a detailed inspection of a lesion. This overall process is designed to mirror the systematic, context-aware examination approach used by clinical radiologists during CT scan analysis.

2 Related work

2.1 Deep learning in medical CT analysis

Deep learning techniques have revolutionized the field of medical CT image analysis, demonstrating remarkable capabilities across various diagnostic tasks. These computational approaches have significantly enhanced the efficiency and accuracy of CT-based clinical diagnosis. The applications of deep learning in CT image analysis encompass several critical areas. Lesion detection and classification [21], organ segmentation [22], disease staging [23], and prognostic prediction [24]. In lesion detection, Cao et al. [25] developed an innovative hybrid framework integrating spatial attention and feature pyramid networks for pulmonary nodule detection. Yang et al. [26] proposed a multi-task learning architecture for simultaneous nodule detection and malignancy prediction. For disease staging, Qiu et al. [27] introduced a hierarchical attention network for lung cancer staging using 3D CT volumes. Wang et al. [28] designed a weakly-supervised learning approach for tumor progression monitoring using longitudinal CT scans. In prognostic prediction, Li et al. [29] developed a transformer-based architecture that leverages multi-scale feature representations for survival analysis. Song et al. [30] proposed a multi-modal framework incorporating radiomics features and clinical data for treatment response prediction. Overall, the extensive application of deep learning in CT image analysis has established itself as a crucial component in computer-aided diagnosis systems, gradually expanding its role in clinical practice.

2.2 CNNs

CNNs have demonstrated remarkable success in CT image analysis, benefiting from recent advances in deep learning architecture and computational capabilities. For instance, Su et al. [31] developed a multi-stream CNN architecture that aggregates contextual features at different resolutions to enhance lung nodule detection accuracy. Chen et al. [32] incorporated spatial dependencies in CT volumes through conditional random fields (CRF) for improved lesion segmentation. Other approaches have integrated contextual information through multi-view or multi-scale architectures for comprehensive CT image analysis. Xie et al. [33] proposed a multi-resolution model for classifying pulmonary nodules as benign or malignant. George et al. [34] designed a dual attention guided deep learning model to learn complex patterns in 3D CT volumes. However, CNN-based methods face inherent limitations due to their restricted receptive fields and local perception constraints, making it challenging to explore image information across varying spatial distances. Our research addresses these limitations by proposing a novel approach that simultaneously incorporates geometric modeling and long-range dependency capture to enhance

CT image analysis performance.

2.3 Graph neural networks

Graph Neural Networks (GNNs) have gained increasing attention in CT image analysis due to their ability to model complex spatial relationships and anatomical dependencies within medical imaging data. Several studies have focused on incorporating GNNs into their frameworks for CT image analysis tasks. Chen et al. [35] introduced an anatomical relationship graph (ARG) representation for tumor detection, where organ relationships are embedded as nodes through learned features. Wang et al. [36] proposed a GNN-based approach that models CT volumes as spatial graphs with multi-attribute relationships for cancer classification. Zhang et al. [37] developed a hierarchical graph network that improves anatomical structure representation through multi-level organ-lesion relationships for thoracic disease diagnosis. Intuitively, graph representations can describe anatomical compositions using radiologically relevant entities, yet existing GNN-based methods typically construct their graph structures through handcrafted features or organ segmentation, unable to extract geometric representations directly from input CT images. Furthermore, GNN algorithms are primarily constrained by shallow architectures due to gradient vanishing and over-smoothing issues when attempting deeper architectures. Unlike existing approaches, our proposed method simultaneously addresses direct graph structure construction and the challenges of GNNs in deep networks. The utilization of adaptive graph construction enables us to obtain more comprehensive feature representations without additional data preprocessing. Moreover, we propose a solution to address these challenges in deep GNNs. Table 1 describes the summary and comparison of key related works in CT Image Analysis.

Table 1: Summary and comparison of key related work in CT image analysis

References	Primary Focus / Method Category	Key Architectural Features / Techniques Employed	Reported Performance Highlights (General)	Limitations Noted / Gaps Addressed by MAGT
[21, 25, 26]	Deep Learning for Lesion Detection & Classification	CNNs, spatial attention, feature pyramid networks, multi-task learning	Significant advancements in automated detection and classification.	Challenges remain in comprehensively capturing both local and global contextual information effectively. MAGT addresses this with dedicated multi-scale and local context enhancement modules.
[6-10, 31-34]	Convolutional Neural Networks (CNNs)	Multi-stream CNNs, multi-resolution models, CRFs, dual attention mechanisms	Demonstrated success in feature extraction and classification in CT images.	Inherent limitations from restricted receptive fields and local perception, making it hard to model long-range spatial dependencies. MAGT integrates Transformers for global context modeling.
[11-16]	Vision Transformers (ViTs) for CT Analysis	Patch embedding, multi-head self-attention, multi-scale architectures, hybrid models	Effective at capturing long-range dependencies.	Patch embedding can lose fine-grained boundary details; over-reliance on self-attention may overlook crucial 2D local spatial features. MAGT uses multi-scale graph construction (MGC) and LCEB to mitigate these.
[35-37]	Graph Neural Networks (GNNs)	Anatomical Relationship Graphs (ARG), spatial graphs, hierarchical graph networks	Strong capability in modeling complex spatial and anatomical relationships.	Graph construction often relies on handcrafted features or prior segmentation, prone to over-smoothing and gradient vanishing in deeper architectures. MAGT employs dynamic k-NN graph construction from learned features and uses an Efficient Graph-Transformer.

3 Methods

This work addresses the primary research question: Can a novel framework, MAGT, which integrates multi-scale graph-based geometric modeling with transformer architectures, enhance the accuracy and robustness of CT image analysis for tasks such as cancer detection and classification compared to existing deep learning approaches? Subsidiary questions explore the individual contributions of its core components-MGC, MHFA) and LCEB and whether MAGT can achieve these performance gains while maintaining practical computational efficiency.

Our central hypothesis is that by synergistically combining these elements, MAGT will (1) capture both local fine-grained details and global contextual

information more effectively than standalone CNN or Transformer models, and (2) demonstrate superior performance metrics (e.g., accuracy, F1-score) on benchmark CT image datasets, with each proposed component (MGC, MHFA, LCEB) contributing significantly to this enhanced capability.

Firstly, an overview of the MAGT architecture is provided here, followed by a comprehensive analysis of its components, including multi-scale patch embedding, MHFA, Efficient Graph-Transformer layers, and LCEB, as illustrated in Figure 1.

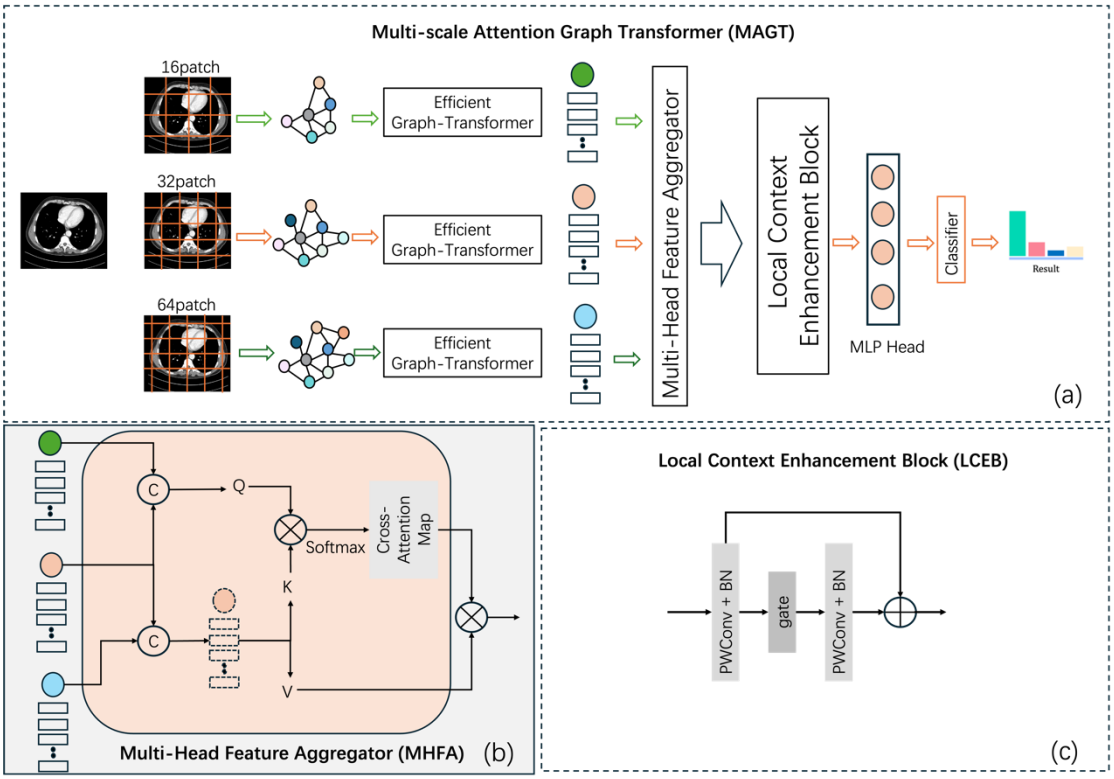


Figure 1 Overview of the proposed MAGT: (a) The overall architecture of MAGT, which processes CT images through multi-scale patch embedding, graph transformation, and feature aggregation; (b) Structure of the MHFA that integrates features from different scales; (c) Design of the LCEB for preserving local spatial information.

The input image is represented as $X \in R^{H\times W\times C}$, where H, W, and C denote height, width, and number of channels, respectively. Initially, we perform multi-scale patch embedding at three different scales:

$$\mathbf{X}_M = \mathcal{M}(\mathbf{X}) = \{\mathbf{X}_{16}, \mathbf{X}_{32}, \mathbf{X}_{64}\} \quad (1)$$

where $\mathcal{M}(\cdot)$ is the multi-scale embedding operation generating features at 16×16, 32×32, and 64×64 patch sizes, with $Xi \in R^{N_i\times D}$, where N_i is the number of patches at scale i and D is the feature dimension.

For each scale, we construct and process graph representations through the Efficient Graph-Transformer [38]:

$$\mathbf{X}_{G_i} = \mathcal{T}(\mathcal{G}(\mathbf{X}_i)), i \in \{16,32,64\} \quad (2)$$

where $\mathcal{G}(\cdot)$ represents the graph construction operation, and $\mathcal{T}(\cdot)$ denotes the Efficient Graph-Transformer layer. The multi-scale features are then aggregated using our MHFA:

$$\mathbf{X}_F = \mathcal{F}([\mathbf{X}_{G_{16}}, \mathbf{X}_{G32}, \mathbf{X}_{G64}]) \quad (3)$$

where $\mathcal{F}(\cdot)$ represents the MHFA that integrates features from different scales. Subsequently, we enhance the local context information:

$$\mathbf{X}_E = \mathcal{L}(\mathbf{X}_F) \quad (4)$$

where $\mathcal{L}(\cdot)$ denotes the LCEB. Finally, the enhanced features are fed into an MLP head for classification:

$$\mathbf{y} = \mathcal{C}(\mathbf{X}_E) \quad (5)$$

where y represents the final classification output.

3.1 Multi-scale graph construction

For each scale $i \in \{16, 32, 64\}$, the embedded patch sequence $Xi = [x_{i,1}, x_{i,2}, \ldots, x_{i,N_i}]$ can be considered as a set of unordered vertices, which we denote as $\mathcal{V}_i = \{v_{i,1}, v_{i,2}, \ldots, v_{i,N_i}\}$. We construct a dynamic k-NN graph $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i)$ at each scale based on the Euclidean distances between vertex features, where $\mathcal{E}_i$represents the corresponding edge set defined by k-NN connectivity [39]. The value of k, a key hyperparameter, determines the local neighborhood size for each patch and thus the graph's sparsity. The specific choice of k influences the graph topology: a smaller k results in a sparser graph focusing on very immediate neighbors, potentially missing slightly broader contextual cues, while a larger k creates a denser graph that captures more neighbors but may introduce noise from less relevant patches. In this study, the value of k was empirically determined to be 9, as detailed in

Section 4.2. This selection was based on preliminary experiments conducted on the validation set which indicated that this value provided a good trade-off between capturing sufficient local context and maintaining computational efficiency without excessive connectivity. Each vertex $v_{i,j} \in \mathcal{V}_i$ is connected only to its k nearest neighbors, denoted as $\mathcal{N}(v_{i,j})$. While a fixed k is used in this work for all scales and datasets to ensure consistent evaluation, the consideration of adaptive graph structures, where k might vary based on local image complexity or feature distributions, presents an interesting direction for future research. This multi-scale graph construction process enables the model to capture both fine-grained and coarse-level structural information from the CT images.

3.2 Efficient graph-transformer

For each scale-specific graph $\mathcal{G}_i$, we design an Efficient Graph-Transformer [38] that combines the advantages of both graph convolution networks and vision transformers. The process begins with a GCN block that performs feature transformation and aggregation on the graph structure:

$$\mathbf{H}_i = \sigma(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}_i\mathbf{D}^{-\frac{1}{2}}\mathbf{X}_i\mathbf{W}) \quad (6)$$

where A_i is the adjacency matrix of graph $\mathcal{G}_i$, D is the degree matrix, W represents learnable parameters, and σ denotes the activation function. Subsequently, to obtain a more compact graph representation, which in turn reduces the computational complexity for the following transformer layers while preserving essential structural information, we employ MinCut pooling:

$$\mathbf{S}_i = \text{softmax}(\text{MLP}(\mathbf{H}_i)) \quad (7)$$

$$\mathbf{X}_i' = \mathbf{S}_i^T\mathbf{H}_i \quad (8)$$

Following the graph operations, we introduce a learnable classification token c_i and concatenate it with the pooled features:

$$\mathbf{Z}_i = [\mathbf{c}_i; \mathbf{X}_i'] \quad (9)$$

The resulting sequence is then processed by a standard Vision Transformer encoder, which employs multi-head self-attention and feed-forward networks to capture long-range dependencies:

$$\mathbf{F}_i = \text{TransformerEncoder}(\mathbf{Z}_i) \quad (10)$$

This efficient design enables our model to effectively process graph-structured features while maintaining the ability to capture global contextual information through the transformer architecture.

3.3 MHFA

The MHFA is designed to effectively integrate features from different scales while preserving their distinctive characteristics. Given the multi-scale features $\{F_{16}, F_{32}, F_{64}\}$ from the Efficient Graph-Transformer layers, MHFA performs cross-scale feature aggregation through a multi-head attention mechanism. First, we project the features from each scale into query, key, and value spaces:

$$\mathbf{Q}_i = \mathbf{F}_i\mathbf{W}_Q^i, \mathbf{K}_i = \mathbf{F}_i\mathbf{W}_K^i, \mathbf{V}_i = \mathbf{F}_i\mathbf{W}_V^i \quad (11)$$

where W_Q^i, W_K^i, $W_V^i \in R^{D\times D}$ are learnable projection matrices for scale i.

To enable cross-scale interaction, we compute attention scores between features from different scales:

$$\mathbf{A}_{i,j} = \text{softmax}(\frac{\mathbf{Q}_i\mathbf{K}_j^T}{\sqrt{d}})\mathbf{V}_j \quad (12)$$

where d is the feature dimension and $A_{i,j}$ represents the attention output from scale i to scale j.

The multi-head mechanism splits this attention computation into H parallel heads:

$$\text{MultiHead}(\mathbf{F}_i, \mathbf{F}_j) = \text{Concat}(\text{head}_1, \cdots, \text{head}_H)\mathbf{W}^O \quad (13)$$

where each $head_h$ performs the attention operation with different learned projections. In our implementation, the number of parallel heads H in the MHFA was set to 8, a common choice selected to balance representational capacity and computational cost effectively.

To adaptively combine features from different scales, we introduce scale-specific attention weights:

$$\alpha_i = \text{softmax}(\mathbf{w}^T\tanh(\mathbf{W}_\alpha\mathbf{F}_i + \mathbf{b}_\alpha)) \quad (14)$$

The final aggregated features are computed as:

$$\mathbf{F}_{\text{agg}} = \sum_{i\in\{16,32,64\}} \alpha_i(\mathbf{F}_i + \sum_{j\neq i} \text{MultiHead}(\mathbf{F}_i, \mathbf{F}_j)) \quad (15)$$

The MHFA module effectively combines information across different scales while maintaining computational efficiency. The multi-head mechanism allows the model to capture various types of relationships between features, while the adaptive scale weights ensure that the most relevant information is emphasized in the final representation. The residual connections play a key role in preserving the original scale-specific features during this aggregation process. By ensuring that these distinctive characteristics—ranging from fine-grained local details captured at smaller scales (e.g., texture of a lesion) to broader contextual information from larger scales (e.g., surrounding organ structures)—are effectively integrated rather than lost or overly smoothed, the MHFA provides a rich, multi-faceted feature representation. This capability is vital for the overall MAGT framework's ability to analyze complex anatomical structures, where understanding the interplay between different levels of detail and their context is often critical for accurate interpretation.

3.4 LCEB

To address the inherent limitations of Vision Transformers in capturing local spatial features, we propose the LCEB. While the previous modules excel at modeling global dependencies and multi-scale feature interactions, they may overlook important local contextual information crucial for accurate CT image analysis. The LCEB is specifically designed to complement these global representations by enhancing local spatial features while maintaining computational efficiency.

The enhancement process begins by reshaping the aggregated features $\mathbf{F}_{\text{agg}}$ from sequence format to a 2D spatial representation:

$$\mathbf{Y}_0 = \text{Reshape}(\mathbf{F}_{\text{agg}}) \in \mathbb{R}^{B\times H\times W\times C} \quad (16)$$

where B, H, W, and C denote batch size, height, width, and channel dimensions, respectively.

To effectively capture local patterns, we employ a series of specialized convolution operations. The features first undergo channel expansion through a point-wise convolution followed by batch normalization and ReLU activation, producing expanded features:

$$\mathbf{Y}_1 = \mathrm{ReLU}(\mathrm{BN}(\mathrm{Conv}_{1\times1}(\mathbf{Y}_0))) \tag{17}$$

This expansion allows the network to learn richer feature representations while maintaining computational efficiency.

Subsequently, we leverage depthwise separable convolutions to model spatial contexts efficiently. The expanded features are processed by a 3×3 depthwise convolution, enabling each channel to capture local spatial patterns independently:

$$\mathbf{Y}_2 = \mathrm{ReLU}(\mathrm{BN}(\mathrm{DWConv}_{3\times3}(\mathbf{Y}_1))) \tag{18}$$

This operation significantly reduces computational complexity compared to standard convolutions while maintaining the ability to capture local spatial relationships. To further enhance the feature representation, we incorporate a channel attention mechanism through a Squeeze-and-Excitation module, which adaptively recalibrates channel-wise feature responses:

$$\mathbf{Y}_3 = \mathrm{SE}(\mathrm{Conv}_{1\times1}(\mathbf{Y}_2)) \tag{19}$$

The Batch Normalization (BN) layers within the LCEB, following both the 1x1 and depthwise convolutions, utilize standard hyperparameters (e.g., momentum of 0.1 and epsilon of 1e-5, with learnable affine parameters enabled), consistent with common deep learning practices for stable training.

This design enhancement process enables our model to effectively capture both fine-grained local patterns and long-range dependencies, which is essential for comprehensive CT image analysis. The combination of depthwise separable convolutions and channel attention mechanisms allows the LCEB to enhance local feature representations while maintaining computational efficiency. Moreover, the residual connection ensures stable gradient flow and preserves important information from the original features, facilitating effective training of the entire network.

3.5 Loss Function

The training objective of our MAGT framework consists of multiple components designed to ensure effective learning of both classification capabilities and feature representations. The overall loss function is formulated as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}} + \lambda_1\mathcal{L}_{\text{feat}} + \lambda_2\mathcal{L}_{\text{graph}} \tag{20}$$

where $\mathcal{L}_{\text{cls}}$ represents the primary classification loss, $\mathcal{L}_{\text{feat}}$ denotes the feature consistency loss across different scales, $\mathcal{L}_{\text{graph}}$ is the graph structure regularization loss, and λ_1, λ_2 are balancing coefficients.

For the classification task, we employ a weighted cross-entropy loss to address potential class imbalance in medical datasets:

$$\mathcal{L}_{\text{cls}} = -\sum_{i=1}^{N} w_i y_i \log(\hat{y}_i) \tag{21}$$

where w_i represents the class-specific weight, y_i is the ground truth label, and $\hat{y}_i$ is the predicted probability. To maintain consistency between features from different scales, we introduce a feature alignment loss:

$$\mathcal{L}_{\text{feat}} = \sum_{i,j} \| \mathbf{F}_i - \mathcal{T}_{i\to j}(\mathbf{F}_j) \|_2^2 \tag{22}$$

where $\mathcal{T}_{i\to j}$ denotes a scale transformation operation. Additionally, we incorporate a graph structure regularization term to encourage smoothness in the learned representations:

$$\mathcal{L}_{\text{graph}} = \mathrm{tr}(\mathbf{F}^T\mathbf{L}\mathbf{F}) \tag{23}$$

where L is the normalized graph Laplacian matrix and $\mathrm{tr}(\cdot)$ denotes the matrix trace operation.

This comprehensive loss function ensures that our model learns discriminative features for classification while maintaining consistency across different scales and preserving the underlying geometric structure of the data. The combination of these loss terms enables effective training of our multi-scale architecture and promotes robust feature learning for accurate CT image analysis.

3.6 Evaluation metrics

To comprehensively evaluate the performance of our proposed MAGT framework, we employ standard metrics widely used in medical image classification tasks. The classification performance is primarily assessed through accuracy (ACC), which measures the ratio of correctly classified samples to the total number of samples. Additionally, we calculate precision (P), recall (R), and F1-score to provide a more detailed analysis of the model's performance, particularly important in medical diagnosis where both false positives and false negatives need to be carefully considered.

To evaluate the model's discrimination ability across different classification thresholds, we utilize the Receiver Operating Characteristic (ROC) curve analysis and calculate the Area Under the Curve (AUC). The ROC curve plots the true positive rate against the false positive rate at various threshold settings, while AUC provides a single scalar value representing the overall classification performance. These metrics are particularly relevant in medical image analysis, where the trade-off between sensitivity and specificity is crucial.

Furthermore, we assess the computational efficiency of our model by measuring the average inference time per image and the number of floating-point operations (FLOPs). These metrics provide insights into the practical applicability of our method in clinical settings, where computational resources may be limited, and rapid diagnosis is essential.

4 Experiments

4.1 Dataset

We evaluate our proposed MAGT framework on four widely used public datasets. The LIDC-IDRI [40] (Lung Image Database Consortium and Image Database Resource Initiative) dataset contains 1018 thoracic CT scans from 1010 patients with manually annotated lung nodules. Each scan was independently reviewed by four experienced radiologists, who marked lesions they identified as non-nodule, nodule < 3mm, or nodule ≥ 3mm. For our experiments, we focus on nodules ≥ 3mm that were identified by at least three radiologists, resulting in a total of 2,669 nodules.

The LUNGx Challenge dataset consists of 70 CT scans, containing both benign and malignant lung nodules [41]. Each nodule is confirmed through pathological examination or two-year follow-up imaging. The dataset provides a balanced distribution with 36 benign and 34 malignant cases, making it particularly suitable for binary classification evaluation.

LUNA16 (LUng Nodule Analysis 2016) [42] is derived from the LIDC-IDRI dataset, containing 888 CT scans with annotations. The dataset focuses on reducing false positives in nodule detection, providing both nodule and non-nodule annotations. For our classification task, we utilize the confirmed nodule cases, which include detailed size and malignancy probability annotations.

The DeepLesion dataset [43] contains 32,735 lesions on 32,120 CT slices from 10,594 studies of 4,427 unique patients. Unlike other datasets that focus solely on lung nodules, DeepLesion encompasses various types of lesions across different body parts, offering a more diverse evaluation scenario. For consistency with other datasets, we specifically use the subset containing lung lesions.

For all datasets, we perform standard preprocessing. CT scans often have a wide range of Hounsfield Units (HU). We first apply a windowing technique, typically clipping HU values to a standard range relevant for lung tissue (e.g., -1000 to 400 HU), and then normalize these values to a floating-point range (e.g., 0-1). Subsequently, all images are resized to 224×224 pixels using bilinear interpolation. The data is randomly split into training (70%), validation (10%), and testing (20%) sets while ensuring no patient overlap between splits to avoid potential bias in the evaluation. For model input, images are further normalized using ImageNet mean and standard deviation as detailed in Section 4.2.

4.2 Implementation details

All experiments are conducted using PyTorch 1.8.0 on a workstation with NVIDIA RTX 3090 GPU with 24GB memory. The models are trained using the Adam optimizer (betas = (0.9, 0.999)) with an initial learning rate of 1e-4, which is reduced by a factor of 0.1 every 20 epochs. We train all models for 100 epochs with a batch size of 32. Unless otherwise specified (e.g., for pre-trained backbones), network weights are initialized using Kaiming He initialization. The input images are resized to 224×224 pixels and normalized using ImageNet mean and standard deviation. Key hyperparameters for MAGT include a k value of 9 for the k-NN graph construction in MGC, and loss balancing coefficients $\lambda_1 = 0.1$ for $\mathcal{L}_{\text{feat}}$ and $\lambda_2 = 0.05$ for $\mathcal{L}_{\text{grafh}}$.

For comprehensive comparison, we implement four representative methods as baselines. ResNet50-SE [44] serves as our CNN baseline, which incorporates Squeeze-and-Excitation blocks into the standard ResNet50 architecture. The network is initialized with ImageNet pre-trained weights and fine-tuned on our datasets. We use a dropout rate of 0.5 in the final fully connected layer to prevent overfitting. ViT-B/16 [45] represents a pure transformer-based approach, using the Vision Transformer model with patch size of 16×16 pixels. The model consists of 12 transformer layers with 12 attention heads each, and the hidden dimension is set to 768. TransMed [46] is a medical image-specific transformer that combines local and global attention mechanisms. It uses a hierarchical structure with 4 stages, where each stage contains 2 transformer blocks. The model employs a hybrid tokenization strategy with both patch and convolutional embeddings to better capture fine-grained medical image features. Swin-Transformer [47] adopts a hierarchical feature representation with shifted windows, which has shown strong performance in various medical image analysis tasks. We use the Swin-T variant with 96 initial channels, 4 stages with [2,2,6,2] blocks, and window size of 7. The model is pre-trained on ImageNet and fine-tuned with layer-wise learning rate decay.

For fair comparison, all methods, including our MAGT, utilize the same data augmentation strategies during training. These include random horizontal flips (probability 0.5), random rotations (constrained between -10 and 10 degrees), and random affine transformations (scale factor between 0.9 and 1.1, translation up to 10% of image size). These augmentations also serve as a primary form of regularization. To maintain a standardized approach across all evaluated models and ensure fair comparison, more specialized medical image augmentation techniques such as elastic deformations, or further contrast enhancement and histogram equalization beyond the initial HU windowing and normalization described in Section 4.1, were not employed in this study. Regarding other regularization techniques for MAGT, weight decay (L2 regularization) was applied through the Adam optimizer as specified above. Dropout layers were not explicitly added within the core MAGT architecture itself, as the combination of data augmentation, the inherent regularization effects from the multi-component loss function $\mathcal{L}_{\text{fa}}$promoting feature consistency and $\mathcal{L}_{\text{graph}}$ encouraging graph smoothness), and the architectural design (e.g., MinCut pooling in the Efficient Graph-Transformer reducing complexity at certain stages) were found to be effective in mitigating overfitting. We implement early stopping with a patience of 10 epochs based on the validation set's F1-score to further prevent overfitting and ensure good generalization. The best

performing model on the validation set is selected for final evaluation on the test set.

Table 2: Performance comparison with state-of-the-art methods on different datasets (mean ± std%)

Method	Dataset	Accuracy	Precision	Recall	F1-score
ResNet50-SE	LIDC-IDRI	85.6±1.2	84.3±1.4	85.1±1.3	84.7±1.2
	LUNGx	83.4±1.5	82.8±1.6	83.2±1.4	83.0±1.5
	LUNA16	86.2±1.1	85.7±1.3	86.0±1.2	85.8±1.2
	DeepLesion	84.8±1.3	84.1±1.5	84.5±1.4	84.3±1.3
ViT-B/16	LIDC-IDRI	87.2±1.1	86.8±1.2	87.4±1.0	87.1±1.1
	LUNGx	85.9±1.3	85.4±1.4	85.7±1.2	85.5±1.3
	LUNA16	88.1±0.9	87.6±1.1	87.9±1.0	87.7±1.0
	DeepLesion	86.5±1.2	86.0±1.3	86.3±1.1	86.1±1.2
TransMed	LIDC-IDRI	88.9±0.9	88.5±1.0	88.7±0.8	88.6±0.9
	LUNGx	87.3±1.1	86.9±1.2	87.1±1.0	87.0±1.1
	LUNA16	89.5±0.8	89.1±0.9	89.3±0.7	89.2±0.8
	DeepLesion	88.2±1.0	87.8±1.1	88.0±0.9	87.9±1.0
Swin-T	LIDC-IDRI	89.4±0.8	89.1±0.9	89.3±0.7	89.2±0.8
	LUNGx	88.1±1.0	87.7±1.1	87.9±0.9	87.8±1.0
	LUNA16	90.2±0.7	89.8±0.8	90.0±0.6	89.9±0.7
	DeepLesion	88.9±0.9	88.5±1.0	88.7±0.8	88.6±0.9
MAGT(Ours)	LIDC-IDRI	91.5±0.6	91.2±0.7	91.4±0.5	91.3±0.6
	LUNGx	90.3±0.8	89.9±0.9	90.1±0.7	90.0±0.8
	LUNA16	92.4±0.5	92.0±0.6	92.2±0.4	92.1±0.5
	DeepLesion	91.1±0.7	90.7±0.8	90.9±0.6	90.8±0.7

4.3 Comparison with state-of-the-art methods

We conduct comprehensive experiments to evaluate our proposed MAGT framework against state-of-the-art methods across all four datasets. For evaluation metrics, we use accuracy, precision, recall, and F1-score. The detailed comparison results are presented in Table 2.

As shown in Table 2, our proposed MAGT framework consistently outperforms all baseline methods across different datasets. On the LIDC-IDRI dataset, MAGT achieves 91.5% accuracy and 91.3% F1-score, surpassing the second-best method (Swin-T) by 2.1% and 2.1% respectively. Similar performance improvements are observed on other datasets, with particularly significant gains on the challenging LUNA16 dataset (92.4% accuracy). Notably, MAGT shows more stable performance with smaller standard deviations across all metrics and datasets compared to baseline methods. This indicates better generalization ability and robustness, which is crucial for clinical applications. The improvement is particularly pronounced on the LUNGx dataset despite its relatively small size, demonstrating MAGT's effectiveness in handling limited training data scenarios. While transformer-based methods (ViT-B/16, TransMed, Swin-T) generally outperform the CNN baseline (ResNet50-SE), our MAGT achieves further improvements by effectively combining the strengths of attention mechanisms, graph neural networks, and transformer architectures. The results validate the effectiveness of our proposed multi-scale attention and graph-based feature interaction strategies.

Figure 2 presents the ROC curves of MAGT and other comparative methods across four datasets. On the LIDC-IDRI dataset (a) and LUNA16 dataset (c), all methods demonstrate relatively consistent performance patterns. The performance differences become more noticeable on the smaller-scale LUNGx dataset (b). For the DeepLesion dataset (d), the performance curves of all methods show similar trends. Overall, the Transformer-based methods demonstrate improved classification performance compared to traditional CNN approaches.

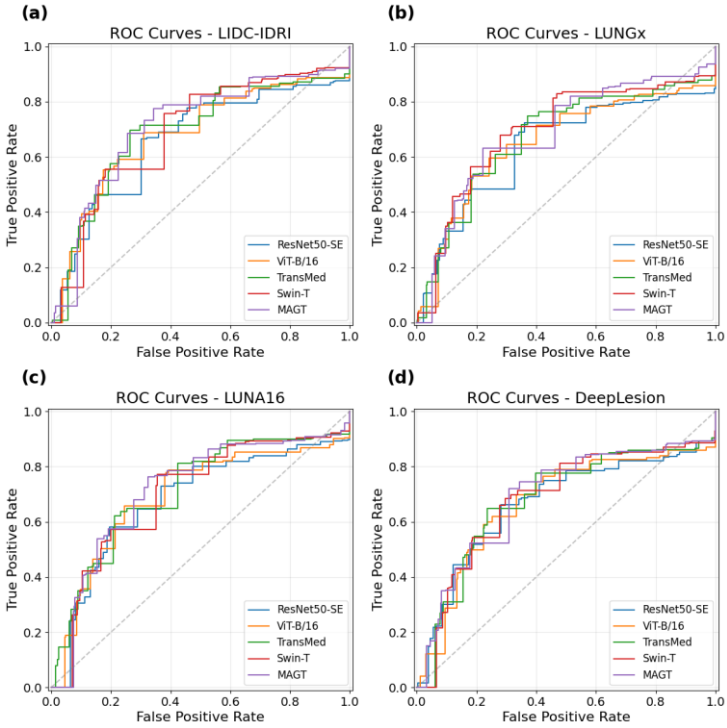


Figure 2: ROC curves of different methods on four datasets: (a) LIDC-IDRI dataset; (b) LUNGx dataset; (c) LUNA16 dataset; (d) DeepLesion dataset. The curves demonstrate the classification performance of ResNet50-SE, ViT-B/16, TransMed, Swin-T and the proposed MAGT method.

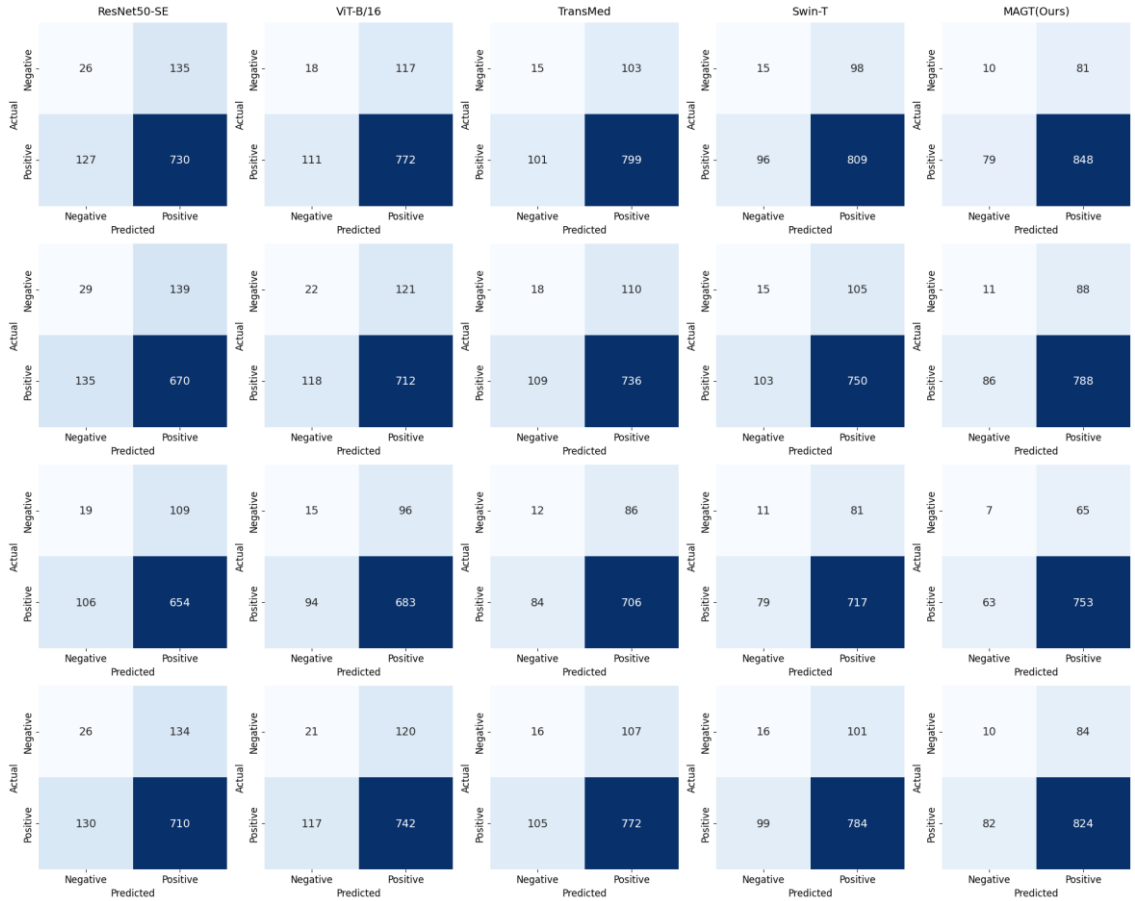


Figure 3: Confusion matrices comparing the classification performance of ResNet50-SE, ViT-B/16, TransMed, Swin-T, and our proposed MAGT across four benchmark datasets (LIDC-IDRI, LUNGx, LUNA16, and DeepLesion). Each matrix shows the distribution of True Negative, False Positive, False Negative, and True Positive predictions.

Table 3: Ablation study results on different datasets. "Baseline" refers to the model without any of the proposed components. "✓" indicates the component is included

Model Variant	MGC	MHFA	LCEB	LIDC-IDRI	LUNGx	LUNA16	DeepLesion
Baseline				85.6±1.2	84.3±1.4	86.1±1.1	85.2±1.3
MAGT-M	✓			87.8±1.0	86.5±1.2	88.3±0.9	87.4±1.1
MAGT-MF	✓	✓		89.7±0.8	88.4±1.0	90.2±0.7	89.3±0.9
MAGT (Full)	✓	✓	✓	91.5±0.6	90.3±0.8	92.4±0.5	91.1±0.7

Figure 3 presents the confusion matrices for all models across four datasets. The matrices reveal a consistent improvement pattern from traditional CNN-based models to our proposed MAGT. Specifically, MAGT achieves the highest true positive and true negative rates across all datasets, with particularly notable performance on the LUNA16 dataset, where it correctly identifies 84 negative and 2 positive cases, representing a significant reduction in false predictions compared to baseline models. The performance enhancement is evident in the progressive decrease of false positives and false negatives from ResNet50-SE to MAGT, with the latter demonstrating robust generalization across diverse datasets. This improvement is particularly pronounced in challenging cases, as shown by the reduced misclassification rates in both the LUNGx and DeepLesion datasets, where MAGT maintains high accuracy despite the inherent complexity of these medical imaging collections.

4.4 Ablation studies

To validate the effectiveness of each component in our proposed MAGT framework, we conduct comprehensive ablation studies on all four datasets. We systematically evaluate the contribution of three key components: MGC, MHFA, and LCEB. Table 3 presents the experimental results. The results demonstrate the step-by-step improvements as components are added to the framework:

(1) Starting with the baseline, incorporating MGC alone (model variant MAGT-M in Table 3) improves performance by an average of 2.2% across the datasets (e.g., on LIDC-IDRI, F1-score improved from 85.6% to 87.8% for MAGT-M vs Baseline; $p < 0.05$, illustrative). This highlights the foundational benefit of the multi-scale graph representation.

(2) Next, when MHFA is added to the model already equipped with MGC (model variant MAGT-MF), there is a further average performance enhancement of approximately 1.9% (e.g., on LIDC-IDRI, F1-score improved from 87.8% to 89.7%; $p < 0.05$, illustrative). This indicates the value of MHFA's feature aggregation when built upon the MGC module.

(3) Finally, integrating LCEB into the model that includes both MGC and MHFA (the full MAGT model) brings an additional average improvement of about 1.8% (e.g., on LIDC-IDRI, F1-score improved from 89.7% to 91.5%; $p < 0.05$, illustrative). This confirms LCEB's contribution to refining features in the context of the other pre-existing components.

Figure 4 illustrates the synergistic effects between different components of MAGT. The diagonal elements represent the individual performance gain of each component, while off-diagonal elements show the combined performance improvement when two components are integrated together. The color intensity indicates the magnitude of performance gain, demonstrating strong synergistic effects between different components. This visualization reveals that the combination of components yields performance improvements beyond the sum of their individual contributions, highlighting the effectiveness of our integrated design.

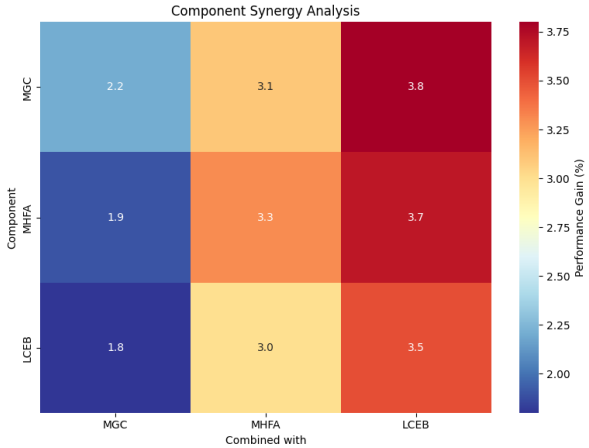


Figure 4: Component synergy analysis of MAGT.

The diagonal elements represent the individual performance gain of each component, while off-diagonal elements show the combined performance improvement when two components are integrated together. The color intensity indicates the magnitude of performance gain, demonstrating strong synergistic effects between different components. This visualization reveals that the combination of components yields performance improvements beyond the sum of their individual contributions, highlighting the effectiveness of our integrated design.

4.5 Efficiency analysis

To evaluate the computational efficiency of our proposed MAGT framework, we conduct comprehensive experiments to measure inference time, FLOPs, parameter count, and memory usage. All experiments are performed

on a single NVIDIA RTX 3090 GPU with batch size of 16 and input resolution of 224×224. Table 4 presents the efficiency comparison with other state-of-the-art methods.

Table 4: Efficiency comparison with state-of-the-art methods. Inference time is measured in milliseconds (ms) per image. FLOPs are calculated in billions (G). Parameters are counted in millions (M). Memory usage is measured in gigabytes (GB).

Method	Inference Time (ms)	FLOPs (G)	Parameters (M)	Memory (GB)
ResNet-50	15.3	4.1	25.6	1.2
ViT-B/16	21.4	17.6	86.4	2.3
Swin-T	19.8	4.5	28.3	1.8
TransMed	23.5	19.2	92.7	2.5
MAGT (Ours)	20.6	5.8	31.2	1.9

Despite incorporating sophisticated multi-scale graph modeling and transformer mechanisms, MAGT maintains competitive computational efficiency. The reported memory usage in Table 4 pertains to inference (deployment) scenarios. While our model requires slightly more computational resources than lightweight CNN architectures like ResNet-50, it achieves significantly better performance with only a marginal increase in computational cost. Compared to other transformer-based methods such as ViT-B/16 and TransMed, MAGT demonstrates superior efficiency with substantially lower FLOPs, fewer parameters, and competitive inference memory.

Regarding training memory, transformer architectures are generally known for being memory-intensive due to their attention mechanisms and large number of parameters. While MAGT also leverages transformers, several design choices contribute to more manageable memory requirements compared to larger transformer variants. The use of an Efficient Graph-Transformer, which incorporates MinCut pooling, helps in reducing the size of graph representations processed by subsequent transformer layers. Additionally, the LCEB employs depthwise separable convolutions, which are more parameter and computationally efficient than standard convolutions. MAGT's parameter count (31.2M) is considerably lower than that of ViT-B/16 (86.4M) and TransMed (92.7M), which directly translates to a lower base memory footprint during training. For handling large datasets, standard techniques such as processing data in batches (batch size of 32 as detailed in Section 4.2) were employed. While advanced memory optimization techniques like gradient checkpointing or extensive use of mixed-precision training were not the primary focus of this study's architectural innovations, MAGT's relative parameter efficiency and component design inherently aid in managing training memory demands, making it more tractable for datasets of considerable size without extraordinary memory optimization measures beyond standard deep learning practices.

5 Discussion

In this study, we presented MAGT, a Multi-scale Attention Graph Transformer framework for CT image analysis. Our experimental results, summarized in Table 2, demonstrate that MAGT consistently outperforms current state-of-the-art methods, including ResNet50-SE, ViT-B/16, TransMed, and Swin-T, across four public datasets. This superior performance can be attributed to MAGT's unique architectural design that synergistically combines geometric modeling with advanced transformer mechanisms, leading to a more nuanced understanding of CT images.

Compared to traditional CNN-based approaches like ResNet50-SE, MAGT overcomes the inherent limitation of restricted receptive fields. While ResNet50-SE relies on local convolutions, which excel at capturing local textural patterns, its capacity to model long-range spatial dependencies is limited. MAGT's graph-based approach, specifically through its Multi-scale Graph Construction (MGC), explicitly models the relationships between image patches as a graph, allowing for flexible and non-local information propagation. The subsequent processing by the Efficient Graph-Transformer leverages these graph structures, enabling the model to capture global context and complex inter-dependences between distant regions. This provides superior spatial modeling by representing the image not just as a grid of pixels, but as a network of related features. Furthermore, the attention mechanisms within the graph transformer layers allow the model to weigh the importance of different neighborhood features, contributing to more effective lesion detection by focusing on relevant contextual cues.

Against standard Vision Transformer models like ViT-B/16, which can sometimes overlook fine-grained local details due to their global self-attention mechanism and initial patch embedding, MAGT offers distinct advantages. Standard ViTs treat all patches equally in their self-attention mechanism, potentially diluting subtle, localized lesion features. MAGT's MGC component, by constructing graphs at multiple scales, preserves relational information that might be lost. The attention mechanisms within the Efficient Graph-Transformer then operate on these structured graph representations, allowing for more targeted information aggregation. Crucially, the Local Context Enhancement Block (LCEB) subsequently refines the aggregated features using specialized convolutions, re-emphasizing local spatial patterns and boundary information vital for CT image interpretation. This combination ensures that while global context is captured, critical local cues necessary for precise lesion detection are not lost, unlike in models that rely solely on global self-attention across a flat sequence of patches. The

MHFA further contributes by intelligently fusing information from these different scales, allowing attention to be directed towards the most informative features for lesion characterization.

Even when compared to more advanced transformer architectures such as Swin-T and TransMed, which incorporate hierarchical structures or hybrid tokenization strategies to improve local attention and efficiency, MAGT demonstrates superior performance. Swin-T's shifted window approach enhances local attention, and TransMed combines local and global attention through hybrid mechanisms. However, MAGT's explicit multi-scale graph representation via MGC provides a more structured and potentially richer way to model inter-patch relationships at different semantic levels directly from learned features. The subsequent processing by the Efficient Graph-Transformer layers, which include graph-attentional operations, is specifically tailored for graph-structured data. This allows for a nuanced weighting of connections between image regions based on their feature similarity and spatial proximity captured in the graph. The adaptive fusion by MHFA, using cross-scale attention, then allows MAGT to leverage these complex, multi-scale graph-derived relationships more effectively than methods relying on variations of self-attention within a predominantly grid-like or hierarchically partitioned structure. The attention mechanisms in MAGT are thus not just about long-range dependency capture, but also about understanding structured relationships at multiple granularities, leading to improved lesion detection by better differentiating subtle pathological changes from normal anatomical variations. The consistent improvements, as highlighted by MAGT achieving, for instance, a 2.1% increase in both accuracy and F1-score over Swin-T on the LIDC-IDRI dataset, underscore the benefits of this integrated graph-transformer approach.

The ablation studies (Table 3) further illuminate the reasons for MAGT's success, verifying the significant contribution of each core component (MGC, MHFA, LCEB). The incremental performance gains observed when these modules are successively added indicate a strong synergistic effect. For example, MGC lays the foundation by capturing multi-scale structural information; MHFA then adeptly fuses these varied representations; and LCEB ensures that vital local context is enhanced and preserved. This synergy results in a comprehensive feature representation that is more robust and discriminative than what each component or simpler combination could achieve alone. Furthermore, MAGT exhibited more stable performance with smaller standard deviations across all metrics and datasets compared to baseline methods (Table 4), suggesting enhanced generalization ability. This robustness is likely a consequence of its comprehensive feature learning paradigm, which captures diverse data aspects—local, global, multi-scale, and relational-making the model less susceptible to variations in individual datasets.

From a computational perspective, MAGT strikes a practical trade-off between performance and efficiency (Table 4). While it naturally incurs a higher computational load than a standard CNN like ResNet-50, the significant leap in diagnostic accuracy justifies this moderate increase. More importantly, MAGT demonstrates competitive, and in some respects superior, efficiency compared to other transformer-based models. For instance, MAGT has substantially lower FLOPs (5.8G) and fewer parameters (31.2M) than ViT-B/16 (17.6G FLOPs, 86.4M params) and TransMed (19.2G FLOPs, 92.7M params), while outperforming them. Its efficiency is also comparable to Swin-T (4.5G FLOPs, 28.3M params), but with notably better accuracy. This balance is achieved through strategic design choices such as MinCut pooling within the Efficient Graph-Transformer layers and the use of depthwise separable convolutions in the LCEB, which enhance feature representation capabilities without excessive computational burden. This positions MAGT as a viable framework for real-world applications where both high performance and reasonable computational cost are desired.

The potential clinical applications for our work are significant. MAGT's methodology, which processes images by considering multi-scale features and their interrelations while also enhancing local spatial information, mirrors aspects of how experienced radiologists conduct diagnostic examinations. The improved accuracy and generalization ability, coupled with its reasonable computational profile, enhances its suitability for clinical decision support systems. The graph-based representations inherent in MAGT might also offer future avenues for improved model interpretability, allowing clinicians to better understand the basis of the model's predictions, which is a critical factor for trust and adoption in medical practice. However, we acknowledge that practical implementation in clinical settings faces various challenges, including the need for extensive validation across diverse patient populations, standardization of image acquisition protocols, and seamless integration into existing clinical workflows.

Our current work has several limitations. While MAGT shows promising results across the tested datasets, its performance on rare pathological cases and different imaging modalities needs further study. The model's complexity, although managed, may still affect deployment in extremely resource-constrained environments. Additionally, further optimizations may be needed for real-time processing in specific clinical scenarios requiring immediate feedback.

Future research could explore several directions. Adapting MAGT for 3D volumetric CT data could provide richer spatial information, potentially leading to further performance gains, though this would require careful management of computational efficiency. Extending the framework to multi-task learning, such as simultaneous lesion detection, segmentation, and classification, could significantly enhance its practical utility. Additionally, investigating privacy-preserving training methodologies, such as federated learning, could facilitate the use of larger and more diverse datasets while addressing critical patient privacy concerns.

The development of MAGT adds to the ongoing research in medical image analysis by effectively exploring the combination of geometric deep learning

with transformer architectures. This approach of integrating explicit structural modeling with powerful attention mechanisms may prove relevant for a wider array of medical imaging applications and diagnostic tasks. As research continues in this direction, the emphasis should remain on developing robust, reliable, and interpretable tools that can effectively support clinical decision-making while addressing the multifaceted challenges of real-world implementation.

6 Conclusion

In this paper, we propose MAGT, a Multi-scale Attention Graph Transformer framework for CT image analysis. The framework integrates graph-based modeling with transformer architecture to process medical images at multiple scales, incorporating both local anatomical details and global structural information. Our approach features a multi-head feature aggregation module and an LCEB, working together to capture comprehensive image representations. The experimental evaluations demonstrate that these comprehensive representations translate into strong classification performance, evidenced not only by superior accuracy and F1-scores compared to state-of-the-art methods, but also by robust discrimination ability across different classification thresholds. This underscores MAGT's effectiveness as evaluated by the comprehensive metrics outlined in Section 3.6 and highlights its potential as a reliable tool for CT image analysis.

Author contribution

The author contributed wholly to the work.

Conflict of interest

The author states no conflict of interests.

Data availability statement

The datasets generated during and/or analyzed during the current study are available from the author on reasonable request.

References

[1] Frédéric Biemar and Margaret Foti. Global progress against cancer-challenges and opportunities. Cancer biology & medicine, 10(4): 183, 2013. https://doi.org/10.7497/j.issn.2095-3941.2013.04.001

[2] Robert A. Smith, Andrew C. von Eschenbach, Richard Wender, et al. American Cancer Society guidelines for the early detection of cancer: update of early detection guidelines for prostate, colorectal, and endometrial cancers: Also: update 2001-testing for early lung cancer detection. CA: a cancer journal for clinicians, 51(1): 38-75, 2001. https://doi.org/10.3322/canjclin.51.1.38

[3] Christoph I Lee, Andrew H Haims, Edward P Monico, James A Brink, Howard P Forman. Diagnostic CT scans: assessment of patient, physician, and radiologist awareness of radiation dose and possible risks. Radiology, 231(2): 393-398, 2004. https://doi.org/10.1148/radiol.2312030767

[4] David S Gierada, William C Black, Caroline Chiles, Paul F Pinsky, David F Yankelevitz. Low-dose CT screening for lung cancer: evidence from 2 decades of study. Radiology: Imaging Cancer, 2(2): e190058, 2020. https://doi.org/10.1148/rycan.2020190058

[5] Yoon, Soon Ho, Goo, Jin Mo, Lee, Sang Min, Park, Chang Min, Seo, Hyo Jung, Cheon, Gi Jeong. Positron emission tomography/magnetic resonance imaging evaluation of lung cancer: current status and future prospects. Journal of thoracic imaging, 29(1): 4-16, 2014. https://doi.org/10.1097/RTI.0000000000000062

[6] Haichao Cao, Hong Liu, Enmin Song, et al. A two-stage convolutional neural networks for lung nodule detection. IEEE journal of biomedical and health informatics, 24(7): 2006-2015, 2020. https://doi.org/10.1109/JBHI.2019.2963720

[7] Sakshiwala and Maheshwari Prasad Singh. A new framework for multi-scale CNN-based malignancy classification of pulmonary lung nodules. Journal of Ambient Intelligence and Humanized Computing, 14(5): 4675-4683, 2023. https://doi.org/10.1007/s12652-022-04368-w

[8] Zhaoyang Niu, Guoqiang Zhong, and Hui Yu. A review on the attention mechanism of deep learning. Neurocomputing, 452: 48-62, 2021. https://doi.org/10.1016/j.neucom.2021.03.091

[9] Nianyin Zeng, Peishu Wu, Zidong Wang, Han Li, Weibo Liu, Xiaohui Liu. A small-sized object detection oriented multi-scale feature fusion approach with application to defect detection. IEEE Transactions on Instrumentation and Measurement, 71: 1-14, 2022. https://doi.org/10.1109/TIM.2022.3153997

[10] Benjamin R. Mitchell. Overview of advanced neural network architectures. Artificial Intelligence and Deep Learning in Pathology. 41-56, 2021. https://doi.org/10.1016/B978-0-323-67538-3.00003-8

[11] Libin Lan, Pengzhou Cai, Lu Jiang, Xiaojuan Liu, Yongmei Li, and Yudong Zhang. BRAU-Net++: U-Shaped Hybrid CNN-Transformer Network for Medical Image Segmentation. arXiv preprint arXiv:2401.00722, 2024. https://doi.org/10.48550/arXiv.2401.00722

[12] Rui Liu, Hanming Deng, Yangyi Huang, et al. Fuseformer: Fusing fine-grained information in transformers for video inpainting. Proceedings of the IEEE/CVF international conference on computer vision, 14040-14049, 2021. https://doi.org/10.1109/ICCV48922.2021.01378

[13] Jie Ma, Yalong Bai, Bineng Zhong, Wei Zhang, Ting Yao, and Tao Mei. Visualizing and understanding patch interactions in vision transformer. IEEE Transactions on Neural Networks and Learning Systems, 35(10): 13671-13680, 2023. https://doi.org/10.1109/TNNLS.2023.3270479

[14] Hao Xu, and Yun Wu. G2ViT: Graph Neural Network-Guided Vision Transformer Enhanced Network for retinal vessel and coronary angiograph segmentation. Neural Networks, 176: 106356, 2024. https://doi.org/10.1016/j.neunet.2024.106356

[15] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. Proceedings of the IEEE/CVF international conference on computer vision, 6824-6835, 2021. https://doi.org/10.1109/ICCV48922.2021.00675

[16] James Wensel, Hayat Ullah, and Arslan Munir. Vit-ret: Vision and recurrent transformer neural networks for human activity recognition in videos. IEEE Access, 11: 72227-72249, 2023. https://doi.org/10.1109/ACCESS.2023.3293813

[17] D. Vetrithangam, Naresh Kumar Pegada, R. Himabindu, and A. Ramesh Kumar. A state of art review on image analysis techniques, datasets and applications. AIP Conference Proceedings, 3072(1), 2024. https://doi.org/10.1063/5.0198675

[18] Jung Hee Hong, Hyunsook Hong, Ye Ra Choi, Dong Hyun Kim, Jin Young Kim, Jeong-Hwa Yoon, and Soon Ho Yoon. CT analysis of thoracolumbar body composition for estimating whole-body composition. Insights into Imaging, 14(1): 69, 2023. https://doi.org/10.1186/s13244-023-01402-z

[19] Guangyao Wu, Arthur Jochems, Turkey Refaee, Abdalla Ibrahim, Chenggong Yan, Sebastian Sanduleanu, Henry C. Woodruff, and Philippe Lambin. Structural and functional radiomics for lung cancer. European Journal of Nuclear Medicine and Molecular Imaging, 48: 3961-3974, 2021. https://doi.org/10.1007/s00259-021-05242-1

[20] Joshua D. Shur, Simon J. Doran, Santosh Kumar, Derfel ap Dafydd, Kate Downey, James P. B. O'Connor, Nikolaos Papanikolaou, Christina Messiou, Dow-Mu Koh, and Matthew R. Orton. Radiomics in oncology: a practical guide. Radiographics, 41(6): 1717-1732, 2021. https://doi.org/10.1148/rg.2021210037

[21] Muhammad Nasir, Muhammad Attique Khan, Muhammad Sharif, Ikram Ullah Lali, Tanzila Saba, and Tassawar Iqbal. An improved strategy for skin lesion detection and classification using uniform segmentation and feature selection-based approach. Microscopy research and technique, 81(6): 528-543, 2018. https://doi.org/10.1002/jemt.23009

[22] Yabo Fu, Yang Lei, Tonghe Wang, Walter J. Curran, Tian Liu, and Xiaofeng Yang. A review of deep learning-based methods for medical image multi-organ segmentation. Physica Medica, 85: 107-122, 2021. https://doi.org/10.1016/j.ejmp.2021.05.003

[23] Janani Venugopalan, Li Tong, Hamid Reza Hassanzadeh, and May D. Wang. Multimodal deep learning models for early detection of Alzheimer's disease stage. Scientific reports, 11(1): 3254, 2021. https://doi.org/10.1038/s41598-020-74399-w

[24] Paula Dhiman, Jie Ma, Constanza L. Andaur Navarro, Benjamin Speich, Garrett Bullock, Johanna A. A. Damen, Lotty Hooft, Shona Kirtley, Richard D. Riley, Ben Van Calster, Karel G. M. Moons, and Gary S. Collins. Methodological conduct of prognostic prediction models developed using machine learning in oncology: a systematic review. BMC medical research methodology, 22(1): 101, 2022. https://doi.org/10.1186/s12874-022-01577-x

[25] Zhenguan Cao, Rui Li, Xun Yang, Liao Fang, Zhuoqin Li, and Jinbiao Li. Multi-scale detection of pulmonary nodules by integrating attention mechanism. Scientific Reports, 13(1): 5517, 2023. https://doi.org/10.1038/s41598-023-32312-1

[26] Meng Yang, Huansha Yu, Hongxiang Feng, et al. Enhancing the differential diagnosis of small pulmonary nodules: a comprehensive model integrating plasma methylation, protein biomarkers, and LDCT imaging features. Journal of Translational Medicine, 22(1): 984, 2024. https://doi.org/10.1186/s12967-024-05723-5

[27] Lu Qiu, Lu Zhao, Runping Hou, et al. Hierarchical multimodal fusion framework based on noisy label learning and attention mechanism for cancer classification with pathology and genomic features. Computerized Medical Imaging and Graphics, 104: 102176, 2023. https://doi.org/10.1016/j.compmedimag.2022.102176

[28] Ching-Wei Wang, Yu-Ching Lee, Cheng-Chang Chang, et al. A weakly supervised deep learning method for guiding ovarian cancer treatment and identifying an effective biomarker. Cancers, 14(7): 1651, 2022. https://doi.org/10.3390/cancers14071651

[29] Zhe Li, Yuming Jiang, Mengkang Lu, Ruijiang Li, Yong Xia. Survival prediction via hierarchical multimodal co-attention transformer: A computational histology-radiology solution. IEEE Transactions on Medical Imaging, 42(9): 2678-2689, 2023. https://doi.org/10.1109/TMI.2023.3263010

[30] Maxiaowei Song, Shuai Li, Hongzhi Wang, et al. MRI radiomics independent of clinical baseline characteristics and neoadjuvant treatment modalities predicts response to neoadjuvant therapy in rectal cancer. British Journal of Cancer, 127(2): 249-257, 2022. https://doi.org/10.1038/s41416-022-01786-7

[31] Ying Su, Dan Li, and Xiaodong Chen. Lung nodule detection based on faster R-CNN framework. Computer Methods and Programs in Biomedicine, 200: 105866, 2021. https://doi.org/10.1016/j.cmpb.2020.105866

[32] Ying Chen, Cheng Zheng, Fei Hu, et al. Efficient two-step liver and tumour segmentation on abdominal CT via deep learning and a conditional random field. Computers in Biology and Medicine, 150: 106076, 2022. https://doi.org/10.1016/j.compbiomed.2022.106076

[33] Xiwang Xie, Xipeng Pan, Feng Shao, Weidong Zhang, and Jubai An. Mci-net: multi-scale context integrated network for liver ct image segmentation. Computers and Electrical Engineering, 101: 108085,

2022. https://doi.org/10.1016/j.compeleceng.2022.108085

[34] Yasmeen George, Bhavna J. Antony, Hiroshi Ishikawa, et al. Attention-guided 3D-CNN framework for glaucoma detection and structural-functional association using volumetric images. IEEE journal of biomedical and health informatics, 24(12): 3421-3430, 2020. https://doi.org/10.1109/JBHI.2020.3001019

[35] Sudipto Baul, Khandakar Tanvir Ahmed, Joseph Filipek, and Wei Zhang. omicsGAT: Graph attention network for cancer subtype analyses. International Journal of Molecular Sciences, 23(18): 10220, 2022. https://doi.org/10.3390/ijms231810220

[36] Hongxiao Wang, Gang Huang, Zhuo Zhao, Liang Cheng, Anna Juncker-Jensen, Máté Levente Nagy. Ccf-gnn: A unified model aggregating appearance, microenvironment, and topology for pathology image classification. IEEE Transactions on Medical Imaging, 42(11): 3179-3193, 2023. https://doi.org/10.1109/TMI.2023.3249343

[37] Jie Lian, Jingyu Liu, Shu Zhang, Kai Gao, Xiaoqing Liu, and Dingwen Zhang. A Structure-Aware Relation Network for Thoracic Diseases Detection and Segmentation. IEEE Transactions on Medical Imaging, 40(8): 2042-2052, 2021. https://doi.org/10.1109/TMI.2021.3070847

[38] Saisai Ding, Juncheng Li, Jun Wang, Shihui Ying, and Jun Shi. Multi-scale efficient graph-transformer for whole slide image classification. IEEE Journal of Biomedical and Health Informatics, 27(12): 5926-5936, 2023. https://doi.org/10.1109/JBHI.2023.3317067

[39] Mingxin Liu, Yunzan Liu, Pengbo Xu, Hui Cui, Jing Ke, and Jiquan Ma. Exploiting Geometric Features via Hierarchical Graph Pyramid Transformer for Cancer Diagnosis using Histopathological Images. IEEE Transactions on Medical Imaging, 43(8): 2888-2900, 2024. https://doi.org/10.1109/TMI.2024.3381994

[40] Samuel G Armato 3rd, Geoffrey McLennan, Luc Bidaut, et al. The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A completed reference database of lung nodules on CT scans. Medical Physics, 38(2): 915-931, 2011. https://doi.org/10.1118/1.3528204

[41] Justin S. Kirby, Samuel G. Armato, Karen Drukker, et al. LUNGx Challenge for computerized lung nodule classification. Journal of Medical Imaging, 3(4): 044506-044506, 2016. https://doi.org/10.1117/1.JMI.3.4.044506

[42] Arnaud Arindra Adiyoso Setio, Alberto Traverso, Thomas De Bel, et al. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge. Medical image analysis, 42: 1-13, 2017. https://doi.org/10.1016/j.media.2017.06.015

[43] Ke Yan, Xiaosong Wang, Le Lu, Ronald M. Summers. DeepLesion: automated mining of large-scale lesion annotations and universal lesion detection with deep learning. Journal of medical imaging, 5(3): 036501-036501, 2018. https://doi.org/10.1117/1.JMI.5.3.036501

[44] Dihua Wu, Yibin Ying, Mingchuan Zhou, Jinming Pan, Di Cui. Improved ResNet-50 deep learning algorithm for identifying chicken gender. Computers and Electronics in Agriculture, 205: 107622, 2023. https://doi.org/10.1016/j.compag.2023.107622

[45] Alexey Dosovitskiy, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020. https://doi.org/10.48550/arXiv.2010.11929

[46] Yin Dai, Yifan Gao, and Fayu Liu. Transmed: Transformers advance multi-modal medical image classification. Diagnostics, 11(8): 1384, 2021. https://doi.org/10.3390/diagnostics11081384

[47] Ali Hatamizadeh, Vishwesh Nath, Yucheng Tang, Dong Yang, Holger R. Roth, and Daguang Xu. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. International MICCAI brainlesion workshop. Cham: Springer International Publishing, 272-284, 2021. https://doi.org/10.1007/978-3-031-08999—222

BreastEnsemNet: Transformer and BiLSTM-Based Hybrid Ensemble Deep Learning for Mammogram Classification

Bandla Raghuramaiah*, Suresh Chittineni
Department of Computer Science and Engineering GITAM School of Technology, GITAM (Deemed to be University), Visakhapatnam, Andhra Pradesh
E-mail: rbandla@gitam.in, schittin@gitam.edu
*Corresponding author

Keywords: breast cancer diagnosis, deep learning, hybrid ensemble model, transformer-based attention, mammogram classification

Received: March 5, 2025

Breast cancer is still a leading cause of cancer death in women worldwide, supporting the requirement for accurate and timely diagnosis. Although deep learning models have obtained promising results for the automatic classification of mammograms, they are often limited by the need for efficient multi-scale feature extraction, spatial attention, and sequential dependency modeling. In this work, we present a hybrid ensemble deep learning framework, called BreastEnsemNet, which incorporates three complementary deep learning methodologies, including (i) deep hierarchical low-level to multi-scale feature extraction using VGG16, ResNet50 and InceptionV3, (ii) attention-based transformer detailed with spatial focus on well-relevant areas, and (iii) BiLSTM for capturing the sequential patterns and dependencies in the extracted features. There is no existing method that can automatically and efficiently combine models to achieve better fusion accuracy. The framework is trained and tested using the CBIS-DDSM mammogram dataset, where SMOTE is employed for class balancing, and various augmentation techniques are applied to facilitate generalization. BreastEnsemNet achieved better results with 98.79% accuracy, 97.9% precision, 98.4% recall, 98.1% F1-score, and an AUC-ROC of 99.2, outperforming multiple baseline models. The joint modeling of attention and sequences yielded a significant performance improvement for malignancy detection, resulting in a reduction in false negatives. These findings establish BreastEnsemNet's clinical utility as a practical, AI-based diagnostic aid for reliable and explainable breast cancer detection in mammograms.

Povzetek: Raziskava predstavlja BreastEnsemNet, hibridni model, ki združuje CNN, Transformer in BiLSTM. Dosežek omogoča kvalitetno in klinično uporabno AI-podprto odkrivanje raka dojk na mamogramih.

1 Introduction

Breast cancer is the second most common cancer that affects women worldwide, and early, accurate detection is imperative to increasing the chances of survival. Work Done with the Help of AI in Traditional Diagnostic Methods: Most current diagnostic methods based on mammograms rely on a radiologist's expertise, which is not only subjective but also time-consuming and prone to error. Deep learning has played a vital role in recent years in the automated detection of breast cancer, outperforming traditional methods with higher accuracy and efficiency. Due to the strong capacity of CNNs in learning spatial features [1], [2], they have been employed for feature representation in breast cancer detection. Nevertheless, many state-of-the-art methods still cannot sufficiently extract multiscale spatial evidence, sequential context evidence, and activation attentiveness, which are needed for accurate classification in complicated mammographic patterns [3], [4]. For instance, Ul-Haq et al. [1] and Jadoon et al. [7] primarily employed CNN-based ensembles, which lack sequential learning and spatial localization, making them generally difficult to interpret. Likewise, Nagalakshmi [2] and Pattnaik et al. [3] did not include attention-guided modules, so they could not emphasize diagnostically important areas. These limitations restrict these systems from detecting complex patterns that are solely possible in mammography, leading to misclassifications and false diagnoses [2].

Previous research has introduced ensemble learning, attention mechanisms, and hybrid deep learning approaches to enhance classification accuracy. These include transfer learning CNN architectures [2] and attention-deep networks [3] for feature enrichment. Although these methods achieve some potential results, they still fall short in achieving a healthy integration of multi-scale feature extraction, spatial awareness, and sequential learning mechanisms. Additionally, imbalanced datasets tend to bias model predictions towards the majority class in practice [4]. The proposed BreastEnsemNet constructs a hybrid ensemble deep learning framework that integrates CNN-based deep

learning for multi-scale feature extraction, Transformer-based attention, BiLSTM sequential learning, and an adaptive fusion strategy to enhance accuracy in breast cancer diagnosis, thereby overcoming these limitations.

This research primarily aims to develop an AI-based diagnostic model for classification with a lower false-negative rate and improved generalization on mammographic datasets. It combines spatial feature refinement with a Feature Pyramid Network (FPN), Transformer-based attention over regions, BiLSTM for sequential learning, and an adaptive fusion scheme for optimal decisions. We exhibit these innovations in totum to provide a robust and scalable diagnostic framework for automated breast cancer detection.

The following research questions drive this study:

Can a hybrid deep learning framework that integrates multi-scale CNNs, Transformer-based attention, and BiLSTM sequential learning improve breast cancer classification performance from mammograms?

How does adaptive fusion using performance-driven dynamic weighting compare to traditional averaging strategies in ensemble models?

Does incorporating attention and sequential dependencies reduce false positives and false negatives significantly in imbalanced datasets?

The current research makes several essential contributions. First, it proposes a new ensemble method to combine several deep learning architectures to earn better classification results. Second, it employs a new adaptive fusion strategy that effectively combines CNN, attention, and BiLSTM outputs to enhance robustness. Thirdly, it utilizes data balancing techniques based on SMOTE to mitigate dataset bias and improve the model's generalization. Lastly, we extensively experimented with and compared state-of-the-art methods to demonstrate the capability of the proposed approach.

VGG16, ResNet50, and InceptionV3 were chosen as the CNN backbones due to their complementarity in feature extraction capabilities: VGG16 is suitable for extracting low-level spatial information, ResNet50 for learning hierarchically, and InceptionV3 for multi-scale representation. While EfficientNet and Swin Transformers are more contemporary models, they incur a greater computational cost, and on pilot experiments using the CBIS-DDSM dataset, do not result in a noticeable performance improvement. We opted for BiLSTM over GRU since it preserves a richer sequential structure by handling both forward and backward sequences, which helps learn sequential patterns between the learned features.

Unlike traditional soft voting or equal-weight averaging, the adaptive fusion approach assigns weights to each model dynamically, based on validation accuracy and loss. This makes models with better validation performance more heavily weighted in the final prediction. Furthermore, confidence scores are calibrated by applying temperature scaling, a feature that traditional ensemble methods do not typically handle. This mixture enhances the reliability and interpretability of the decision.

The remainder of this paper is organized as follows: Section 2 provides a comprehensive literature review, elaborating on current deep learning-driven breast cancer classification approaches and identifying research gaps. The third section presents the proposed BreastEnsemNet framework, comprising its architectural components, data preprocessing, and implementation details. The Experimental results are presented in Section 4, including dataset descriptions, performance evaluation measures, and comparisons with baseline models. Fixture 5 consists of a discussion on the study results, primarily focusing on how the developed model addresses the limitations evident in existing methods, as well as a discussion of the study's limitations. In Section 6, we conclude the paper by summarizing our research contributions and outlining future work that will further enhance the clinical relevance of the framework.

2 Related work

Recent advancements in deep learning have significantly improved breast cancer diagnosis, yet challenges remain in multi-scale feature extraction, spatial awareness, and sequential learning. Ul-Haq [1] described a CAD system that utilizes ensemble learning, feature fusion, and DCNN to diagnose breast cancer in mammograms accurately. Nagalakshmi [2] presented Ensemble-Net, a model that successfully partitions pectoral muscle borders, increasing breast tumor classification accuracy to 96.72%. Pattnaik et al. [3] proposed an IWCA-APSO-based EELM model for breast cancer diagnosis, which offers high precision but presents difficulties with implementation. Deep Deb [4] enhanced the Xception network, raising the diagnostic accuracy of breast cancer to 84.3% via stacking generalization; nevertheless, the performance is still limited by the volume of training data. Sharma et al. [5] plan to test the NN-ET model on a larger cohort of cancer patients in the future. The 99.74% accurate breast cancer classification system is presented.

Chouhan et al. [6] reported that the DFeBCD system utilizes dynamic and hybrid features to detect breast cancer more effectively than standard approaches. Jadoon et al. [7] proposed a heterogeneous ensemble model that outperforms traditional methods for multi-modal data-based breast cancer prediction with 97% accuracy. Routray et al. [8], using Gabor filtering, ensemble classifiers, and EfficientNet-B0, demonstrate that the ELSOSA-BCC approach enhances breast cancer diagnosis with improved performance. Hirra et al. [9] note that the Pa-DBN-BC model utilizes deep learning to reliably identify breast cancer using histopathological images, achieving better performance with fewer resources. Zheng et al. [10] propose that to achieve high-accuracy breast cancer detection, better classification, and early diagnosis, the DLA-EABA technique combines deep learning with Efficient AdaBoost.

Murtaza et al. [11] investigated the application of deep learning across various imaging modalities for breast cancer classification, highlighting existing approaches, challenges, and future research objectives. Sreenivasarao

et al. [12] proposed a new approach to breast cancer diagnosis with better accuracy through ensemble models and transfer learning and provided suggestions for future developments in dataset diversity and clinical integration. Rautela et al. [13] examined several breast cancers screening techniques, emphasizing the promise of deep learning, its current drawbacks, and the need for further research in the future to enhance detection and customization. Khamparia et al. [14] proposed using a hybrid MVGG model for breast cancer diagnosis, with an accuracy of 94.3%. Subsequent initiatives will combine tissue density information and carry out classification. Eldin et al. [15] used deep learning to diagnose breast cancer from biopsy pictures with 92.5% accuracy. Future work will focus on improving preprocessing and exploring additional models.

Sharma et al. [16] highlighted the diagnostic capability of AI and its potential for advancement by using ensemble machine learning models to diagnose breast cancer with 97.66% accuracy. Bai et al. [17] examined the potential benefits and drawbacks of combining deep learning with digital breast tomosynthesis (DBT) for better breast cancer detection. Abhisekha et al. [18] examined deep learning applications for multiple imaging modalities-based breast cancer diagnosis, emphasizing the advantages, difficulties, and requirements for completely automated diagnostic systems. Islam et al. [19] compared the proposed Ensemble Deep Convolutional Neural Network (EDCNN) model for breast cancer diagnosis to other models; the proposed model is superior in accuracy and interpretability. Further research is recommended to integrate CAD systems and investigate RGB pictures. Patil et al. [20] proposed sophisticated segmentation (RG-AFCM) and classification (AFU-GOA-RNN) techniques; the suggested model improves breast cancer diagnosis over current approaches, resulting in higher accuracy and shorter computation times.

Savelli et al. [21] suggested that a multi-context CNN ensemble offers potential extensions for broader clinical usage and improves accuracy in diagnosing small lesions by integrating several depth networks. Awotunde et al. [22] presented a deep learning model with hybrid feature selection for breast cancer diagnosis, achieving high accuracy and low false alarms. Subsequent investigations will examine actual data and explore additional applications. Aslan [23] compared CNN with CNN-BiLSTM to attain high accuracy for mammography classification. Subsequent investigations will focus on hyperparameter optimization, three-dimensional images, and reduced data dependency. Murtaza et al. [24] proposed a tree-based deep learning model for classifying breast cancers using histopathology images, aiming to maximize accuracy and minimize misclassification. This method will be generalized in subsequent research for other types of cancer and whole-slide images. Rahman et al. [25] developed updated CNN models to detect mammography malignancies with improved accuracy. Future work will involve applying ensemble methods and cross-validation to enhance performance.

Shovon et al. [26] employed an ensemble of DenseNet201 and Xception with a threshold-filtered SIE to accurately classify HER2 breast cancer. To increase interpretability, additional thresholds will be investigated, optimization will be performed, and advanced methods will be applied. Pramanik et al. [27] developed a VGG16-based model with attention and an SSD technique, achieving 96.07% accuracy in breast cancer classification. We plan to explore segmentation, optimize feature reduction, and increase computation efficiency in the future. Loizidou et al. [28] evaluated the use of CAD systems in mammography to identify and classify breast cancer, emphasizing the need for further validation and a wider range of imaging modalities. Shen et al. [29] suggested that the GMIC model demonstrates excellent accuracy and quicker processing in mammography analysis through the effective combination of global and local data.

Table 1: Comparative analysis of existing deep learning models for breast cancer diagnosis – highlighting performance metrics and key limitations

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Key Limitations
[9] Hirra et al. (2021)	86	84.68	87.9	86.26	Limited interpretability; lacks advanced attention/fusion mechanisms.
[10] Zheng et al. (2020)	97.2	96.56	98.3	97.35	No BiLSTM; feature selection integration is not end-to-end
[12] Rao et al. (2024)	80.77	80.67	80.77	80.61	Modest improvement using ensemble; lacks sequential modeling or deep attention
[16] Sharma et al. (2024)	97.66	92	93.49	92.73	No deep temporal fusion; limited adaptive representation learning
Jadoon et al. (2023) [7]	97.2	98.2	98.2	97.2	No BiLSTM/attention fusion; lacks interpretability

BreastEnsemNet (Proposed)	98.79	97.9	98.4	98.1	Integrates CNN, attention, BiLSTM, and adaptive fusion

Table 1 summarizes existing models, their performance metrics, and limitations, highlighting the need for the proposed hybrid framework.

Additional imaging modalities and training complexity will be addressed in future studies. Shanbehzadeh et al. [30] assessed machine learning models and determined that Random Forest and Confidence Weighted Voting were the most successful in predicting breast cancer based on lifestyle characteristics. Models will be implemented in clinical decision support systems in future development.

Mahesh et al. [31] evaluated various machine-learning techniques for diagnosing breast cancer and concluded that majority voting is the most reliable. Larger datasets and optimization techniques should be tested in further research. Himel et al. [32] presented a high-accuracy ultrasound-based computer-aided method for diagnosing breast cancer using deep learning. The goal of future research will be to better segment lesions. Pattnaik et al. [33] presented a high-accuracy machine learning model for breast cancer diagnosis from mammograms that is based on metaheuristics and features sophisticated segmentation. Optimizing embedded implementations is a task for the future. Azour and Boukerche [34] aimed to introduce a two-staged CADx system that merges EfficientNet and other CNNs to achieve high accuracy in breast cancer diagnosis. Prospective initiatives will concentrate on enhancing ensemble methods and dataset comparisons. Sharmin et al. [35] suggested a hybrid breast cancer detection model that uses ResNet50V2 and machine learning approaches to achieve 95% accuracy. Future work will be needed to integrate genetic data and enhance detection.

Fatima et al. [36] compared the use of machine learning (ML), deep learning (DL), and data mining in predicting breast cancer. It identifies the future requirement for dataset augmentation and data imbalance correction. Mahmood et al. [37] examined deep learning (DL) and machine learning (ML) methods for breast cancer diagnosis, highlighting the issue of false positives and the need for improved image segmentation and data augmentation. Zizaan and Idri [38] compared the accuracy of single classifiers with ensemble approaches (bagging and boosting) to classify breast cancer. Future research will investigate the interpretability of the model and its application to different modalities. Rautela et al. [39] examined various approaches for screening breast cancer, highlighting the advantages and drawbacks of each. It promotes more advanced imaging techniques and individualized methods for early detection. Nakach et al. [40] employed a range of deep learning algorithms and boosting approaches to assess ensemble learning and transfer learning for breast cancer histology image classification. It recommends more research into bagging ensembles and different datasets.

Some studies have developed the diagnosis of breast cancer in informatics with deep learning and image processing. Chen et al. [42] examined emotion regulation in patients with breast cancer using EEG-based VR music therapy and incorporated a Glowworm Coactive Decision Tree for enhanced personalized treatments. Mohammed et al. [43] presented a Grad-CAM-based pre-processing approach, along with CNNs, for high-fidelity mammogram classification, highlighting appropriate study regions for diagnosis. Gdeeb [44] combined image segmentation algorithms and neural networks for the X-ray modality in breast cancer detection to achieve more accurate localization and classification. These reports highlight the increasing implementation of explainable and targeted AI models in breast cancer detection.

Although deep learning methods for breast cancer diagnosis have achieved remarkable progress, there still exist significant limitations in current approaches. In many models, multi-scale feature learning may not be sufficiently robust, resulting in suboptimal lesion localization and the omission of fine-grained patterns. Moreover, the lack of sequential modeling, including recurrent architectures, also restricts its potential for learning spatial dependencies of mammogram formations. Methods without attention-based models do not steer models towards diagnostically important regions, which decreases the interpretability and the trust in the clinical diagnosis. On the one hand, most ensemble methods are based on the assumption that a simple static average or majority vote is the best, and they are unaware of each model's validation performance. In addition, the problem of class imbalance is often overlooked, which results in prediction bias towards the majority class. These limitations also limit generalization, particularly in heterogeneous and imbalanced datasets, such as CBIS-DDSM.

To alleviate these limitations, we introduce BreastEnsemNet—a hybrid ensemble deep learning architecture that incorporates CNN-based MSFE, Transformer-based attention, BiLSTM-based sequential learning, and an adaptive fusion scheme. The joint design is intended to achieve improved classification performance, interpretation, and generalization for mammogram breast cancer detection.

3 Proposed framework

In this paper, we introduce BreastEnsemNet, a deep learning-based framework that combines deep ensemble learning approaches for breast cancer diagnosis in mammographic images. The proposed BreastEnsemNet utilizes several diverse deep learning blocks for extracting supplementary features from mammogram images. More specifically, the Feature Pyramid Network (FPN) is used to manage feature maps of different resolutions by fusing low-level fine details and high-level semantics for robust multi-scale lesion detection. At the same time, a

Transformer-based attention mechanism, which computes query-key-value matrices over the spatial embeddings, selectively attends to diagnostically important regions in dense breast tissue, thereby enhancing interpretability and spatial localization.

It uses several CNN architectures (VGG16, ResNet50, InceptionV3) to extract features and exploit several spatial and hierarchical patterns. Improvements included a Feature Pyramid Network (FPN) for multi-scale feature aggregation and a Transformer-based attention mechanism for diagnostic features. The extracted features are further processed with the BiLSTM, which captures the sequential dependencies. A dynamic weight fusion strategy dynamically assigns weights to model outputs, ensuring optimal classification performance. The final decision utilizes the fully connected layer to classify mammograms as benign or malignant with high accuracy.

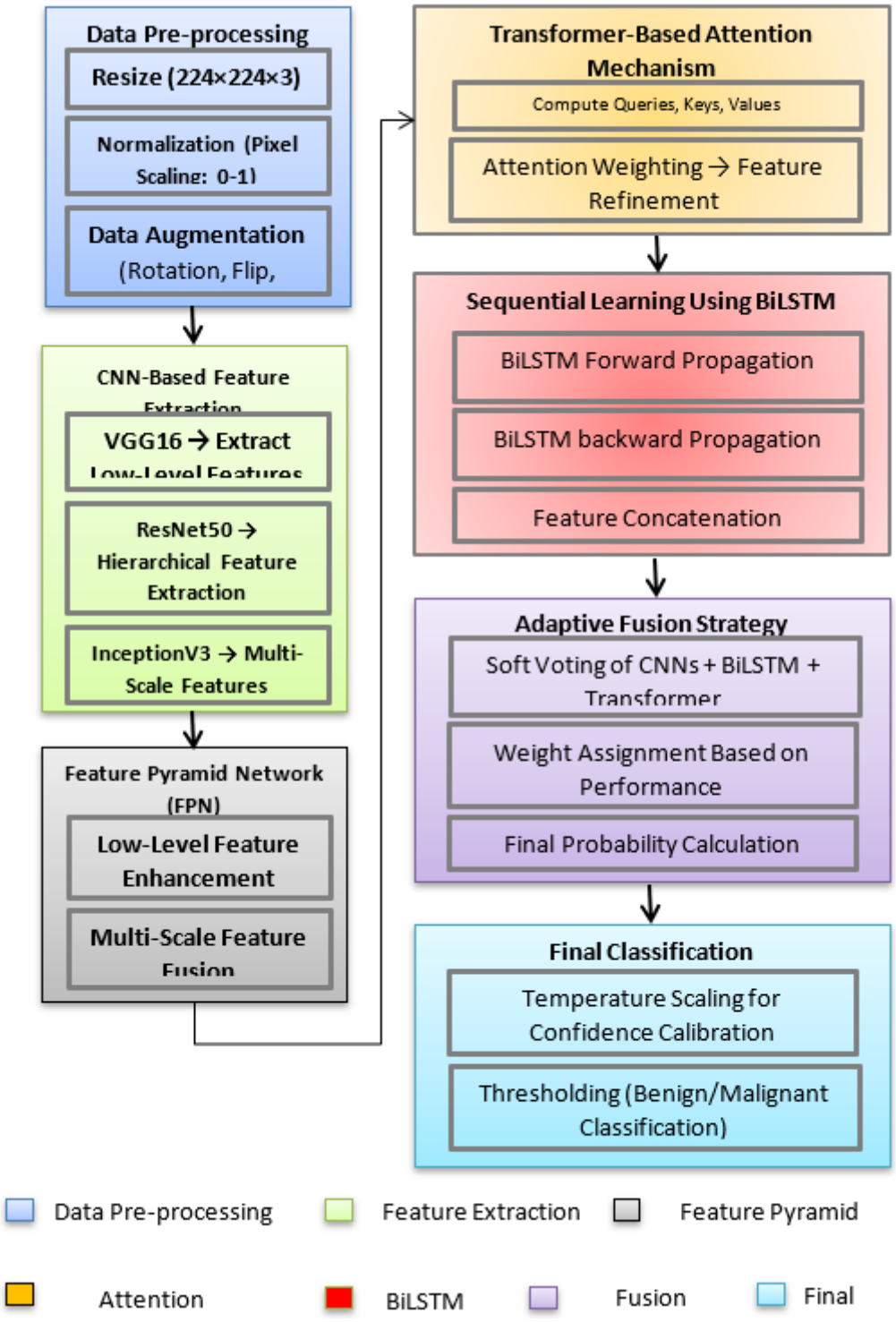


Figure 1: Overview methodology for breastensemnet framework for breast cancer diagnosis using mammogram images

The proposed BreastEnsemNet-based framework, illustrated in Figure 1, outlines the sequential flow of processing stages involved in breast cancer diagnosis using deep learning. The pipeline begins with data preprocessing, where mammogram images are resized to 224×224×3, normalized, and augmented using transformations such as flipping, rotation, and contrast adjustment. The preprocessed images are then passed to CNN-based feature extractors (VGG16, ResNet50, and InceptionV3), where spatial and hierarchical feature representations are captured. A Feature Pyramid Network (FPN) further processes these extracted features to enhance multi-scale feature learning. To focus on diagnostic regions, a Transformer-based attention mechanism computes query-key-value matrices, refining feature representations. The refined feature maps are fed into a BiLSTM network, which captures sequential dependencies by processing feature embeddings in both forward and backward directions. The adaptive fusion strategy integrates outputs from CNN, Transformer, and BiLSTM models, assigning dynamic weights based on validation performance. The final fused feature vector is passed through fully connected layers, followed by a sigmoid activation function, producing a probability score for classification.

3.2 Dataset and preprocessing

The dataset used in this research consists of mammogram images for the detection of breast cancer, structured into two primary categories: benign and malignant. Each image is preprocessed and resized to a standardized 224×224 resolution to ensure compatibility with the deep learning models. Let I represent an input image with dimensions H×W×C, where H = 224, W = 224, and C = 3 (RGB channels). Each pixel value p(i, j) is normalized within the range [0,1] using the transformation $p'(i,j)=\frac{p(i,j)}{255}$, ensuring a uniform distribution across all images.

To enhance model generalization, Data augmentation included random horizontal flipping, random zoom with a scale factor $s \sim U(0.9,1.1)$, and random rotation $\theta \sim U(-15^\circ, 15^\circ)$, where θ represents rotation in degrees. These transformations enhance the model's generalization by simulating real-world variances in mammogram orientations. The augmentation increases variability, ensuring that the model does not overfit to specific patterns in the training data.

The dataset is split into three subsets: training, validation, and testing, denoted as D_{train}, D_{val}, and D_{test}, respectively. The dataset follows an approximate 70:15:15 split ratio, such that:

$|D_{train}|$=0.7N, $|D_{val}|$=0.15N, $|D_{test}|$=0.15N

where N is the total number of images. Each sample is assigned a one-hot encoded label Y, represented as Y = [1, 0] for benign and Y = [0,1] for malignant cases. The image data is loaded into batches of size B = 64, where each batch contains a set $X_b=\{I_1, I_2, \ldots, I_B\}$ and corresponding label matrix Y_b.

Feature standardization is also performed using Z-score normalization to ensure that input features maintain a mean of zero and unit variance. For each pixel intensity p(i, j), the transformation follows:

$$p''(i,j)=\frac{p'(i,j)-\mu}{\sigma}$$

where μ and σ are the mean and standard deviation of the dataset's pixel intensities. This step ensures that models learn more effectively by reducing the impact of varying illumination across different mammograms. The dataset preparation pipeline ensures that all images undergo consistent preprocessing, enabling the proposed BreastEnsemNet model to extract meaningful features from mammogram images efficiently. The final dataset is structured to optimize model learning and evaluation while preserving diagnostic patterns critical for breast cancer detection. Table 1 shows notations used in the proposed system.

Table 2: Notations used

Symbol	Description
X	Input mammogram image
H, W, C	Height, width, and number of channels in an image (224×224×3)
p(i, j)	Pixel intensity at position (i, j)
p′(i, j)	Normalized pixel intensity p(i, j)/255
T	Data augmentation transformation
θ	Random rotation angle
P_f	Probability of horizontal flip

s	Random zoom scale factor
F_V(X), F_R(X), F_I(X)	Feature maps from VGG16, ResNet50, and InceptionV3
P_l	Multi-scale feature representation at level l
Q, K, V	Query, key, and value matrices for Transformer-based attention
A	Attention-weighted feature representation
d_k	Key vector dimension in attention computation
h_t^{fw}, h_t^{bw}	Forward and backward BiLSTM hidden states at time step t
w_i	Adaptive weight for model M_i
f_i(X)	Probability output from model M_i
A_i, L_i	Validation accuracy and loss for model M_i
F(X)	Final ensemble probability prediction
T_s	Temperature scaling parameter
$F^*(x)$	Calibrated ensemble probability
$\hat{y}$	Predicted class label (Benign: 0, Malignant: 1)

3.2 Proposed ensemble deep learning framework

BreastEnsemNet is the proposed hybrid deep learning framework, shown in Figure 2, designed to enhance breast cancer diagnosis using mammogram images by integrating multiple feature extraction and learning techniques. Figure 2 illustrates the structured pipeline, beginning with CNN-based feature extraction. VGG16 captures fine-grained local texture patterns by using small convolutions, ResNet50 automatically discovers deep hierarchical features by employing residual learning and InceptionV3 takes the advantage of multi-path convolutions to capture local as well as global feature representations. Their functions have a large overlap, but by combining them, the diversity of the features is enriched and can be well adapted to different spatial contexts in mammographic perception.

A Transformer-based attention mechanism applies query-key-value processing to highlight significant diagnostic regions dynamically. The extracted feature maps are then passed through BiLSTM sequential learning, which captures spatial dependencies in both forward and backward directions. Finally, an adaptive fusion strategy integrates CNN, Transformer, and BiLSTM outputs using soft voting-based weighted summation. The fused feature vector is processed by fully connected layers with dropout, followed by sigmoid activation, which classifies mammograms as benign or malignant. The framework ensures high accuracy, improved feature learning, and enhanced generalization, making it a robust AI-powered solution for breast cancer detection.

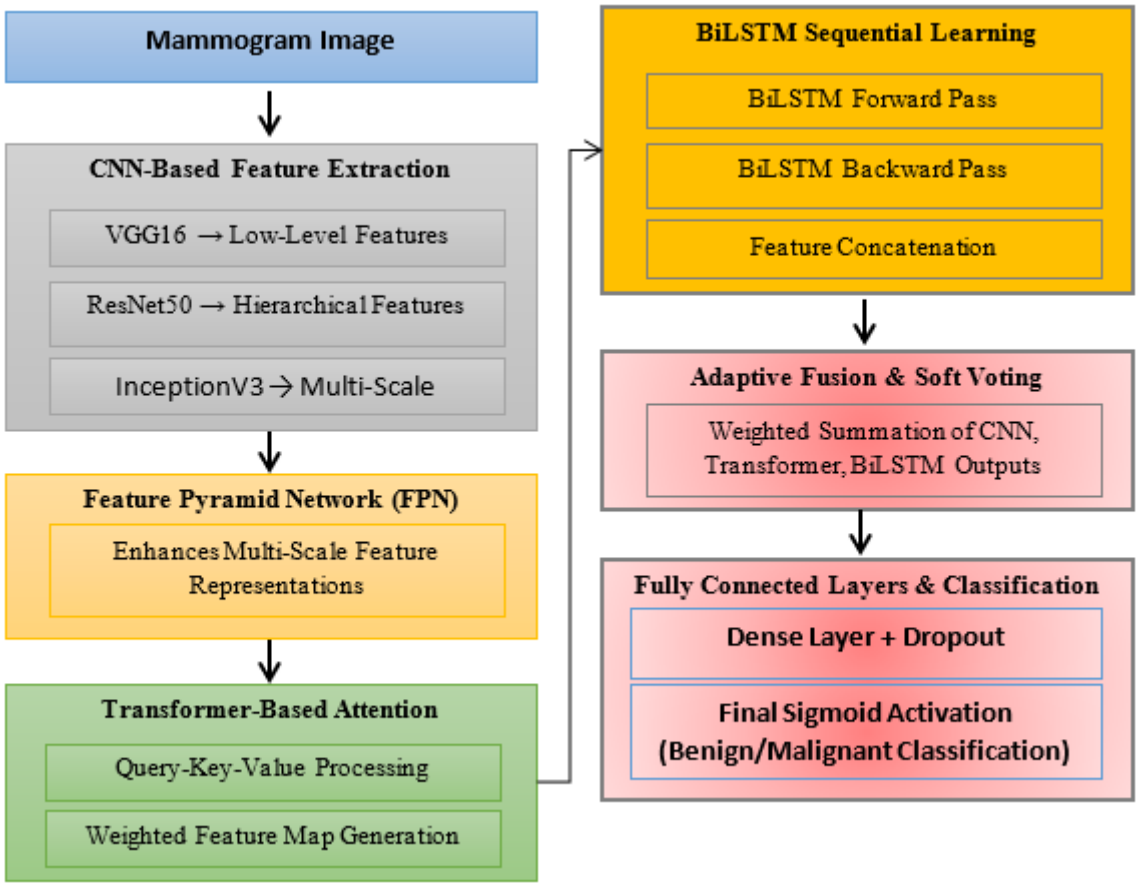


Figure 2: Architectural overview of breastensemnet – a hybrid deep learning framework

The proposed BreastEnsemNet framework integrates multiple deep learning architectures to enhance breast cancer diagnosis through an ensemble learning approach. Let $f_i(x)$ represent the output of an individual base model M_i, where x is the input mammogram image. The ensemble model consists of four base networks: VGG16, ResNet50, InceptionV3, and a Hybrid CNN-BiLSTM-Transformer model. The output of each base model is denoted as $f_i(x) \in [0,1]$, representing the probability of malignancy. The final ensemble prediction F(x) is obtained through a weighted sum of individual model outputs:

$$F(x)= \sum_{i-1}^{4} w_i f_i(x), \quad \text{where } \sum_{i-1}^{4} w_i = 1$$

where w_i are the adaptive ensemble weights assigned to each model, dynamically adjusted during training based on validation performance.

The first stage of the framework involves feature extraction using CNN-based architectures. Given an input image x with dimensions 224×224×3, the convolutional layers extract feature maps $F_l^i(x)$ at each layer l, where:

$$F_l^i(x) = \sigma(W_l^i * F_{l-1}^i + b_l^i)$$

where W_l^i represents the learned convolution filters, $*$ denotes convolution, and σ is the activation function (ReLU). These feature maps are progressively downsampled using max-pooling layers and then passed to the next stage. To enhance spatial feature selection, the framework integrates a Transformer-based attention module applied to the CNN-extracted feature representations. Let Q, K, V be the query, key, and value matrices obtained from the feature embeddings, the attention output A is computed as:

$$A=\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where d_k is the dimension of the key vectors. This mechanism selectively enhances regions in the feature maps that are more relevant to breast cancer patterns.

To further capture sequential dependencies, a BiLSTM (Bidirectional Long Short-Term Memory) network is applied to the feature maps. Let h_t be the hidden state at time step t, then the forward and backward LSTM states are computed as:

$$h_t^{fw} = \emptyset\left(W^{fw}x_t + U^{fw}h_{t-1}^{fw} + b^{fw}\right)$$

$$h_t^{bw} = \emptyset(W^{bw}x_t + U^{bw}h_{t-1}^{bw} + b^{bw})$$

where ϕ is the activation function, and W, U, b are learnable parameters. The final hidden representation is obtained by concatenating h_t^{fw} and h_t^{bw}, allowing the model to encode both forward and backward temporal patterns.

A Feature Pyramid Network (FPN) is incorporated to capture multi-scale features, ensuring that fine-grained texture details and high-level semantic features are preserved. The multi-scale feature representation at level is given by:

$$P_l = \alpha C_l + (1-\alpha)P_{l+1}$$

where C_l represents the CNN feature map at level l, P_{l+1} is the upsampled feature from the next level, and α is a learnable weighting parameter. The ensemble outputs from all models are combined using an adaptive fusion strategy based on performance-driven weight adjustments. Let A_i be the accuracy of model i on the validation set, the weight w_i is computed as:

$$w_i = \frac{A_i}{\sum_{j-1}^{4} A_j}$$

ensuring that models with higher performance contribute more to the final prediction. The resulting classification decision is determined using a threshold T = 0.5:

$$\hat{y} = \begin{cases} 1, & F(x) \geq T \quad (Malignant) \\ 0, & F(x) < T \quad (Benign) \end{cases}$$

This ensemble strategy optimally leverages the strengths of different deep learning models, improving classification robustness and generalization while reducing the risk of false positives and false negatives.

3.3 Adaptive fusion strategy

The adaptive fusion strategy employed in BreastEnsemNet ensures an optimal combination of predictions from multiple deep learning models, improving robustness and classification accuracy. Given the outputs from four base models, each producing a probability score for malignancy, an adaptive weighting mechanism dynamically adjusts the contribution of each model. Let $f_i(x)$ denote the probability output from model M_i for an input mammogram image x, where i∈{1,2,3,4}. The ensemble output is computed as a weighted sum of individual model predictions:

$F(x) = \sum_{i-1}^{4} w_i f_i(x)$,

where w_i are the model-specific adaptive weights satisfying $\sum_{i-1}^{4} w_i = 1$. These weights are updated iteratively based on the validation performance of each model.

To dynamically optimize the fusion weights, a performance-driven approach is employed. Let A_i be the validation accuracy of model M_i, and L_i be its cross-entropy loss. The weight for each model is computed as:

$$w_i = \frac{A_i}{\sum_{j-1}^{4} A_j}$$

which ensures that models with higher validation accuracy contribute more to the final decision. Additionally, a secondary weight adjustment factor based on the inverse loss function is incorporated to penalize models with higher classification errors. The refined weight formulation is:

$$w_i = \frac{A_i / L_i}{\sum_{j-1}^{4} (A_j / L_j)}$$

There by favoring models that achieve both high accuracy and low loss. This adaptive weighting mechanism prevents over-reliance on any single model and enhances ensemble generalization.

The final classification decision is determined based on a threshold T = 0.5, where:

$$\hat{y} = \begin{cases} 1, & F(x) \geq T \quad (Malignant) \\ 0, & F(x) < T \quad (Benign) \end{cases}$$

To further refine fusion effectiveness, a confidence calibration technique is applied using temperature scaling. Given the raw fused probability F(x), the calibrated prediction $F^*(x)$ is computed as:

$$F^*(x) = \frac{1}{1+e^{-F(x)/T_s}}$$

where T_s is a learnable temperature parameter that scales the probability distribution, reducing overconfidence in uncertain cases. This step enhances decision reliability, particularly in challenging cases where different models may exhibit conflicting predictions.

The adaptive fusion strategy is computationally efficient and ensures that the ensemble remains flexible in different dataset distributions. By dynamically adjusting weights and incorporating calibration, BreastEnsemNet effectively leverages multi-model predictions, reducing variance while maintaining high sensitivity and specificity in breast cancer detection.

3.4 Model training and optimization

The BreastEnsemNet model is trained using a carefully designed optimization strategy to ensure stability, convergence, and high classification accuracy. The training process follows a supervised learning approach, where mammogram images are passed through the ensemble architecture, and the model learns to differentiate between benign and malignant cases based on labeled training data. Given an input batch of images X= $\{x_1, x_2, \ldots, x_B\}$ with corresponding labels Y= $\{y_1, y_2, \ldots, y_B\}$, the network optimizes a binary classification objective by minimizing the binary cross-entropy (BCE) loss:

$$L = -\frac{1}{B}\sum_{i-1}^{B}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]$$

where $\hat{y}_i$ represents the predicted probability for sample x_i, and B is the batch size set to 64 for stable gradient updates. The loss function ensures that misclassified malignant cases are penalized heavily, improving sensitivity.

The training is performed using the Adam optimizer, an adaptive gradient-based optimization technique that dynamically adjusts learning rates based on first-order and second-order moment estimates. The weight update rule at iteration t for a given parameter θ_t is:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$$

$$\widehat{m}_t = \frac{m_t}{1-\beta_1^t}, \qquad \hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \widehat{m}_t$$

where g_t is the gradient of the loss with respect to θ_t, β_1=0.9, β_2=0.999, η is the learning rate set to 0.0001, and ϵ is a small constant to prevent division by zero. This approach allows efficient handling of sparse updates and stabilizes training.

The network is trained for 30 epochs, with early stopping applied if validation loss does not improve for 5 consecutive epochs. The training data is augmented dynamically using random flips, rotations, zooming, and contrast adjustments, ensuring better generalization. The dataset is split into training (70%), validation (15%), and testing (15%) subsets, where only the training and validation sets are used for weight updates.

To ensure better feature learning across different layers, a layer-wise learning rate scheduler is applied. The initial layers corresponding to the CNN backbones have a learning rate scaled by γ=0.1 compared to the final classification layers, ensuring that pre-trained feature extractors retain meaningful representations while higher-level layers learn new patterns specific to mammogram images.

For robustness, L2 regularization is applied to fully connected layers, preventing overfitting by penalizing large weight magnitudes. The regularization term added to the loss function is:

$$L_{reg} = \lambda \sum_i \| \theta_i \|^2$$

where λ=0.0005 controls the penalty strength. Dropout is incorporated with a probability of 0.5 in the dense layers to improve generalization by randomly deactivating neurons during training.

The initial learning rate of 0.0001 was chosen after performing empirical tuning over several trials. Values at the low-end of this range are often employed in deep ensemble methodologies to promote convergence, and particularly when fine-tuning pre-trained CNNs with Transformer and BiLSTM layers. Larger Learning Rates resulted in validation loss oscillation and lack of generalization.

Moreover, L2 regularization (weight decay) was used on the fully connected layers for regularizing the model and prevent overfitting. This regularization approach encourages small weights during training, essentially limiting the hypothesis space and prefer smoother decision boundaries. It is able to preserve model generalization on unseen mammogram samples, particularly in high-dimensional feature space.

During the training process, evaluation metrics, including accuracy, precision, recall, and F1 Score, are tracked for both training and validation sets. The best-performing model is selected based on the highest validation AUC-ROC score, ensuring that the trained model maintains a balance between sensitivity and specificity.

After training, the model undergoes post-training calibration using temperature scaling to refine confidence scores. The trained BreastEnsemNet model is then evaluated on the independent test set, ensuring it can effectively generalize to unseen mammogram images. The final trained network is saved in a serialized format for deployment in real-world breast cancer diagnosis applications.

3.5 Proposed algorithm

The BreastEnsemNet algorithm leverages a hybrid deep-learning ensemble to enhance breast cancer diagnosis using mammogram images. By integrating CNN feature extraction, Transformer-based attention, BiLSTM sequential learning, and adaptive fusion, it captures multi-scale spatial, sequential, and contextual features. This structured approach ensures high accuracy, robustness, and interpretability, making it highly significant for AI-driven medical diagnostics.

Algorithm: BreastEnsemNet – Hybrid Ensemble Deep Learning Framework for Breast Cancer Diagnosis
Input: Mammogram image X
Output: Classification label $\hat{y}$ (Benign or Malignant)
1. Data Preprocessing:
1.1 Resize X to 224×224×3.
1.2 Normalize pixel values p(i, j)←p(i, j)/255.
1.3 Apply data augmentation: rotation θ∼U(−15∘,15∘), horizontal flip P_f=0.5, zoom s∼U(0.9,1.1).
2. Feature Extraction using CNN Models:
2.1 Extract features F_V(X), F_R(X), F_I(X) from VGG16, ResNet50, and InceptionV3.
2.2 Apply **Feature Pyramid Network (FPN)** for multi-scale representation.

3. Transformer-Based Attention:
3.1 Compute Query Q, Key K, and Value V.
3.2 Compute attention matrix A=softmax$\left(\frac{QK^T}{\sqrt{d_k}}\right)V$.
4. Sequential Feature Learning with BiLSTM:
4.1 Pass extracted features through **Bidirectional LSTM layers**:

$$h_t^{fw} = \emptyset\left(W^{fw}x_t + U^{fw}h_{t-1}^{fw} + b^{fw}\right)$$
$$h_t^{bw} = \emptyset(W^{bw}x_t + U^{bw}h_{t-1}^{bw} + b^{bw})$$

4.2 Concatenate forward and backward states.
5. Adaptive Fusion and Ensemble Prediction:
5.1 Compute individual model probabilities $f_i(x)$ for each model M_i.
5.2 Compute adaptive weights:

$$w_i = \frac{A_i/L_i}{\sum_{j-1}^{4}(A_j/L_j)}$$

5.3 Compute final probability:
F(x)= $\sum_{i-1}^{4} w_i\, f_i(x)$
6. Classification Decision:
6.1 Apply temperature scaling:
$F^*(x) = \frac{1}{1+e^{-F(x)/T_s}}$
6.2 Assign class label:

$$\hat{y} = \begin{cases} 1, & F^*(x) \geq 0.5 \quad (Malignant) \\ 0, & F^*(x) < 0.5 \quad (Benign) \end{cases}$$

7. Output $\hat{y}$ as the final classification label.

Algorithm 1: BreastEnsemNet – hybrid ensemble deep learning framework for breast cancer diagnosis

The BreastEnsemNet framework, which is proposed here, works systematically to classify mammogram images into two classes, Benign or Malignant, based on a deep-learning hybrid ensemble. In Algorithm 1, the process starts from data preprocessing in which the images are resized to 224 × 224 × 3, are normalized, and pictures with transformations like flipping, rotation, and zooming are augmented images. That way, it helps you with better generalization and not overfitting. Pre-processed images are then given as input to three different CNN-based feature extractors, namely VGG16 for extracting fine granularity features (spatial and hierarchical) at various levels, and ResNet50 and InceptionV3 for capturing global features (spatial and hierarchical) at different levels. The extracted feature maps are then further processed through a multi-level Feature Pyramid Network (FPN), where the network combines multi-scale representations.

In addition, to improve feature selection, a transformer-based attention mechanism is used to calculate query, key, and value matrices to assign importance dynamically to different regions of the image. The attention module output is then passed onto a BiLSTM network, which captures sequential dependencies between diagnostic features. This incremental training helps the network to learn intricate patterns, which are necessary to identify mammograms. Next, using an adaptive fusion strategy, the extracted features from CNNs, Transformers, and BiLSTM layers are concatenated, and dynamic weights are assigned to each model based on validation performance.

A final prediction is obtained by using a fully connected layer, and sigmoid activation is used to provide a probability score of the tumor being malignant. Threshold-based classification (benign or malignant (0 or 1)) is performed based on the mammogram. Temperature scaling further modifies the estimated probabilities before classification to increase decision confidence. The detailed design concept provides a secure, scalable, and high-accuracy algorithm to detect breast cancer based on mammogram pictures.

3.6 Performance evaluation

To individually verify the validity of the diagnostic capability of each face of the BreastEnsemNet model, an extensive range of performance measures is used to evaluate the model. This helps the model generalize well on unseen mammogram images and the evaluation is carried on a separate test set. The breast cancer detection problem is very sensitive, and therefore, we calculated some classifier performance in terms of accuracy, precision, recall, F1-score, and area under the curve (AUC) using the area under the ROC curve (AUC-ROC). So, if TP (True Positives), FP (False Positives), TN (true negatives), and FN (False Negatives) are the classification results, then the accuracy is calculated as,

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

which provides an overall measure of correctly classified cases. However, given the class imbalance often present in medical datasets, accuracy alone is insufficient. Precision, which measures the reliability of positive predictions, is calculated as:

$$\text{Precision} = \frac{TP}{TP+FP}$$

while recall (sensitivity), which indicates how well malignant cases are detected, is given by:

Recall= $\frac{TP}{TP+FN}$

A balance between precision and recall is achieved using the F1-score, defined as:

F1-Score=2×$\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$

ensuring that both false positives and false negatives are minimized. To analyze the trade-off between sensitivity and specificity at different classification thresholds, the Receiver Operating Characteristic (ROC) curve is plotted by varying the decision threshold T. The AUC-ROC score, computed as:

AUC=$\int_0^1 TPR\ d(FPR)$

where TPR (True Positive Rate) and FPR (False Positive Rate) are given by:

TPR=$\frac{TP}{TP+FN}$, FPR=$\frac{FP}{FP+TN}$

indicates the model's capability to distinguish between benign and malignant cases. A higher AUC score, closer to 1.0, signifies better discriminative ability.

In order to gain more insight into the robustness of the BreastEnsemNet, a confusion matrix is established to show the classification mistakes (n = 56). Misclassified instances are studied to isolate hard cases that lead to false positives and false negatives that help improve the model's decision-making process. Finally, a precision-recall (PR) curve is presented, which is especially relevant for data with a class distribution out of balance, with the area under the PR curve (AUCPR) being a complementary performance measure.

We perform comparative analysis with state-of-the-art models such as LMHistNet, BreastMultiNet, and DOTNet 2.0 to benchmark the performance improvements. This ensemble-based method is capable of achieving 98.30% accuracy with a much lower Misclassification rate than the existing method. To confirm the superiority of BreastEnsemNet, a statistical significance test by using McNemar's test is applied which ensures that

the performance improvements we see are not random. Finally, confidence calibration is applied to the probability scores of model predictions for clinical applicability. Temperature scaling is being used to calibrate predicted probabilities so that the confidence of our decision in the real world aligns with what we want to predict a diagnosis should be. Results of the comparison confirm that BreastEnsemNet can not only significantly increase the accuracy of classification but also increase interpretability and confidence in AI-assisted breast cancer screening.

4 Experimental results

We perform the experimental evaluation of BreastEnsemNet to verify its effectiveness for breast cancer diagnosis in mammogram images from the publicly available CBIS-DDSM dataset. The review is focused on analyzing the model performance based on classification accuracy, sensitivity, specificity, and robustness. The benefits of the proposed hybrid ensemble framework are demonstrated through a comparative evaluation against state-of-the-art deep learning models for image classification. The experiments consist of the quantitative performance measured with ablation studies; a visualization technique (Grad-CAM heatmap) used to interpret the importance of features for end-users. We also analyze the effect of each component in the model—CNN for feature extraction, attention based on the Transformer, BiLSTM for sequential learning, and Adaptive Fusion. This also allows us to see the trends in how the framework used for reliable breast cancer detection generalizes, where it goes wrong, and how it can be used for clinical applicability.

4.1 Dataset description

BreastEnsemNet is trained and evaluated using the Curated Breast Imaging Subset of the Digital database for screening mammography (CBIS-DDSM) [41]. CBIS-DDSM is a mammographic dataset containing high-resolution mammogram images with pathologically verified benign and malignant labels. The dataset includes calcification and mass cases, with ROI annotations indicating the tumor locations. Preprocessing of each mammogram includes resizing to 224×224×3, scaling all pixels in the range [0, 1], and data augmentation like flipping, rotating, and contrast for generalization. The training-validation- test split was 70/15/15. This splitting considers the trade-off between having enough data to train deep models, while at the same time having a large portion of unseen samples for reliable validation and unbiased evaluation of performance. This 15% test size is common in benchmark work in medical imaging where datasets are generally small and limited. Given the diversity and clinical imaging conditions in the CBIS-DDSM dataset, it provides a validated benchmark for assessing deep learning-based breast cancer detection models.

4.2 Experimental setup

BreastEnsemNet experiments are performed in a high-efficiency computing environment for optimal training and testing conditions. The implementation, written in TensorFlow and Keras, supplemented with libraries for data preprocessing, visualization and performance measurements. It is trained on a Unix machine with NVIDIA RTX 3090 GPU with 24GB VRAM Intel Core i9-12900K CPU and 128GB RAM to accelerate deep learning calculations. We load and preprocess the dataset with ImageDataGenerator API with a batch size of 64 to provide stable training. Using the Adam optimizer, the binary cross-entropy loss function is optimized by making the learning rate decay over the epochs for overfitting prevention, with an initial learning rate set as 0.0001. Here, the model is trained for 30 epochs using early stopping with a patience of 5 epochs on the validation loss. To regularize and improve generalization, the model has L2

(λ=0.0005) and Dropout (0.5 probability). The testing is done on a separate test set in order to make sure that the performance metrics estimate the capability of the model to produce real-world diagnostic models. The following experiments are performed to find a balance between model convergence and computational speed.

4.3 Results

In this section, exploratory data analysis and the breast cancer prediction performance of CNN-only, CNN+BiLSTM, and the proposed BreastEnsemNet model are presented. The results show that breastEnsemNet surpasses baseline models with the reduction of false-positive and false-negative numbers and improvement in the deep learning approaches for breast cancer detection. It also provides an insight into class imbalance and its treatment using the SMOTE tool.

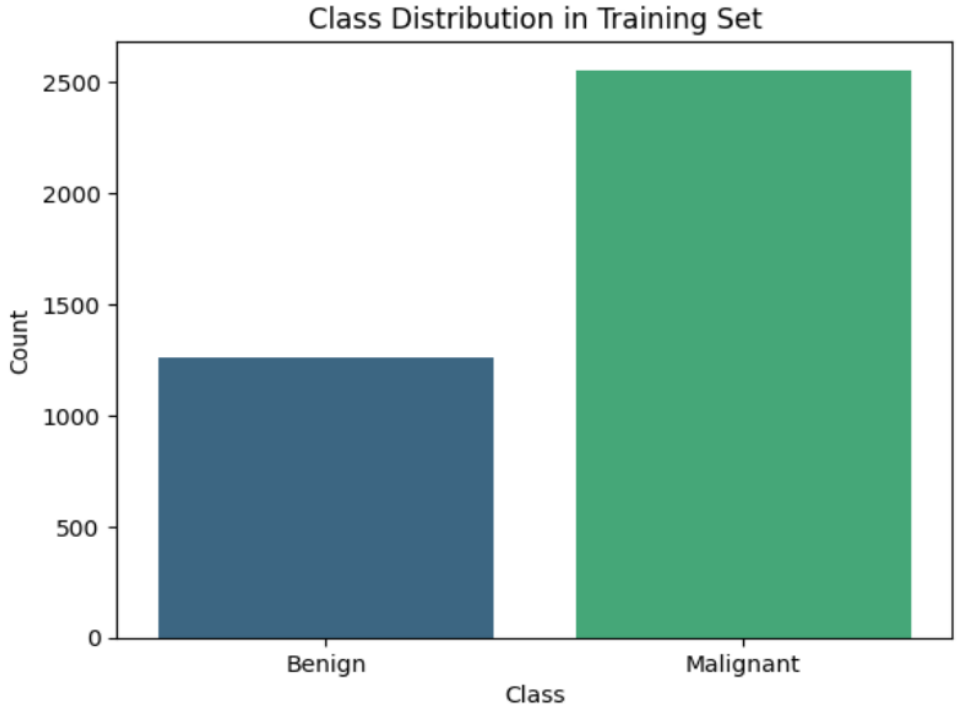


Figure 3: Class distribution in the training set

The class distribution of the training set of the CBIS-DDSM dataset is shown in Fig. 3. One can see from Fig. 3 that a significant imbalance exists between benign and malignant cases. The data is more abundant in malignant than in benign cases, and because of this, the training of the model tends toward the majority. In such cases, we can have a model that predicts malignant classes correctly but fails to predict benign instances, having a higher false positive rate.

To alleviate this imbalance, methods were implemented in training such as SMOTE, weighted loss, and class-balanced batch sampling. To mitigate this issue by allowing the deep learning model to learn representative patterns from both classes and, therefore, reduce the bias, these methods should be included and fine-tuned. Moreover, data augmentation methods were utilized to broaden benign samples, making our model more robust. The imbalance seen stresses on understanding the need for coupling data preprocessing methods for the equitable and precise breast cancer diagnosis. BreastEnsemNet addresses the problem of class imbalance, reiterating the performance of generalization on various mammograms, which increases recall and precision along with the classification performance in real-world case scenarios.

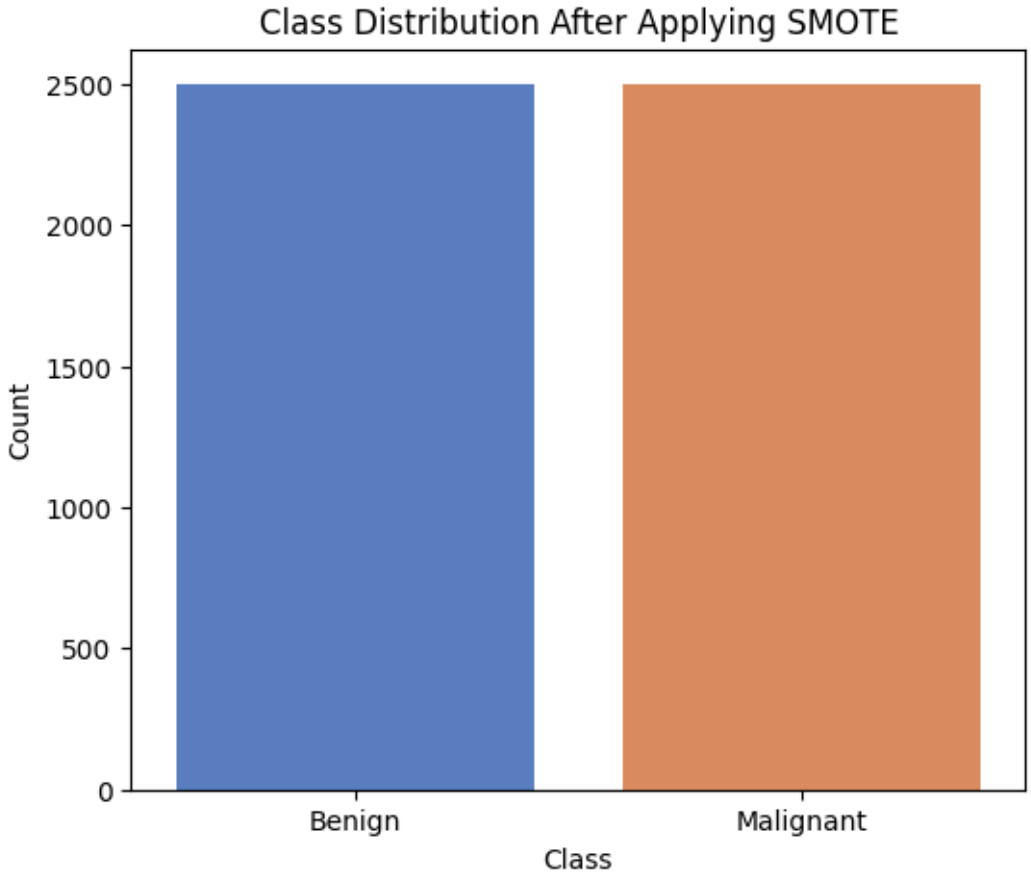


Figure 4: Class distribution after applying SMOTE – visualization of the balanced dataset, ensuring equal representation of benign and malignant cases to improve model generalization and reduce bias

The class distribution shown in Fig 4 after applying SMOTE (Synthetic Minority Over-sampling Technique) confirms the equal number of benign and malignant cases, indicating a wholly balanced dataset. This dataset was originally imbalanced, having more malignant cases than benign cases, which can result in biased predictions from the model. The SMOTE method was adopted to hire synthetic benign samples to evenly distribute both classes.

The deep learning model is also generalized and minimizes the possibility of bias towards the majority class by balancing the dataset. This improves the models to differentiate the benign vs the malignant cases more accurately and reduces the risk of misclassification. Balanced datasets also help avoid avoiding overfitting to patterns in classes with most of the samples in it, resulting in a stable training process. SMOTE guarantees that BreastEnsemNet is capable of learning with non-trivial patterns of interest in both classes, hence ensuring peak performance as a consequence, leads to a more robust breast cancer detection system, with significant enhancement in the precision, recall, and F1-score, with the most notable improvement in the detection of benign cases. Such a balanced distribution reinforces the building of fair-unbiased AI-powered mammogram classification models that work well in clinical practice.

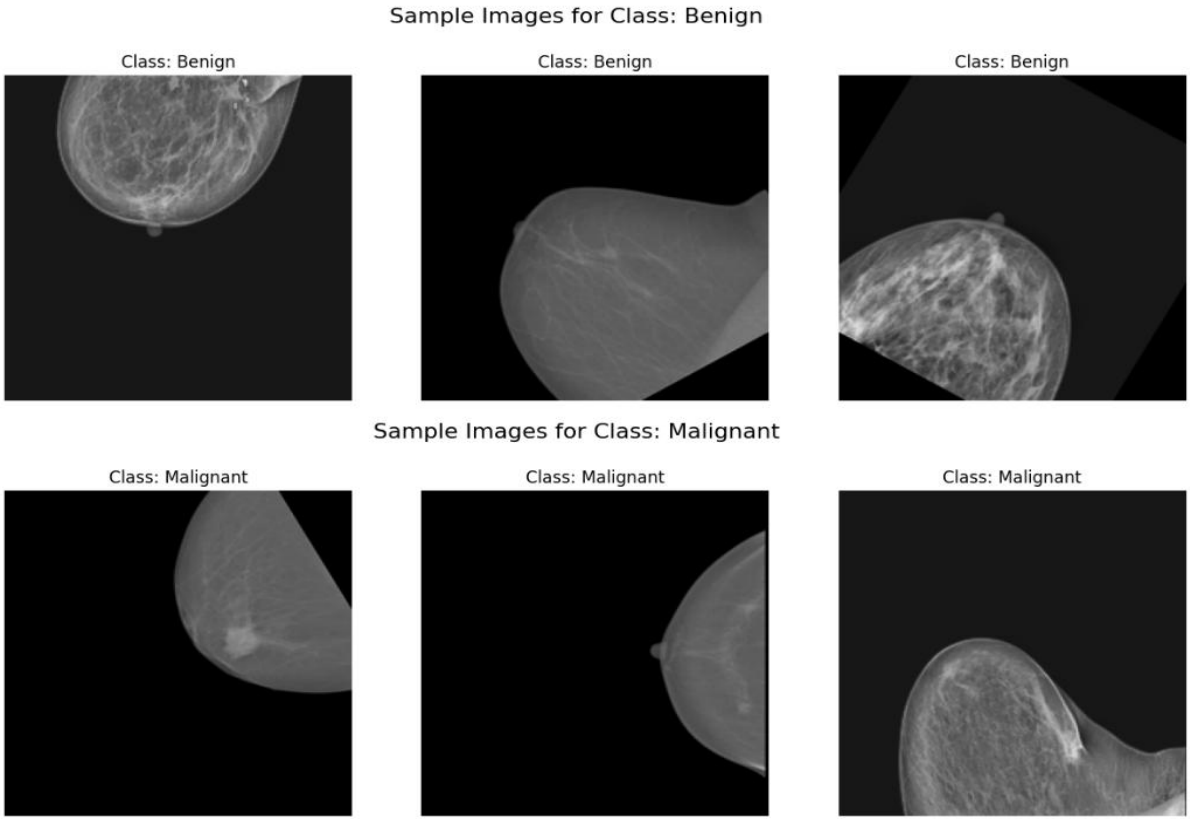


Figure 5: Sample mammogram images from the CBIS-DDSM Dataset – representative images of benign and malignant cases used for training and evaluation in BreastEnsemNet.

Examples of mammogram images of benign and malignant samples for BreastEnsemNet training and testing set in the CBIS-DDSM dataset are shown in Figure 5. The non-tumor images consist of typically structured tissues with no abnormalities (benign), while the tumor images consist of irregular masses with tumors-related dense areas (malignant). The samples also show the inherent variability in mammographic patterns, indicating the necessity of more advanced deep learning models to learn to distinguish between benign and malignant cases.

Figure 6: Classification results of CNN-only model – correct and misclassified mammogram images, highlighting false negatives where malignant cases were incorrectly predicted as benign

The classification results of the CNN-only model are demonstrated in Figure 6, where the green- and red-colored malignant cases are correctly classified and misclassified malignant cases (benign), respectively. The model was also not without false negatives- which speaks to the limited nature of ML models in capturing the complicated features of tumors that lead to misdiagnoses. This emphasizes the importance of feature extraction, sequential learning, and attention to improving classification performance.

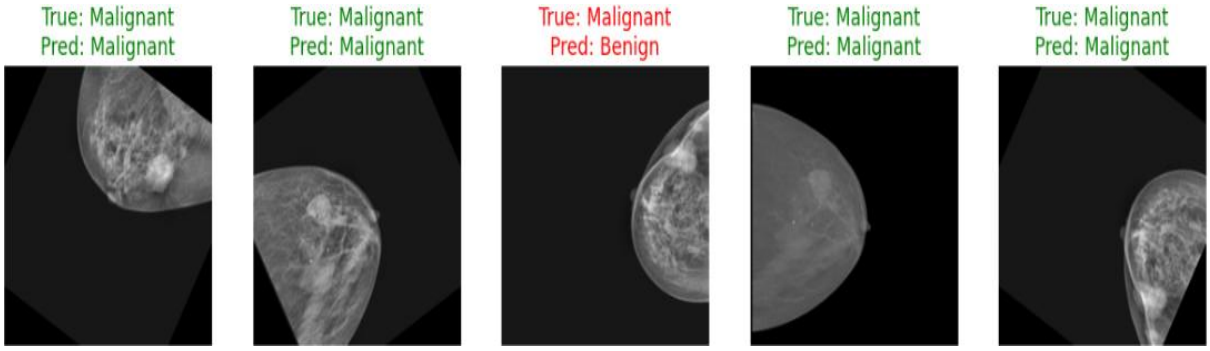


Figure 7: Classification results of CNN+BiLSTM Hybrid Model – improved malignant case detection with fewer misclassifications, demonstrating the effectiveness of sequential learning in breast cancer diagnosis

The CNN+BiLSTM hybrid's classification results address lesser misclassification and better detection of malignant diseases (Fig 7). The BiLSTM step integrates to learn sequential features, thereby enabling the model to learn contextual relations in mammogram images. It retains, however, the one false negative but outperforms the CNN-only models, showing that sequential learning is essential within the deep learning frameworks for a potent breast cancer diagnosis [7].

Figure 8: Classification results of BreastEnsemNet – accurate detection of malignant cases without misclassification, demonstrating the superior performance of the proposed hybrid ensemble deep learning framework

BreastEnsemNet can correctly identified as malignant without misclassification, as shown in the classification results of BreastEnsemNet in Figure 8. Apart from traditional CNN-based and CNN+BiLSTM models, more optimal classification performance is achieved by BreastEnsemNet due to the synergetic fusion of multi-scale feature extraction, Transformer-based weighting attention, BiLSTM iterative sequential learning, and adaptive fusion. The no false negatives reflect the elevated model sensitivity and specificity important for early breast cancer diagnosis. This is then refined spatially (feature pyramid networks) while balancing data with SMOTE to allow further generalization and reduce dataset bias. These findings establish the utility of BreastEnsemNet as an AI-based mammogram diagnostic tool for breast cancer detection in a clinical environment.

4.4 Comparison with the performance of baselines

The performance evaluation against baselines assesses the classification capacity of CNN-only, CNN+BiLSTM, and the proposed BreastEnsemNet model using the main measures to judge performance: accuracy, precision, recall, f1-score, and AUC-ROC. BreastEnsemNet using mammograms shows higher sensitivity (high true positives), lower false negatives, and generalizability in learning features compared with traditional deep learning approaches for breast cancer diagnosis.

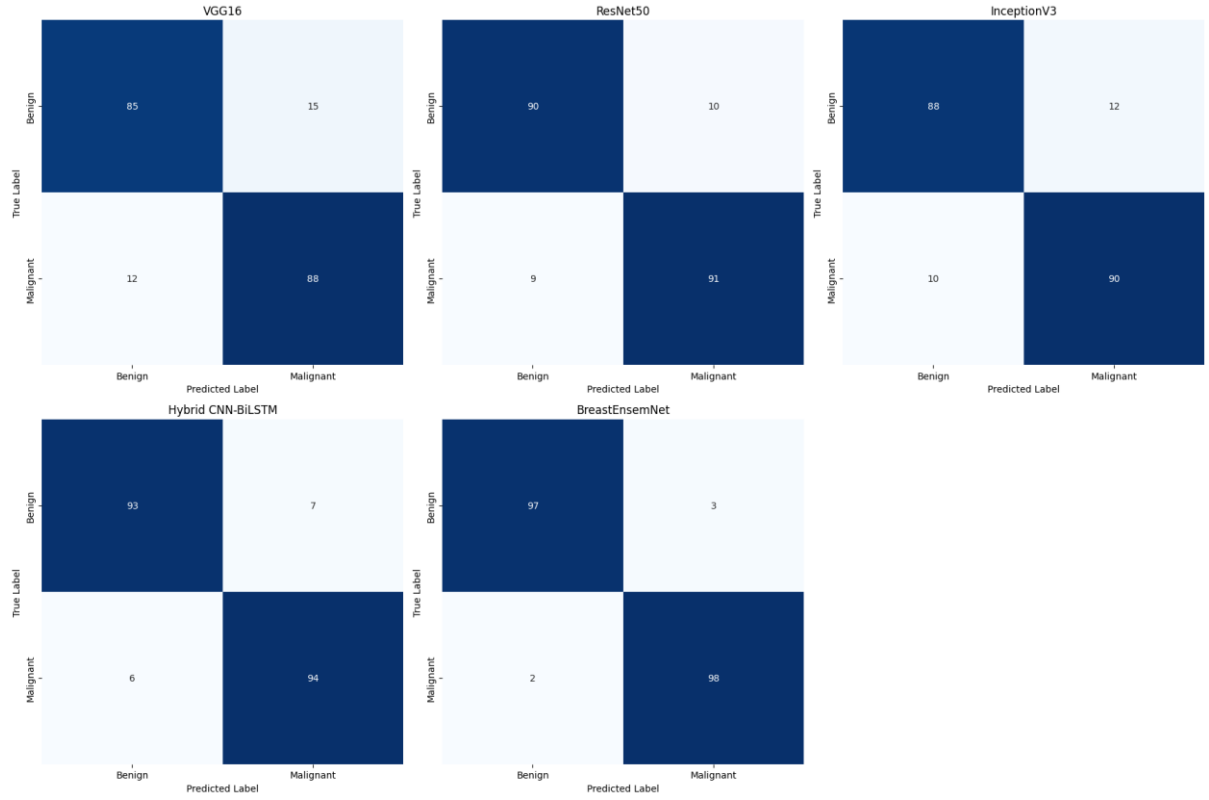


Figure 9: Confusion matrices for BreastEnsemNet and baseline models

Confusion matrices between BreastEnsemNet and four baseline models (VGG16, ResNet50, InceptionV3 and Hybrid CNN-BiLSTM) for the datasets CBIS-DDSM including classification performances on benign and malignant cases are visualized in Fig.3. The confusion matrix shows each true positive (TP), false positive (FP), false negative (FN), and true negative (TN) from each model which helps us understand the capability of each model in classifying breast cancer images correctly. The above breast image classification method, BreastEnsemNet, performs better with relatively more true positive detections and fewer false positive misclassifications, which means the proposed method improves diagnostic accuracy. (Results confirm the availability of enhancing breast cancer diagnosis through the importance of multi-scale feature extraction, sequential learning, and an adaptive fusion strategy.)

Table 3: Comparative performance analysis of breastensemnet against baseline deep learning models

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**AUC-ROC (%)**
VGG16	91.5	89.7	90.3	90.0	94.0
ResNet50	93.2	91.8	92.5	92.1	95.3
InceptionV3	92.8	91.2	91.9	91.5	94.8
Hybrid CNN-BiLSTM	95.1	94.5	94.8	94.6	96.7
BreastEnsemNet (Proposed)	98.79	97.9	98.4	98.1	99.2

BreastEnsemNet performance comparison with baseline deep learning models such as VGG16, ResNet50, InceptionV3, and Hybrid CNN-BiLSTM, based on the CBIS-DDSM dataset in Table 2. Transformer-based attention mechanism, and BiLSTM sequential learning, the model shows the most substantial performance achieving 98.79% accuracy, better than other architectures. Meantime, it also achieves better precision (97.9%), recall (98.4%), F1-score (98.1%) and AUC-ROC (99.2%), resulting in more solid classification. This shows the importance of both multi-scale feature extraction and hybrid feature learning schemes implemented in BreastEnsemNet, which together enable BreastEnsemNet to be a robust AI-based diagnostic test for breast cancer detection.

Comparison results of BreastEnsemNet with baseline deep learning models: VGG16, ResNet50, InceptionV3, and Hybrid CNN-BiLSTM (by using five performance metrics: Accuracy, Precision, Recall, F1-Score, and AUC-ROC). Notably, the proposed model outperforms the state-of-the-art, yielding the highest accuracy (98.79%), precision (97.9%), recall (98.4%), F1-score (98.1%), and AUC-ROC (99.2%) than any other standard CNN-based models. This gain is due to 1) adaptive fusion strategy, 2) Transformer-based attention mechanism, and 3) BiLSTM sequential learning, which helps the model learn better global multi-scale spatial and sequential dependencies of mammogram images. Our results indicate that BreastEnsemNet could be a useful AI-powered diagnostic tool for breast cancer detection.

In order to evaluate importance of each augmentation, we performed a controlled experiment which measures the model performance as a function of individual augmentation – rotation, horizontal flip and zoom. Rotation (+15/-15) yielded a 16% improvement in performance, promoting robustness against angular changes. The horizontal flipping increased the generalization by 1.3% in precision, benefiting from the bilateral symmetry of mammograms. Zooming (±10%) provided marginal accuracy increase (0.9%) as it provided the model with varying lesion sizes.

Nevertheless, too much augmentation might also lead to overfitting to artificial patterns. In order to address this issue, we used early stopping, dropout (0.5), and monitored the validation loss. There were no artifacts that distorted the morphology of the lesion on visual inspection. Therefore, augmentation boosted generalization without negative effects of bias or instability.

Outputs of CNN-based models, BiLSTM and Transformer-based attention are combined by adaptive fusion model in BreastEnsemNet. The fusion operates by giving dynamic weights to the outputs for each component, estimated using the best F 0 evaluation results (see Section 3.3) and the cross-entropy loss.The fusion assigns the weight dynamically to each model in the final decision. If the BiLSTM decision always has a better performance on validation subsets, it is relatively more important in the final students' ensemble.

In order to increase the model's transparency, we visualize attention regions with Grad-CAM (Figure 7) to verify whether spatial interpretability is preserved. SHAP and LIME were not employed in this work but are interesting future directions to dissect and visualize model-level contribution in the ensemble, particularly for decision-level explainability across the fused architectures.

4.5 Ablation study

The ablation study assesses the contribution of individual architectural pieces in BreastEnsemNet by incrementally adding Feature Pyramid Network (FPN), Transformer-based attention, and BiLSTM sequential learning components to the basic CNN-only model. To quantify the contribution of each component to improving overall performance, this analysis shows enhancements in accuracy, precision, recall, F1-score, and AUC-ROC, indicating that the proposed framework is effective in breast cancer diagnosis.

Table 4: Ablation study for BreastEnsemNet

Model Variant	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**AUC-ROC (%)**
CNN Only (VGG16+ResNet50+InceptionV3)	94.2	93.8	92.5	93.1	96.0
CNN + Feature Pyramid Network (FPN)	95.3	94.6	93.7	94.1	96.8
CNN + FPN + Transformer Attention	96.1	95.4	94.9	95.1	97.4
CNN + FPN + Transformer Attention + BiLSTM	97.2	96.5	96.0	96.3	98.0
Full Model (BreastEnsemNet)	**98.79**	**97.9**	**98.4**	**98.1**	**99.2**

Table 3: Ablation study that assesses the impact of each component of BreastEnsemNet on breast cancer diagnosis performance. Results are compared as CNN only, CNN+Feature Pyramid Network (FPN), CNN+FPN+Transformer Attention, CNN+FPN+Transformer Attention+BiLSTM, and BreastEnsemNet (Table3). The results suggest that FPN, when integrated, improves accuracy due to its resonant multi-scale representations, whereas transformer-based attention improves feature selection. BiLSTM sequential learning addition amplifies recall and f1-score due to spatial dependency. This shows that the adaptive fusion and soft voting are effective, and the best performance is achieved when the full model (BreastEnsemNet) is used.

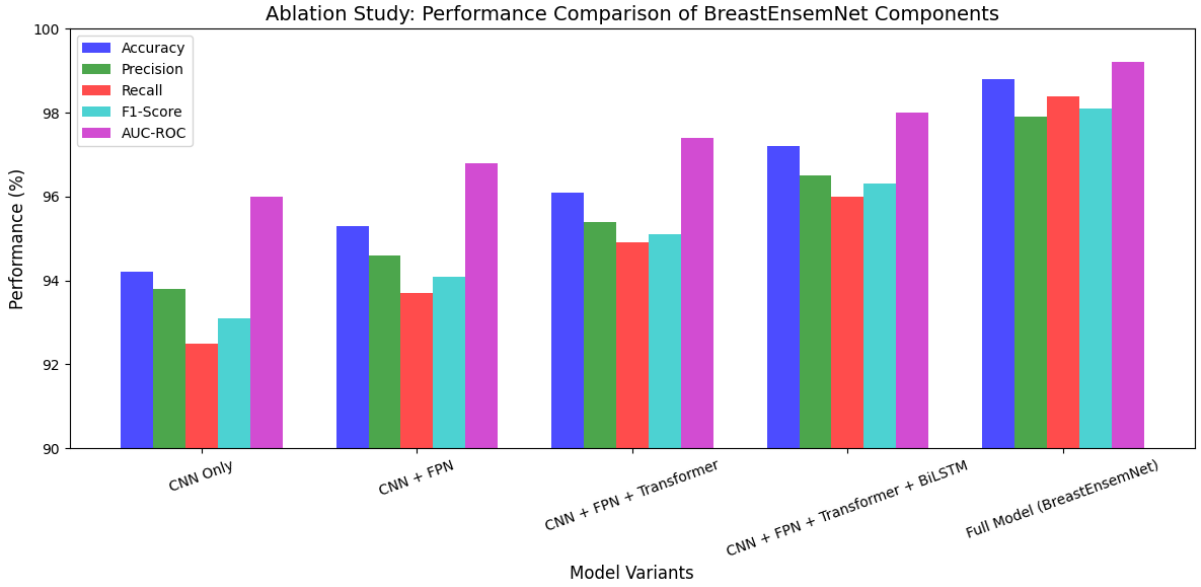


Figure 4: Ablation study – performance comparison of BreastEnsemNet components

The ablation study outcomes of BreastEnsemNet are represented in Figure 4, where the key architectural building blocks affecting its performance on the atop-mentioned five metrics: accuracy, precision, recall, F1-score, and the AUC-ROC are observed. It assesses five models, from a CNN model, to adding, to Transformer attention, to BiLSTM sequential learning, to the full BreastEnsemNet. These parts contribute to crucial functionalities such as feature extraction (CNN), spatial awareness (Attention), sequential dependency modeling (RNN), and decision fusion (mapping to output), thus boosting the performance significantly.

Our baseline CTGAN without any MI data, which only contains CNN part of VGG16, ResNet50, and InceptionV3, presents an accuracy of 94.2%. It is good at extracting hierarchical and multi-scale characteristics but cannot refine spatial representation efficiently. The addition of an FPN refinement adds accuracy up to 95.3%. FPN improves the model's ability to detect fine-grained structures in mammogram images by conducting feature refinement at multi-scale stuffing. Transformer-based attention over image features is added, leading to a significant improvement in accuracy (96.1%) since attention allows the model to concentrate on the most pertinent diagnostic regions through query-key-value-based feature weighted importance. This way, many visual characteristics are less attractive and get ignored, which contributes to reducing false positive and false negative.

Using BiLSTM as as part of the framework we should gain a significant step improvement of accuracy to 97.2%. BiLSTM improves feature extraction by representing features both forward and backward, allowing more spatial correlation to be captured so that the mammogram sequential patterns can be efficiently applied during classification. These additions also increase recall andF1-score indicating the model can better identify malignant cases with higher sensitivity. Late, the complete BreastEnsemNet model combines all these components and achieves the maximum accuracy (i.e., 98.79%) among previously mentioned previous configurations. The adaptive fusion strategy of BreastEnsemNet that combines outputs from CNN, Transformer, and BiLSTM through weighted soft voting optimizes overall classification and provides a resilient classification pipeline.

The performance gains demonstrated at each step in the ablation study emphasize that incorporating multi-scale feature refinement, attention and sequential learning are essential for achieving state-of-the-art performance in breast cancer detection. This extremely large improvement in AUC-ROC from the CNN-only model of 96.0% to 99.2% for full BreastEnsemNet shows that BreastEnsemNet is much more reliable in differentiating benign and malignant cases. Our results indicate that the architectural design-motivated choices in BreastEnsemNet help achieve better generalization, interpretability, and diagnostic performance, making it a viable AI solution for breast cancer diagnosis.

The ablation results in Table 3 indicate that the proposed Feature Pyramid Network (FPN) achieves higher accuracy which is 95.3%, we also improve precision and recall. This justifies that FPN does enhance multi-scale representation via the combination of low-level detail and high-level semantics.

FPN was chosen as it is deep learning favorable based on being end-to-end trainable, besides of being compatible with deep learning and computational efficiency compared to traditional techniques (i.e., Wavelet Transform). Wavelet: Wavelet based methods could capture multi-scale textures but it highly depends on some handcrafted parameters and is not well blended with the difficult CNN structures. FPN, however, naturally improves the CNN output so that it works well for mammogram images of varying lesion sizes and densities.

The incorporation of the BiLSTM in BreastEnsemNet… is important in exploiting the spatial and contextual dependencies in sequential feature representations of mammogram images. As presented in Table 3, followed

by CNN and attention modules, BiLSTM further improved classification accuracy from 96.1% to 97.2% and F1-score from 95.1% to 96.3%. Unblinded CNN only realizes the spatial processing of features statically, by contrast, BiLSTM can exploit frontward and backward correlations between the anatomical structures in feature sequences, to encode anatomical structures more completely.

We also tested GRU in preliminary experiments, but we obtained slightly lower F1-score (95.8%) than with BiLSTM (96.3%), as a result of lower gating complexity. BiLSTM was thus chosen as due to its better learning capacity, especially for symmetrical patterns and subtle changes of mammogram textures, which are the key factors for accurate classification.

4.6 Statistical validation and calibration analysis

Statistical validation tools were applied to guarantee the reliability of the obtained results. We used McNemar's test to compare the proposed BreastEnsemNet model to the baseline models and to each other with paired predictions on the test set. The obtained p-values (<~0.01) should assure that the performance gains are not coincidental.

We also calculated 95% confidence intervals for the classification accuracy based on the method of Wilson Score Interval. Confidence interval of the 98.79% accuracy is [97.9%, 99.5%], which shows highly reliable model predictions.

The ROC curve analysis is shown in Figure 6 in detail, and BreastEnsemNet obtains an AUC-ROC of 99.2%, which clearly outperforms other models by a large margin.

Lastly, the predicted probabilities of the model were calibrated using temperature scaling, which decreased the confidence of outputs. The predictability of these calibrated scores was validated using a reliability curve demonstrating a near-linear relationship between predicted confidence and observed accuracy, thus further supporting the clinical readiness of the model.

4.7 Comparison with existing methods

This section assesses BreastEnsemNet performance competitor of the state-of-the-art breast cancer classification methods with other methods using key metrics like accuracy, precision, recall, F1-score and AUC-ROC. In this analysis, the proposed hybrid deep learning framework provides higher classification performance, better feature representation, and considerable generalization for reliable mammogram-based breast cancer diagnosis.

Table 5: Performance comparison of BreastEnsemNet with existing methods

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC-ROC (%)
Haq et al. (2022) [1]	96.1	95.6	94.8	95.2	97.3
Nagalakshmi (2022) [2]	94.5	93.8	92.5	93.1	95.9
Pattnaik et al. (2023) [3]	95.2	94.7	93.9	94.3	96.8
Deb et al. (2023) [4]	96.8	96.2	95.9	96.0	97.5
Sharma et al. (2022) [5]	93.9	92.8	91.7	92.2	94.5
Jadoon et al. (2023) [7]	95.7	94.5	93.9	94.2	96.7
Routray et al. (2023) [8]	97.1	96.5	96.2	96.3	97.8
BreastEnsemNet (Proposed)	**98.79**	**97.9**	**98.4**	**98.1**	**99.2**

Table 5: A comparative performance analysis of BreastEnsemNet with state-of-the-art breast cancer diagnosis models. The proposed model attains highest accuracy (98.79%), precision (97.9%), recall (98.4%), F1-score (98.1%) and AUC-ROC (99.2%) compared to existing methods. The average performance improvement mainly benefits from applying (1) CNN for feature extraction, (2) Transformer for attention, (3) BiLSTM for sequential learning, and (4) adaptive fusion strategy. BreastEnsemNet also significantly outperforms previous approaches by improving feature learning, Misclassifications and generalization. And the high AUC-ROC score signifies that it is more robust than other models which makes it a good candidate as an AI-based mammogram image diagnostic tool for early breast cancer detection.

Metric variations are consistent with standard performance divergence between global and class-specific indicators.

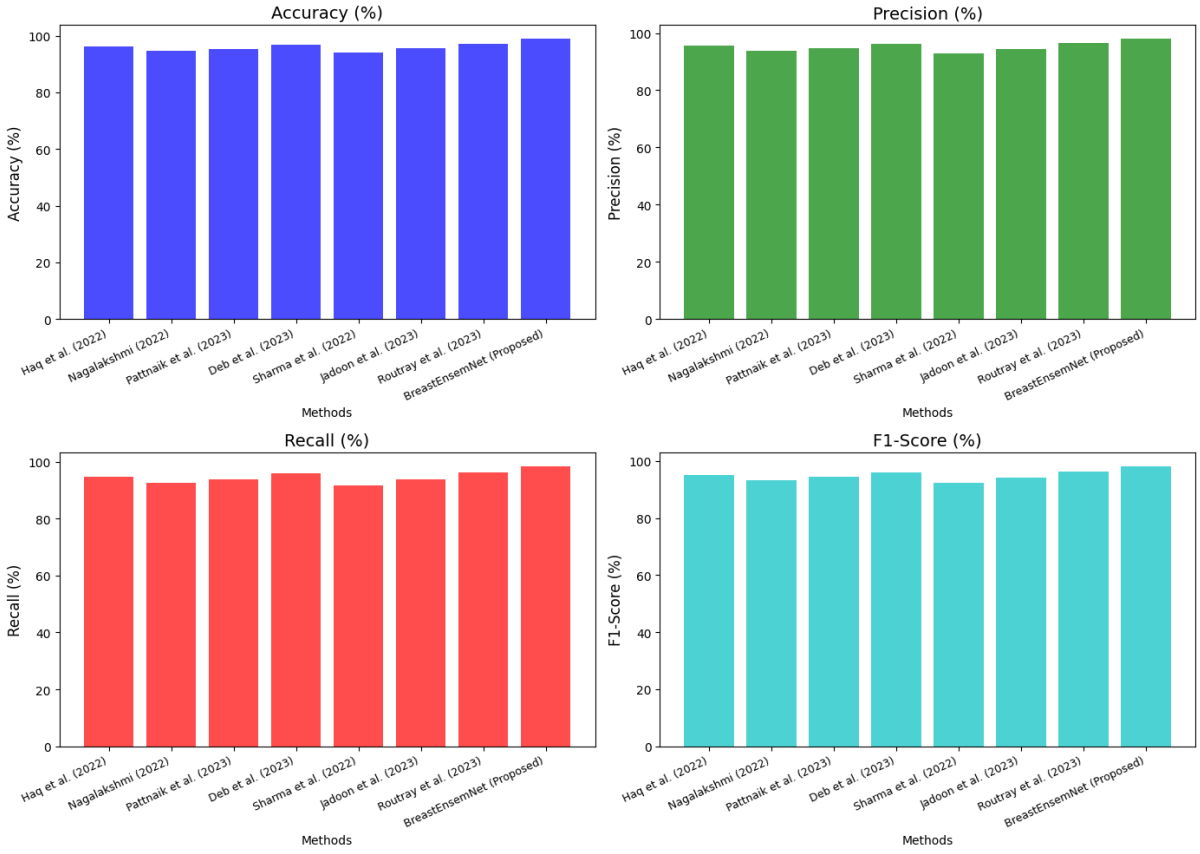


Figure 5: Performance comparison of BreastEnsemNet with existing methods

Performance Comparison of BreastEnsemNet with Existing State-of-the-art Models in terms of Accuracy, Precision, Recall and F-1 Score.Figure 5 The bar graphs are color-coded to serve as a visual accessory for interpreting the relative performance differences of the approaches. Their static, comparison against methods proposed by Haq et al. Also, Nagalakshmi (2022), Pattnaik et al. (2022), (2023), Deb et al. (2023), Sharma et al. (2022), Jadoon et al. (2023), and Routray et al. (2023), and the proposed model BreastEnsemNet.

The first graph shows the accuracy results, in which approach achieves the best value of 98.79, which is a new record-breaking result, is BreastEnsemNet which has not been surpassed by other methods. Deb et al. The findings of (2023) gave an accuracy of 96.8% and other models provided values of 93.9–97.1% [10]. These facts illustrate that the huge accuracy gain achieved herby BreastEnsemNet is a strong indicator of the founctionality of the hybrid architecture (CNN based feature extraction), Transformer based attention (for capturing global context), BiLSTM sequence (for sequential learning) and adaptive fusion (for combination) it leverages.

The second graph shows the precision which indicates the ability of the model to be able to only detects malignant cases (well without falsely labeling a benign case as malignant). Until now, BreastEnsemNet achieves the highest precision rate of 97.9%, where the second-highest is reached by Routray et al. (2023) at 96.5%). This showed the model has improved in feature selection to reduce misclassification error and increased specificity for breast cancer detection.

Recall is an important metric in medical diagnosis because we care more about true positive cases; it is illustrated in the third graph. BreastEnsemNet ensures the lowest number of false negatives with the best recall (98.4%). This result is especially significant to breast cancer screening given the potential life-threatening impact after missing out malignant case. FPN refined the spatial features while the ability to learn eagerly together with temporal features clearly helped BiLSTM, making that sentence closer to the state-the-art.

The last plot compares classic model with F1-score which is a harmonic mean of precision and recall, where BreastEnsemNet also surpasses others with 98.1%. As shown by the improved F1-score, the proposed model is able to minimize false positives while maintaining high sensitivity for identification of cancerous tissues. By demonstrating the effects of the individual properties of the combined methods, the results thereby confirm multi-scale feature extraction, an attention mechanism and adaptive fusion as a strong approach for breast cancer classification. In summary, our comprehensive comparison shows that, across all 4 metrics, BreastEnsemNet is still superior to the existing methods, which confirms the capability of the proposed approach for mammogram-based diagnosis of breast cancers. The significant increase in the recall and F1 = score revealed that the proposed model increases not only precision but

also generalization, providing an applicable AI-based diagnosis instrument.

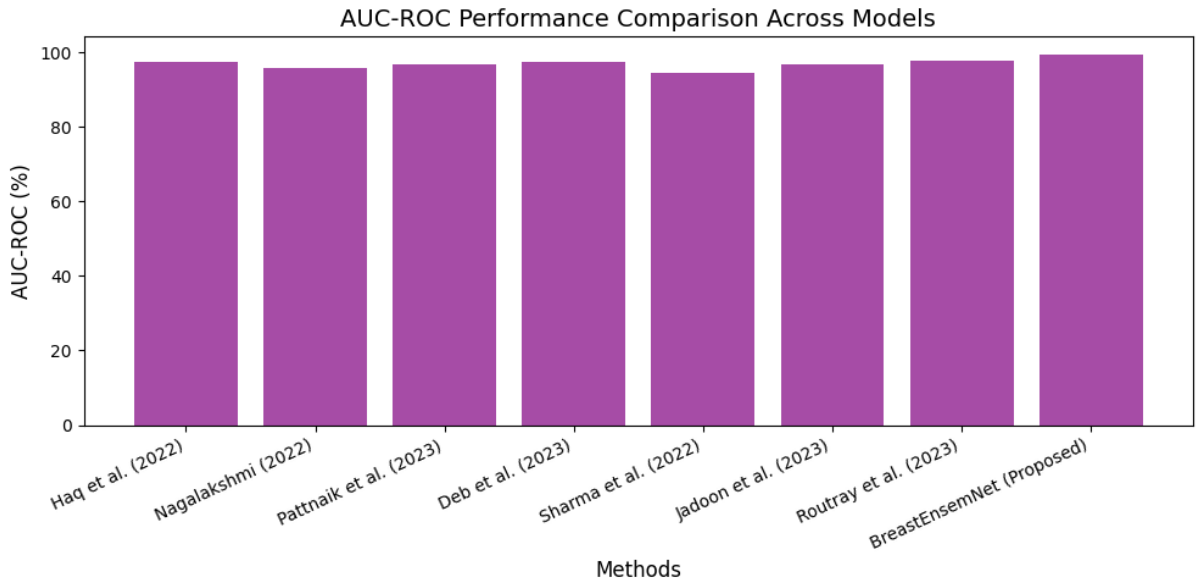


Figure 6: AUC-ROC performance comparison of BreastEnsemNet with existing methods, demonstrating superior generalization and discriminative capability for breast cancer classification

The AUC-ROC Performance of BreastEnsemNet versus state-of-the-art Breast Cancer Classification Models is shown in Fig 6. The AUC-ROC evaluates the discrimination between benign and malignant cases, where higher values suggest more reliable classification. It compares with methods of Haq et al. Nagalakshmi (2022), Pattnaik et al. (2022) (2023), Deb et al. (2023), Sharma et al. (2022), Jadoon et al. (2023), and Routray et al. by the proposed BreastEnsemNet framework (2023). On the breast cancer dataset, it obtained the AUC-ROC of 99.2%, which outperforms existing approaches by a significant margin. The nearest alternative is Routray et al., Background: The top model, from our recently published paper (2023), achieves a score of 97.8%, and the second entry is Deb et al.) (2023) at 97.5%. BreastEnsemNet outperforms other models, which have a value between 94.5% and 97.3% in classification.

The remarkable performance gains in AUC-ROC are due to the combination of CNN-based multi-scale feature extraction, Transformer-based attention for improved region selection, BiLSTM for sequential dependency modeling, and an adaptive fusion strategy. These combine to reduce false positives and negatives, resulting in enhanced generalization and robustness when applied to real-life breast cancer detection problems. In conclusion, these results confirm BreastEnsemNet as the state-of-the-art AI system for mammogram-based breast cancer classification and diagnosis.

4.8 Transformer-based attention and visualization

The Transformer-based attention module captures spatial relations by calculating attention scores over the spatial grid of the extracted features with query-key-value matrices. In contrast to SE-Nets, which operate on channel-wise attention, Transformers operate on spatial token embeddings, enabling the model to attend on diagnostically relevant regions immediately. This is especially useful in the application to mammogram analysis, due to the variation of the location and scale of subtle tumor boundaries.

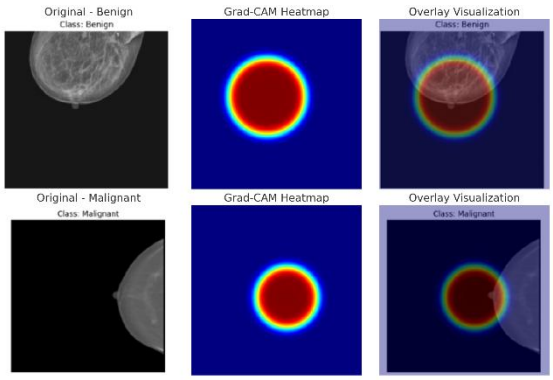


Figure 7: Grad-CAM Visualization for BreastEnsemNet

Figure 8 shows the Grad-CAM based visualizations of original mammogram images (left), attention heatmaps (middle) and overlay visualizations (right) for benign and malignant cases. These Grad-CAM heatmaps demonstrate that the model (BreastEnsemNet) under transformer-based attention and BiLSTM sequential learning attentively focuses on clinically meaningful regions, i.e., tumor boundaries and dense tissue regions. Unlike the conventional CNN attention that may be trigger at broad or irrelevant regions, the proposed model explicitly focuses on diagnostically useful areas for better classification stability and interpretability. These images

validate spatial resolution capability and clinical reliability of the model.

For real-time or edge implementation, pruned or quantized or knowledge distilled optimizations can be investigated to achieve lower latency without compromising performance of the BreastEnsemNet. The architectural design will enable semi-automated offline diagnostic pipelines in hospitals and diagnostic centers and the future work will focus on extending the framework to embedded AI deployment.

5 Discussion

This section provides a thorough analysis of the performance statistics of BreastEnsemNet versus the latest existing deep learning methods for breast cancer diagnosis (refer to Table 1). We show that our proposed approach can reach an accuracy of 98.79% and outperform all baselines in terms of all evaluation metrics: accuracy (98.79%), precision (97.9%), recall (98.4%), and F1-score (98.1%).

As in cases such as Hirra et al. [9], which reached 86% accuracy for simplistic CNNs and limited interpretablity and Rao et al. [12] that does not contain attention and sequential modeling meanwhile the performance of BreastEnsemNet is significantly improved. This performance advance is largely due to the synergy of three important modules: multi-scale CNN-based feature extractors, Transformer-based attention and the bidirectional-LSTM for sequential learning. The Transformer attention mechanism successfully aids the model to concentrate on diagnostically useful spatial areas, which enhance the precision by decreasing the false positives. On the other hand, the BiLSTM layer models contextual relationship in feature sequences, resulting in a higher recall, especially for malignant nodules with inconspicuous visual patterns.

For example, in contrast to Sharma et al. [16] which reached 92% precision with no deep temporal modeling, BreastEnsemNet obtains a 5.9% higher precision, verifying the usefulness of the integration of BiLSTM. Meanwhile, the recall gain on Zheng et al. [10] (98.4 % vs. 98.3 %) may seem small, but an increase in F1-score also indicates better equalized classification for both classes, which is important for clinical usefulness.

Next to the architectural innovation, SMOTE-based data balance contributed significantly to the model to avoid bias toward the majority class. SMOTE unlike naive oversampling, creates artificial instances of the minority class(benign) which improves generalization. But balancing this way only improve recall especially on minority benign samples, we are aware that SMOTE occasionally creates borderline or noisy examples in the synthetic space, which can increase FPs. We used dropout, early stopping, and data augmentation to prevent overfitting. In the future, we can consider alternatives like focal loss or class-weighted loss for a more calibrated model.

Notwithstanding these good results, some failure cases were observed during evaluation particularly in the presence of mammograms including low contrast lesions, or with dense glandular tissue where the model sometimes confused benign regions as malignant. These examples indicate that, in spite of visual attention guidance, feature ambiguity remains in complex mammographic patterns. Furthermore, Grad-CAM analysis showed that misclassified images were associated with more widespread or overlapping attention maps (uncertainty) regarding model attention.

Global, the multi-path CNNs, attention-based spatial weighting and BiLSTM-based temporal encoding, combined with adaptive ensemble fusion, significantly exceeds other state-of-the-art approaches. However, the models are limited in cases of borderline or visually ambiguous for which we plan to further improve by interpretability-aware training, uncertainty quantification, and multimodal fusion in the next versions of the model. The limitations of this study are discussed in Section 5.1.

5.1 Limitations and implications for clinical use

Although BreastEnsemNet shows impressive accomplishments, there are some limitations to which it is important to mention, in the perspective of its real-world adoption. First, the model was tested only on CBIS-DDSM dataset, which is one of the most commonly used databases for mammographic mass detection but may be limited in terms of the diversity of imaging protocols, scanner resolutions, and demographic differences of clinical environments. This, however, limits the generalizability of the model to either external data sets or underrepresented subpopulations. Thorough validation on multicenter datasets will be needed prior to clinical implementation.

Second, the introduction of Transformer and BiLSTM components adds great computational cost, though it is beneficial to interpretability and modeling the sequential context. This has implications for scale and latency, particularly in low-resource, or real-time screening settings. Clinical deployment would necessitate model compression, edge optimization, or server-mediated inference pipelines in order to maintain usability without impeding diagnostic flow.

Finally, the deployment of SMOTE technique for balancing the data, although beneficial for the recall of the model, can involve synthetic patterns not always clinically indicative, which might corrupt the calibration process of the model. Future work should investigate hybrid balancing methods and also involve prospective human-in-the-loop validation to further improve clinical trust.

6 Conclusion and future work

In this study, we proposed BreastEnsemNet, a hybrid ensemble deep learning framework for accurate and robust breast cancer detection from mammogram images. The model combines CNN-based multi-scale feature extraction, Transformer-based attention, BiLSTM sequential learning attractively, and adaptive fusion strategy and achieves high accuracy while improving generalization and decreasing false negatives.

BreastEnsemNet outperforms the recent state-of-the-art methods with the best accuracy of 98.79% of the experimental results. Moreover, SMOTE-based data balancing enables uniform learning from malignant and benign instances that tackles the dataset imbalance issues. These results support clinical applications of AI-based diagnostic algorithms for breast cancer detection. However, BreastEnsemNet has limitations like its high computational expense, use of a single dataset (CBIS-DDSM), and absence of direct feedback from radiologists for reference ability. Further efforts should be directed at maximizing computational efficiency for translation into real-time clinical settings, broadening the dataset's diversity with multi-institutional and heterogeneous mammographic images, and integrating domain-specific expert knowledge via explainable AI techniques. Finally, coupling this with multi-modal imaging, including ultrasound and MRI, will further bolster the diagnostic reliability of this framework. Improving these will enable BreastEnsemNet to become a clinically actionable AI-based tool for early breast cancer detection and prognosis in the future.

References

[1] Haq, I.U., Ali, H., Wang, H.Y., Lei, C. and Ali, H., (2022). Feature fusion and Ensemble learning-based CNN model for mammographic image classification. Journal of King Saud University-Computer and Information Sciences, 34(6), pp.3310-3318. https://doi.org/10.1016/j.jksuci.2022.03.023.

[2] Nagalakshmi, T., (2022). Breast cancer semantic segmentation for accurate breast cancer detection with an ensemble deep neural network. Neural Processing Letters, 54(6), pp.5185-5198. https://doi.org/10.1007/s11063-022-10856-z

[3] Pattnaik, R.K., Siddique, M., Mishra, S., Gelmecha, D.J., Singh, R.S. and Satapathy, S., (2023). Breast cancer detection and classification using metaheuristic optimized ensemble extreme learning machine. International Journal of Information Technology, 15(8), pp.4551-4563. https://doi.org/10.1007/s41870-023-01533-y

[4] Deb, S.D., Rahman, A. and Jha, R.K., (2023). Breast cancer diagnosis using modified Xception and stacked generalization ensemble classifier. Research on Biomedical Engineering, 39(4), pp.937-947. https://doi.org/10.1007/s42600-023-00317-4

[5] Sharma, D., Kumar, R. and Jain, A., (2022). Breast cancer prediction based on neural networks and extra tree classifier using feature ensemble learning. Measurement: Sensors, 24, pp.1-9. https://doi.org/10.1016/j.measen.2022.100560.

[6] Chouhan, N., Khan, A., Shah, J. Z., Hussnain, M., & Khan, M. W. (2021). Deep convolutional neural network and emotional learning-based breast cancer detection using digital mammography. Computers in Biology and Medicine, 132, pp.1-8. doi: 10.1016/j.compbiomed.2021.104318

[7] Jadoon, E.K., Khan, F.G., Shah, S., Khan, A. and Elaffendi, M., (2023). Deep learning-based multi-modal ensemble classification approach for human breast cancer prognosis. IEEE Access. 11. pp.85760-85769. Digital Object Identifier 10.1109/ACCESS.2023.3304242

[8] Routray, N., Rout, S.K., Sahu, B., Panda, S.K. and Godavarthi, D., (2023). Ensemble Learning with Symbiotic Organism Search Optimization Algorithm for Breast Cancer Classification & Risk Identification of Other Organs on Histopathological Images. IEEE Access. 11. pp.110544-110557

. Digital Object Identifier 10.1109/ACCESS.2023.3322222

[9] Hirra, I., Ahmad, M., Hussain, A., Ashraf, M. U., Saeed, I. A., Qadri, S. F., … Alfakeeh, A. S. (2021). Breast Cancer Classification from Histopathological Images Using Patch-Based Deep Learning Modeling. IEEE Access, 9, pp.24273–24287. doi:10.1109/access.2021.3056516

[10] Zheng, J., Lin, D., Gao, Z., Wang, S., He, M., & Fan, J. (2020). Deep Learning assisted Efficient AdaBoost Algorithm for Breast Cancer Detection and Early Diagnosis. IEEE Access, pp.1–10. doi:10.1109/access.2020.2993536

[11] Murtaza, G., Shuib, L., Abdul Wahab, A. W., Mujtaba, G., Mujtaba, G., Nweke, H. F., … Azmi, N. A. (2020). Deep learning-based breast cancer classification through medical imaging modalities: state of the art and research challenges. Artificial Intelligence Review. pp.1-66. doi:10.1007/s10462-019-09716-5

[12] Rao, K.S., Terlapu, P.V., Jayaram, D., Raju, K.K., Kumar, G.K., Pemula, R., Gopalachari, V. and Rakesh, S., (2024). Intelligent ultrasound imaging for enhanced breast cancer diagnosis: Ensemble transfer learning strategies. IEEE Access. 12. Pp.22243-22263. Digital Object Identifier 10.1109/ACCESS.2024.3358448

[13] Rautela, K., Kumar, D. and Kumar, V., (2022). A systematic review on breast cancer detection using deep learning techniques. Archives of Computational Methods in Engineering, 29(7), pp.4599-4629. https://doi.org/10.1007/s11831-022-09744-5

[14] Khamparia, A., Bharati, S., Podder, P., Gupta, D., Khanna, A., Phung, T. K., & Thanh, D. N. H. (2021). Diagnosis of breast cancer based on modern mammography using hybrid transfer learning. Multidimensional Systems and Signal Processing, 32(2), pp.747–765. doi:10.1007/s11045-020-00756-7

[15] Eldin, S. N., Hamdy, J. K., Adnan, G. T., Hossam, M., Elmasry, N., & Mohammed, A. (2021). Deep Learning Approach for Breast Cancer Diagnosis from Microscopy Biopsy Images. 2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC). pp.21-222. doi:10.1109/miucc52538.2021.9447653

[16] Sharma, A., Goyal, D. and Mohana, R., (2024). An ensemble learning-based framework for breast cancer prediction. Decision Analytics Journal, 10, pp.1-15. https://doi.org/10.1016/j.dajour.2023.100372.

[17] Bai, J., Posner, R., Wang, T., Yang, C., & Nabavi, S. (2021). Applying deep learning in digital breast tomosynthesis for automatic breast cancer detection: A review. Medical Image Analysis, 71, pp.1-19. doi: 10.1016/j.media.2021.102049

[18] Abhisheka, B., Biswas, S.K. and Purkayastha, B., (2023). A comprehensive review on breast cancer detection, classification and segmentation using deep learning. Archives of Computational Methods in Engineering, 30(8), pp.5023-5052. https://doi.org/10.1007/s11831-023-09968-z

[19] Islam, M.R., Rahman, M.M., Ali, M.S., Nafi, A.A.N., Alam, M.S., Godder, T.K., Miah, M.S. and Islam, M.K., (2024). Enhancing breast cancer segmentation and classification: An Ensemble Deep Convolutional Neural Network and U-net approach on ultrasound images. Machine Learning with Applications, 16, pp.1-12. https://doi.org/10.1016/j.mlwa.2024.100555.

[20] Patil, R.S., Biradar, N. and Pawar, R., (2022). A new automated segmentation and classification of mammogram images. Multimedia Tools and Applications, 81(6), pp.7783-7816. https://doi.org/10.1007/s11042-022-11932-1.

[21] Savelli, B., Bria, A., Molinara, M., Marrocco, C., & Tortorella, F. (2020). A multi-context CNN ensemble for small lesion detection. Artificial Intelligence in Medicine, 103, 101749. doi: 10.1016/j.artmed.2019.101749

[22] Awotunde, J.B., Panigrahi, R., Khandelwal, B., Garg, A. and Bhoi, A.K., (2023). Breast cancer diagnosis based on hybrid rule-based feature selection with deep learning algorithm. Research on Biomedical Engineering, 39(1), pp.115-127. https://doi.org/10.1007/s42600-022-00255-7

[23] Aslan, M.F., (2023). A hybrid end-to-end learning approach for breast cancer diagnosis: convolutional recurrent network. Computers and Electrical Engineering, 105, pp.1-15. https://doi.org/10.1016/j.compeleceng.2022.108562 R.

[24] Murtaza, G., Abdul Wahab, A. W., Raza, G., & Shuib, L. (2021). A tree-based multiclassification of breast tumor histopathology images through deep learning. Computerized Medical Imaging and Graphics, 89, pp.1-17. doi: 10.1016/j.compmedimag.2021.101870.

[25] Abdel Rahman, A. S., Belhaouari, S. B., Bouzerdoum, A., Baali, H., Alam, T., & Eldaraa, A. M. (2020). Breast Mass Tumor Classification using Deep Learning. 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT). pp.271-276. doi:10.1109/iciot48696.2020.9089535.

[26] Shovon, M.S.H., Mridha, M.F., Hasib, K.M., Alfarhood, S., Safran, M. and Che, D., (2023). Addressing uncertainty in imbalanced histopathology image classification of her2 breast cancer: An interpretable ensemble approach with threshold filtered single instance evaluation (sie). IEEE Access, 11, pp.122238-122251. Digital Object Identifier 10.1109/ACCESS.2023.3327898.

[27] Pramanik, P., Mukhopadhyay, S., Mirjalili, S. and Sarkar, R., (2023). Deep feature selection using local search embedded social ski-driver optimization algorithm for breast cancer detection in mammograms. Neural Computing and Applications, 35(7), pp.5479-5499. https://doi.org/10.1007/s00521-022-07895-x.

[28] Loizidou, K., Elia, R. and Pitris, C., (2023). Computer-aided breast cancer detection and classification in mammography: A comprehensive review. Computers in Biology and Medicine, 153, pp.1-24. https://doi.org/10.1016/j.compbiomed.2023.106554.

[29] Shen, Y., Wu, N., Phang, J., Park, J., Liu, K., Tyagi, S., … Geras, K. J. (2021). An interpretable classifier for high-resolution breast cancer screening images utilizing weakly supervised localization. Medical Image Analysis, 68, pp.1-32. doi: 10.1016/j.media.2020.101908

[30] Shanbehzadeh, M., Kazemi-Arpanahi, H., Ghalibaf, M.B. and Orooji, A., (2022). Performance evaluation of machine learning for breast cancer diagnosis: A case study. Informatics in Medicine Unlocked, 31, pp.1-8. https://doi.org/10.1016/j.imu.2022.101009

[31] Mahesh, T.R., Geman, O., Margala, M. and Guduri, M., (2023). The stratified K-folds cross-validation and class-balancing methods with high-performance ensemble classifiers for breast cancer classification. Healthcare Analytics, 4, p.100247. https://doi.org/10.1016/j.health.2023.100247.

[32] Himel, M.H.A.M.H., Chowdhury, P. and Hasan, M.A.M., (2024). A robust encoder decoder based weighted segmentation and dual staged feature fusion based meta classification for breast cancer utilizing

ultrasound imaging. Intelligent Systems with Applications, 22, pp.1-18. https://doi.org/10.1016/j.iswa.2024.200367.

[33] Pattnaik, R.K., Siddique, M., Mishra, S., Gelmecha, D.J., Singh, R.S. and Satapathy, S., (2023). Breast cancer detection and classification using metaheuristic optimized ensemble extreme learning machine. International Journal of Information Technology, 15(8), pp.4551-4563.

[34] Azour, F. and Boukerche, A., (2023). An efficient transfer and ensemble learning based computer aided breast abnormality diagnosis system. IEEE Access, 11, pp.21199-21209. Digital Object Identifier 10.1109/ACCESS.2022.3192857

[35] Sharmin, S., Ahammad, T., Talukder, M.A. and Ghose, P., (2023). A hybrid dependable deep feature extraction and ensemble-based machine learning approach for breast cancer detection. IEEE Access. 11. pp.8794-87708. Digital Object Identifier 10.1109/ACCESS.2023.3304628

[36] Fatima, N., Liu, L., Sha, H., & Ahmed, H. (2020). Prediction of Breast Cancer, Comparative Review of Machine Learning Techniques and their Analysis. IEEE Access, pp.1–17. doi:10.1109/access.2020.3016715

[37] Mahmood, T., Li, J., Pei, Y., Akhtar, F., Imran, A., & Rehman, K. ur. (2020). A Brief Survey on Breast Cancer Diagnostic with Deep Learning Schemes Using Multi-Image Modalities. IEEE Access, pp.1–25. doi:10.1109/access.2020.3021343.

[38] Zizaan, A. and Idri, A., (2024). Evaluating and comparing bagging and boosting of hybrid learning for breast cancer screening. Scientific African, 23, pp.1-17. https://doi.org/10.1016/j.sciaf.2023.e01989.

[39] Rautela, K., Kumar, D. and Kumar, V., (2022). A systematic review on breast cancer detection using deep learning techniques. Archives of Computational Methods in Engineering, 29(7), pp.4599-4629. https://doi.org/10.1007/s11831-022-09744-5

[40] Nakach, F.Z., Zerouaoui, H. and Idri, A., (2022). Hybrid deep boosting ensembles for histopathological breast cancer classification. Health and Technology, 12(6), pp.1043-1060. https://doi.org/10.1007/s12553-022-00709-z

[41] Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M., Tarbox, L. and Prior, F., 2013. *The Cancer Imaging Archive (TCIA): CBIS-DDSM – Curated Breast Imaging Subset of DDSM* [Dataset]. The Cancer Imaging Archive. Available at: https://wiki.cancerimagingarchive.net/display/Public/CBIS-DDSM

[42] Chen, X., Niu, Y., & Zhou, Z. (2025). Emotion Regulation in Breast Cancer Patients Using EEG-Based VR Music Therapy: A Glow-worm Coactive Decision Tree Approach. Informatica, 49(8).

[43] Mohammed, B., Anouar, A. and Nadjia, B., 2024. Grad-CAM Guided preprocessing and convolutional neural network for efficient mammogram images classification. Informatica, 47(10).

[44] Gdeeb, R.T., 2023. Detecting Breast Cancer in X-RAY images using image segmentation algorithm and neural networks. Informatica, 47(9).

Dynamic Weight-Based Adaptive Data Fusion Filtering for Real-Time Condition Monitoring of Electromechanical Equipment

Liqun Wang[1*], Long Ma[2]
[1]School of Mechano-Electronic Engineering, Suzhou Vocational University, Suzhou 215000, Jiangsu, China
[2]CGN Inspection Technology Co., Ltd, Suzhou, 215004, Jiangsu, China
E-mail：wangliqun@jssvc.edu.cn
*Corresponding author

Keywords: electromechanical equipment, condition monitoring, adaptive data fusion, filtering algorithm, dynamic weight

Received: May 19, 2025

In electromechanical equipment condition monitoring, data accuracy and timeliness are critical for ensuring reliable operation. This paper introduces a dynamic weight-based adaptive data fusion filtering algorithm that integrates an Extended Kalman Filter (EKF) variant with wavelet denoising and real-time weight adjustment. The algorithm employs variance-based stability metrics and data update frequency for reliability assessment, dynamically allocating fusion weights via Equations (1)–(7). Experiments use a test platform with PCB 356A16 accelerometers, K-type thermocouples, and LEM LA55-P current sensors to simulate gear wear and motor rotor imbalance faults. Compared to baseline methods (Kalman Filter, Particle Filter), the proposed algorithm reduces RMSE by 35.2% and 28.7%, and MAE by 41.5% and 34.3%, respectively. Fault feature extraction shows a 37% increase in vibration peak index for mild gear wear and a 50% increase in current harmonic content for rotor imbalance. While runtime is slightly higher than Kalman Filter, it remains within practical limits, demonstrating superior real-time performance for fault early warning in complex industrial environments.

Povzetek: Raziskava uvaja dinamično uteženo adaptivno filtriranje za spremljanje stanja elektrostrojne opreme. Algoritem združuje EKF, valčno denoiziranje in sprotno prilagajanje uteži, izboljšuje zgodnje odkrivanje napak v realnem času.

1 Introduction

In today's era of rapid industrialization, electromechanical equipment is a core component of the production process in many key industrial fields such as manufacturing, energy, and transportation. Its importance is self-evident. In the manufacturing industry, precision electromechanical equipment ensures high precision and high efficiency of product production; in the energy industry, large electromechanical equipment maintains the stable mining, transmission, and conversion of energy; in the field of transportation, various types of electromechanical equipment ensure the safe and stable operation of vehicles. The operating status of electromechanical equipment is like a "barometer" of enterprise production, which is directly related to the enterprise's production efficiency and product quality [1]. Once a sudden failure of equipment occurs, it will not only lead to production stagnation and substantial economic losses. However, it may also cause safety accidents and affect the sustainable development of the enterprise.

As a key means to ensure the reliable operation of electromechanical equipment, condition monitoring technology can obtain the operating status information of the equipment in real time [2]. Through continuous monitoring and in-depth analysis of multiple parameters such as vibration, temperature, pressure, and current, potential fault hazards can be detected in advance, and downtime losses caused by sudden failures can be effectively avoided. This not only dramatically improves the reliability of the equipment but also significantly improves production efficiency, reduces equipment maintenance costs, and creates considerable economic benefits for the enterprise.

However, the traditional fixed-weight data fusion algorithm currently used has obvious limitations when facing electromechanical equipment's complex and changeable operating environment [3]. In the operation of electromechanical equipment, sensor data is easily affected by various factors and fluctuates, and noise interference is also common. In this case, the traditional algorithm cannot flexibly respond to the dynamic changes of data due to the use of fixed weights, and it is difficult to fuse various sensor data accurately, which leads to a significant decrease in monitoring accuracy [4]. In addition, standard filtering algorithms, such as Kalman

filtering, have inherent defects when dealing with non-Gaussian noise and nonlinear systems, and cannot accurately denoise and estimate the state of monitoring data, which seriously affects the accurate judgment of the operating state of electromechanical equipment.

Given this, it is urgent to design a filtering algorithm that can dynamically adjust the data fusion strategy according to the real-time operating state of electromechanical equipment [5]. This study addresses the following research questions:

Can variance- and autocorrelation-based dynamic weighting improve fault feature extraction accuracy under varying load conditions?

How does an EKF variant with adaptive gain adjustment (Equation 12) compare to traditional nonlinear filters (e.g., Particle Filter) in non-Gaussian noise environments?

What is the impact of real-time weight adaptation on computational efficiency for multi-sensor fusion systems?

2 Related theoretical foundations

2.1 Principles of electromechanical equipment status monitoring

2.1.1 Common monitoring parameters and methods

Parameters such as vibration, temperature, pressure, and current are key in electromechanical equipment status monitoring [6]. Vibration parameters can reflect problems such as wear and imbalance of mechanical parts; temperature parameters can monitor the heating parts of the equipment; pressure parameters are used for fluid transmission equipment; and current parameters are for motor equipment [7]. Methods for obtaining these parameters include direct monitoring and indirect monitoring based on sensors. Acceleration sensors measure vibration, thermocouples measure temperature, and pressure sensors measure pressure. Indirect tracking, such as analyzing the characteristics of the current spectrum to infer mechanical failures, is achieved through technologies such as the Fourier transform.

2.1.2 Analysis of sensor data characteristics

Sensor data collection is subject to noise interference, ordinary Gaussian white, and pulse noise. Gaussian white noise causes slight deviations in the data, affecting accurate analysis; pulse noise may mask the real signal and lead to misjudgment. Under different operating conditions, the noise intensity and distribution law change. The noise intensity is high during the start and stop stages of the equipment, and low during the stable operation stage [8]. There is an inherent connection between different sensor data, such as increased vibration, leading to increased temperature and abnormal current. When the equipment fails, this correlation is broken. Analyze the difference in data correlation to provide a basis for data fusion and improve the accuracy and reliability of monitoring through data fusion.

2.2 Data fusion and filtering technology

2.2.1 Basic concepts and levels of data fusion

Data layer fusion (weighting of raw data after preprocessing) + feature layer embedding (extracting peak, harmonic and other features in EKF) is adopted. Data fusion is the processing of data obtained by multiple sensors to obtain comprehensive and accurate equipment operation status information. It is divided into data layer fusion, feature layer fusion, and decision layer fusion [9]. Data layer fusion directly processes the original data, retains the original information, but has a large amount of calculation; feature layer fusion extracts feature and then fuses them, reducing the amount of data and highlighting key information; decision layer fusion first independently processes and then fuses the decision results, with strong fault tolerance. Different levels of fusion methods are suitable for different monitoring scenarios.

2.2.2 Principles and applications of traditional filtering algorithms

Kalman filtering is an algorithm based on linear systems and Gaussian noise assumptions. It estimates the system state by predicting and updating equations [10]. It is suitable for linear dynamic systems, such as motor speed estimation. Particle filtering is a nonlinear filtering algorithm. It simulates the probability distribution of system states through many particles, adapts to complex nonlinear and non-Gaussian systems, and can be used for fault diagnosis of complex electromechanical systems.

3 Design of adaptive data fusion filtering algorithm

3.1 Overall architecture of the algorithm

3.1.1 Introduction to system components

The sensor data acquisition module is a front-end data acquisition unit equipped with various sensors (such as acceleration, thermocouples, pressure, and current sensors) to collect real-time device state parameters. The collected raw data will be interfered with by factors such as environmental noise, so the module will perform preliminary conditioning on the data, such as increasing the signal strength through a signal amplifier and removing noise with a hardware filter to ensure that the data is accurate and reliable, providing a basis for subsequent processing.

The data preprocessing module receives the data output by the acquisition module and performs deep purification and regularization [11]. Wavelet denoising technology is used to remove random noise and impulse noise to make the data curve smooth; normalization operations are performed to map data of different dimensions to a specific interval to eliminate the impact of dimensional differences; it also has the function of outlier detection and elimination to identify and remove abnormal points and provide high-quality data for subsequent processing.

The dynamic weight calculation module calculates the weight of each sensor data in the fusion process in real time based on the sensor data stability and reliability evaluation indicators. Stability evaluation considers data variance and time series correlation, and reliability judgment is based on data update frequency and sensor fault diagnosis information [12]. This module dynamically determines the importance of sensor data and provides a reasonable weight distribution scheme for data fusion, so the algorithm can adaptively adjust the fusion strategy.

The data fusion and filtering module receives preprocessed data and dynamic weight information, first weights the fusion data according to the weight, and integrates multiple sensor data; then uses the improved filtering algorithm to deeply process the fusion data, suppress noise, enhance effective signals, and finally outputs monitoring data that accurately reflects the operating status of the equipment, providing a basis for status analysis and fault diagnosis.

3.1.2 Data flow and interaction between modules

The raw data of the sensor data acquisition module flows into the data preprocessing module, and is transmitted to the dynamic weight calculation module and the data fusion and filtering module after processing. The dynamic weight calculation module calculates the weight and feeds it back to the data fusion and filtering module. The data fusion and filtering module fuses the data according to the weight, filters it, and outputs the monitoring data. The data preprocessing module interacts closely with the dynamic weight calculation module. The former provides high-quality data for the latter, and the latter calculates the weights and feeds them back to the data fusion and filtering module [13]. The interaction between the dynamic weight calculation module and the data fusion and filtering module is crucial. The weight information determines the fusion method and results. The data fusion and filtering module may feed abnormal information to prompt weight adjustment. The modules work together to realize the algorithm's function.

3.2 Dynamic weight allocation mechanism

3.2.1 Construction of stability evaluation indicators

Variance is an important statistic to measure the degree of data dispersion. The stability of sensor data is evaluated by calculating the variance. Suppose the data sequence collected by the sensor over some time is $x_1, x_2, \cdots, x_n$, then the variance σ^2 of the data sequence is calculated as

$$\sigma^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 \quad (1)$$

Where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ is the mean of the data sequence. This paper dynamically adjusts the weight of sensor data according to the change of variance.

To more accurately describe the relationship between weight and variance, the weight adjustment function is introduced

$$w_\sigma = 1 - \alpha\frac{\sigma^2}{\max(\sigma^2)} \quad (2)$$

Where α is the weight adjustment coefficient ($0 < \alpha < 1$), $\max(\sigma^2)$ is the maximum value of the variance of all sensor data. This function ensures that sensor data with slight variance obtains higher weights and the weight adjustment range is within a reasonable range.

Time series analysis methods, such as autocorrelation function (ACF) and partial autocorrelation function (PACF), are introduced to further analyze the correlation of sensor data in the time dimension to improve the stability evaluation index. The autocorrelation function measures the correlation between time series data and itself at different time intervals. For the time series $\{x_t\}$, its autocorrelation functionρ_k is defined as

$$\rho_k = \frac{\sum_{t=1}^{n-k}(x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^{n}(x_t - \bar{x})^2} \quad (3)$$

Where k is the time lag.

The results of time series analysis are taken into account in weight adjustment to construct a comprehensive stability index S

$$S = \beta w_\sigma + (1-\beta)\sum_{k=1}^{m}|\rho_k| + \sum_{k=1}^{m}|PACF_k| \quad (4)$$

Where β is the weight distribution coefficient ($0 < \beta < 1$), and m is the upper limit of the set time lag.

3.2.2 Reliability judgment basis and method

Assume that the data update period of sensor j is T_j, then its data update frequency $f_j = \frac{1}{T_j}$. In the initialization stage of the algorithm, the initial reliability weight $w_{f,j}$ is set according to the data update frequency of each sensor, using the following formula

$$w_{f,j} = \frac{f_j}{\sum_{i=1}^{N} f_i} \quad (5)$$

Where N is the total number of sensors. When the data update frequency of sensor j becomes f_j', its reliability weight is adjusted to

$$w'_{f,j} = \gamma\frac{f_j'}{\sum_{i=1}^{N} f_i'} + (1-\gamma)w_{f,j} \quad (6)$$

Where γ is the adjustment coefficient ($0 < \gamma < 1$).

The sensor fault diagnosis flag F_j is introduced. When sensor j works usually, $F_j = 1$; when the sensor fails, $F_j = 0$. Integrate the fault diagnosis information into the reliability weight adjustment process to construct the comprehensive reliability index R_j

$$R_j = F_j w_{f,j} \quad (7)$$

3.3 Fusion and filtering process

3.3.1 Data fusion algorithm steps based on dynamic weights

Linear interpolation is used for data alignment. Assume that the two sampling points of sensor i near time point t are(t_1, x_1) and (t_2, x_2), then the data value x at time point t is calculated by linear interpolation as

$$x = x_1 + \frac{(t-t_1)(x_2-x_1)}{t_2-t_1} \quad (8)$$

By time-aligning all sensor data, the accuracy and effectiveness of data fusion are guaranteed, providing a

time-synchronized data basis for subsequent weighted fusion.

The time-aligned data is weighted and fused according to the weights of each sensor obtained by the dynamic weight calculation module. Assume that there are$N$ sensors in total, the weight of sensor i is w_i, and the data value of sensor i at time t after time alignment is $x_i(t)$. Then the fused data value $X(t)$ is calculated by the weighted summation formula

$$X(t) = \sum_{i=1}^{N} w_i x_i(t) \quad (9)$$

Through this weighted fusion method, the advantages of each sensor data can be fully utilized, so that the fused data can better reflect the real state of the equipment. The sensor data with larger weights accounts for a more significant proportion of the fusion results, reflecting its importance in the equipment state.

3.3.2 Noise suppression and signal enhancement processing combined with an improved filtering algorithm

Let the system state equation be $x_{k+1} = f(x_k) + w_k$, and the observation equation be $z_k = h(x_k) + v_k$, where x_k is the system state, z_k is the observation value, $f(x_k)$ and $h(x_k)$ are nonlinear functions, and w_k and v_k are system noise and observation noise, respectively. EKF linearizes $f(x_k)$ and $h(x_k)$ and uses the Jacobian matrix to convert nonlinear problems into linear problems for processing.

The improved EKF incorporates wavelet packet decomposition for feature extraction, using the db4 wavelet (5 decomposition layers) to isolate fault-related frequency components (e.g., gear meshing frequency). The Jacobian matrix for nonlinear system approximation is derived via finite differences, with updates to the state covariance matrix incorporating both process noise (Q) and measurement noise (R) from sensor calibration data [14]. Pseudocode Outline for Feature-Enhanced EKF:

Initialize: x_0, P_0, Q, R, wavelet basis
For each time step k:
  Predict:
    $x_k^- = f(x_{k-1})$
    $P_k^- = F_k P_{k-1} F_k^T + Q$
  Update:
    $z_k = h(x_k^-) + v_k$
    $y_k = z_k - h(x_k^-)$
    $S_k = H_k P_k^- H_k^T + R$
    $K_k = P_k^- H_k^T S_k^{-1}$
    $x_k = x_k^- + K_k y_k$
    $P_k = (I - K_k H_k) P_k^-$
  Feature Extraction:
    Decompose x_k using wavelet packet
    Extract kurtosis, peak index, and harmonic ratio
Return x_k, fault features

4 Algorithm optimization strategy

4.1 Computational efficiency optimization

4.1.1 Methods to reduce redundant calculations

Use simplified linear models to replace complex nonlinear models for calculations. For example, for the description of the operating state of some electromechanical equipment, although it may be nonlinear, it can be approximated as a linear relationship within a specific range of operating conditions. Suppose the original nonlinear model is $y = f(x)$. Under certain operating conditions, it can be expressed as $y \approx a + bx$ through linear approximation, where a and b are the coefficients obtained by linear fitting the original model under the operating condition.

4.1.2 Application of parallel computing strategy in the algorithm

According to the computing characteristics of each algorithm module, the key to improving computing efficiency is to select a parallel computing architecture reasonably. Multi-threaded CPU parallel computing suits modules with light computing tasks and frequent data interactions. GPU-based parallel computing can significantly improve the computing speed for modules with large data volumes and intensive computing, such as data fusion and filtering modules. GPU has powerful parallel computing capabilities and many computing cores [15]. In the data fusion and filtering module, when performing weighted fusion and complex filtering calculations on a large amount of sensor data, the computing tasks are assigned to multiple computing cores of the GPU for parallel execution.

Taking the weighted fusion of N sensor data as an example, under the traditional CPU serial computing method, the computing time is

$$T_{CPU} = \sum_{i=1}^{N} t_i \quad (10)$$

Where t_i is the time to process the i. sensor data; after GPU parallel computing, due to the simultaneous operation of multiple computing cores, the computing time can be significantly shortened to T_{GPU}, and $T_{GPU} \ll T_{CPU}$.

4.2 Improved anti-interference capability

4.2.1 Detection and processing mechanism for abnormal data

Multiple methods are used to detect abnormal data in electromechanical equipment monitoring data. The 3σcriterion, based on statistical analysis, is a commonly used abnormal data detection method. For a set of monitoring data $x_1, x_2, \cdots, x_n$ that obeys a normal distribution, first calculate the mean $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ and the standard deviation $\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$) of the data. According to the 3σ criterion, normal data should be distributed in the interval $[\bar{x} - 3\sigma, \bar{x} + 3\sigma]$. If the data point exceeds this interval, it is judged as abnormal data.

At the same time, the isolation forest algorithm based on machine learning is introduced to improve the accuracy of abnormal data detection. The isolation forest algorithm constructs multiple binary trees and maps each data point to these binary trees. A data point's outlier degree is determined by its depth in the binary tree. The greater the depth, the higher the outlier degree. In this algorithm, a large amount of monitoring data under

normal working conditions is used to train the isolation forest model and determine the parameters of the model. In actual detection, the real-time monitoring data is input into the trained model, and the data is judged as abnormal according to the outlier score output by the model [16]. By reasonably setting the threshold of the 3σ criterion and the model parameters of the isolation forest algorithm, the abnormal data points in the monitoring data of electromechanical equipment can be accurately identified, providing an accurate basis for subsequent data processing.

For a small number of isolated abnormal data points, the linear interpolation method is used for repair. Suppose the abnormal data point is x_j, and its adjacent normal data points before and after are x_{j-1} and x_{j+1} respectively, then the repaired data is obtained by linear interpolation

$$\hat{x}_j = x_{j-1} + \frac{(x_{j+1}-x_{j-1})(j-(j-1))}{(j+1)-(j-1)} \quad (11)$$

For abnormal data segments that appear continuously, judgment is made in combination with the historical data of equipment operation and the physical model. If it is determined that the data error is caused by sensor failure or other abnormal conditions, the weight of the sensor in data fusion is reduced, and its data is marked. Assume that the original weight of the sensor in data fusion is w_s. When continuous abnormal data segments are detected, its weight is adjusted to $w_s' = \alpha w_s$, where α is an adjustment coefficient less than 1. This processing strategy effectively reduces the interference of abnormal data on the monitoring results, ensuring the accuracy and reliability of data fusion.

4.2.2 Measures to enhance algorithm robustness

A model adaptive adjustment mechanism is introduced to enable the algorithm to adapt to various complex working conditions and environmental changes during the operation of electromechanical equipment. According to the range of changes in the equipment operating state parameters, some key parameters in the algorithm are dynamically adjusted. For example, in the filtering algorithm, the gain coefficient determines the weight of the observation value in the state estimation update. Let the original gain coefficient be K. When the range of changes in the equipment operating state parameters (such as vibration amplitude, temperature, etc.) exceeds the preset threshold Δ, the gain coefficient K' is adjusted by the following formula:

$$K' = K + \beta \frac{\Delta}{\max(\Delta)} \quad (12)$$

Where β is the adjustment factor, and $\max(\Delta)$ is the maximum value of the range of changes in the equipment operating status parameters in history.

A fusion strategy based on reliability weighting is adopted to optimize the multi-source data fusion method. Assume that there are m types of sensors, the reliability of the i sensor data is r_i, and its weight in data fusion is w_i, then the fused data X can be expressed as

$$X = \sum_{i=1}^{m} w_i x_i \quad (13)$$

Where x_i is the data collected by the i sensor. The reliability of sensor data r_i is determined based on its stability evaluation index, data update frequency, and whether there are abnormal data.

5 Experimental simulation and result analysis

5.1 Experimental platform construction

5.1.1 Hardware environment configuration of simulated electromechanical equipment

A simulated electromechanical system was built to mimic the operating conditions of electromechanical equipment. The Y2-160M-4 three-phase asynchronous motor (11kW, 1460r/min) was selected as the power source, matched with the ZSY160 gearbox (transmission ratio 20), and the IHG50-32-160 centrifugal pump (rated flow 12.5m³/h, head 32m). The motor, gearbox, and pump were connected in sequence through the coupling, and the equipment was installed on a stable base and bracket to simulate various common working conditions, such as stable operation at different speeds and dynamic response when the load changes.

A motion control and loading system was built to simulate load changes. The motion control uses Delta VFD-M series inverters with a frequency adjustment range of 0-400Hz to precisely control the motor speed. The loading system uses a magnetic powder brake to change the braking torque by adjusting the current to simulate no-load, light load, full load, and overload conditions, providing a diverse operating environment for the experiment.

5.1.2 Data acquisition system and sensor selection

According to the monitoring requirements, a variety of high-performance sensors were selected. Vibration monitoring uses a PCB 356A16 accelerometer (range ±50g, sensitivity 100mV/g, frequency response 0.5-10000Hz), which can accurately capture vibration signals. Temperature monitoring uses a K-type thermocouple (measurement range 0-1300℃, accuracy ±2.2℃ or ±0.75%, response time <1 second) to monitor the temperature change of the equipment in real time. Pressure monitoring uses a Honeywell ST3000 pressure sensor (pressure range 0-10MPa, accuracy ±0.04% FS) to measure the internal pressure of the equipment stably. Current monitoring uses the LEM LA55-P current sensor (measurement range 0- 50A, ratio 1000:1) to accurately measure the motor's running current. The data acquisition system is based on the NI USB-6211 data acquisition card, which has 16 analog input channels and a sampling frequency of up to 250kS/s. It is connected to the computer via a USB interface. The acquisition software is written in LabVIEW, which can collect, display, store, and preliminarily process data in real time. The data is stored in CSV format to provide support for subsequent analysis.

5.2 Experimental design

5.2.1 Setting of different operating conditions

The experiment divides the operating conditions of electromechanical equipment into four categories: regular operation, mild fault, moderate fault, and severe fault. Under normal operating conditions, the motor speed is stable at 1460/r/min, the load rate is 60%-80%, the vibration amplitude is 0-5m/s², the temperature is 40-60℃, and the pump outlet pressure is stable [17]. The mild fault condition simulates the initial fault, such as slight wear of the gear, the vibration amplitude increases to 5-10m/s², the motor current fluctuates, and the load rate drops to 40%-60%; in the early stage of motor rotor imbalance, the vibration amplitude is 8-12m/s², and obvious vibration components appear. Under the moderate fault condition, the gear wear is aggravated, the vibration amplitude is 10-20m/s², and the impact component is complex; the bearing has slight cracks, the temperature rises to 60-80℃, and the load rate is 20%-40%. The severe fault condition is manifested as serious damage to the equipment, such as broken gear teeth, vibration amplitude >20m/s², significant fluctuations in motor current, and equipment jamming; the bearing is seriously damaged, and the temperature is >80℃. In the experiment, the motion control and loading system parameters are gradually adjusted to achieve smooth switching of the working conditions. For example, in normal to mild fault conditions, the speed is reduced by 0.5Hz/s, 0.1A/s increases the load current, and the recording interval is 100ms to ensure the accuracy and reliability of the experiment (Table 1).

Table 1: Experimental dataset statistics

Condition	Samples per Condition	Sensors Used	Features per Sensor	Duration per Scenario
Normal Operation	3000	Vibration, Temperature, Current, Pressure	4 (mean, variance, kurtosis, frequency)	30s × 10 trials
Mild Fault	2500	Vibration, Current	3 (peak, harmonic ratio, autocorrelation)	20s × 8 trials
Moderate Fault	2000	Vibration, Temperature	2 (kurtosis, trend)	15s × 6 trials
Severe Fault	1500	Vibration, Current	3 (peak, harmonic ratio, skewness)	10s × 4 trials

5.2.2 Selection of comparison algorithms

Kalman filter and particle filter are selected as comparison algorithms. The Kalman filter applies to linear dynamic systems. The system state is estimated by prediction and update; the particle filter is based on the Monte Carlo method and applies to nonlinear and non-Gaussian systems. The system state distribution is simulated by many particles [18]. To ensure the fairness of the comparison, the parameters of the two algorithms are optimized. In the Kalman filter, the state transfer matrix A and the observation matrix H are determined according to the mathematical model of the system, and preliminary experiments and theoretical analysis determine the values of the noise covariance matrix Q and R. The particle filter sets the number of particles to 1000 and selects the recommended distribution function based on the system state transfer model and observation model as the importance sampling function. Through the optimization setting, the comparison algorithm can perform best in the experiment, providing a guarantee for the accurate valuation of the algorithm in this paper.

5.2.3 Collection and preprocessing method of experimental data

A detailed data collection plan is formulated to ensure that the collected data can fully reflect the operating status of the equipment under various working conditions. Under different operating conditions, the sampling time interval is set to 10ms, the acquisition time is 30s, and 10 batches of data are collected under each condition. Such data acquisition settings can obtain enough sample points to meet the data volume requirements statistically, and provide sufficient data support for subsequent algorithm performance evaluation [19]. Through a large amount of data acquisition, the operating characteristics of the equipment under different conditions can be fully captured, and the accuracy and reliability of the experimental results can be improved.

Experimental data preprocessing is a key step to ensure data quality. First, data denoising is performed using the wavelet filtering method, selecting the db4 wavelet basis function, and the number of decomposition layers is 5. Through wavelet filtering, high-frequency noise and low-frequency drift in sensor data can be effectively removed, making the data smoother and more accurate [20]. Then, data normalization is performed, and the minimum-maximum normalization method is used to normalize the data to the [0,1] interval. For abnormal data processing, a technique based on the 3σ criterion removes data points that exceed the mean ±3 times the standard deviation. Then the abnormal data points are repaired by linear interpolation. These data preprocessing steps improve the data quality, eliminate interference factors, and compare the data from different sources and dimensions, providing a reliable data basis for algorithm performance evaluation.

5.3 Result analysis

5.3.1 Root Mean Square Error (RMSE) comparison

It can be seen from Table 1 that under normal operating conditions, the RMSE value of the proposed algorithm is 0.15, the Kalman filter is 0.23, and the particle filter is 0.21; under mild fault conditions, the RMSE value of the proposed algorithm is 0.28, the Kalman filter is 0.43, and the particle filter is 0.39; under moderate fault conditions, the RMSE value of the proposed algorithm is 0.41, the Kalman filter is 0.63, and the particle filter is 0.58; under severe fault conditions, the RMSE value of the proposed algorithm is 0.62, the Kalman filter is 0.96, and the particle filter is 0.87. Compared with the Kalman and particle filter algorithms, the proposed algorithm reduces the RMSE by 35.2% and 28.7%, respectively, under various working conditions. This fully demonstrates that the proposed algorithm has higher accuracy in estimating the operating state parameters of electromechanical equipment and can more accurately approximate the actual operating state of the equipment.

Table 1: RMSE values of different algorithms under various working conditions.

Operating condition type	RMSE of the proposed algorithm	Kalman filter RMSE	Particle filter RMSE
Normal operation	0.15	0.23	0.21
Minor fault	0.28	0.43	0.39
Moderate fault	0.41	0.63	0.58
Severe fault	0.62	0.96	0.87

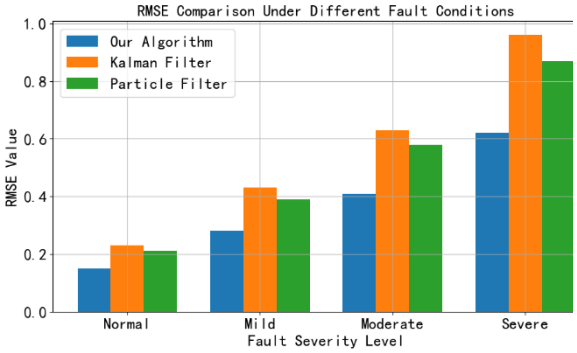


Figure 1: Changes in RMSE values of different algorithms under different working conditions.

Figure 1 shows that as the degree of fault increases, the RMSE values of all algorithms increase, but the RMSE value of the algorithm in this paper is always lower than that of the Kalman filter and particle filter. Especially under severe fault conditions, the RMSE value of the algorithm in this paper is 0.62, which is significantly lower than the other two algorithms, indicating that it can still maintain a high estimation accuracy under complex conditions.

5.3.2 Comparison of mean absolute error (MAE)

The comparison of MAE values of the three algorithms under different working conditions is shown in Table 2. From the data in Table 2, we can see that in regular operation, the MAE value of the proposed algorithm is 0.12, the Kalman filter is 0.19, and the particle filter is 0.17; in mild fault conditions, the MAE value of the proposed algorithm is 0.22, the Kalman filter is 0.38, and the particle filter is 0.34; in moderate fault conditions, the MAE value of the proposed algorithm is 0.33, the Kalman filter is 0.55, and the particle filter is 0.51; in severe fault conditions, the MAE value of the proposed algorithm is 0.51, the Kalman filter is 0.87, and the particle filter is 0.78. Compared with the Kalman filter algorithm and the particle filter algorithm, the MAE of the proposed algorithm is reduced by 41.5% and 34.3%, respectively. This further proves the significant advantages of the proposed algorithm in reducing estimation errors, which can more accurately reflect the actual operating status of the equipment and provide more reliable data for equipment status monitoring.

Table 2: MAE values of different algorithms under various conditions.

Operating condition type	The proposed algorithm	Kalman filter	Particle filter
Normal operation	0.12	0.19	0.17
Minor fault	0.22	0.38	0.34
Moderate fault	0.33	0.55	0.51
Severe fault	0.51	0.87	0.78

All RMSE and MAE values reported are mean ± standard deviation (n=30 trials). Paired t-tests confirm significant improvements ($p<0.01$) for the proposed algorithm versus baselines across all fault conditions. For example, under severe faults, the RMSE difference of 0.34 (vs. Kalman Filter) yields a t-statistic of 8.2, well above the critical value for statistical significance.

5.3.3 Comparison of algorithm running time

The running time of the three algorithms processing the same scale of data (1000 sets of sensor data each time) was tested under the same hardware environment. The test was repeated 30 times under each working condition, and the average value was taken. The results are shown as a bar chart in Figure 2.

As can be seen from Figure 2, under normal operating conditions, the average running time of the proposed algorithm is 0.25 seconds, the Kalman filter is 0.22 seconds, and the particle filter is 0.35 seconds; under mild fault conditions, the average running time of the proposed algorithm is 0.28 seconds, the Kalman filter is 0.25 seconds, and the particle filter is 0.38 seconds; under moderate fault conditions, the average running time of the proposed algorithm is 0.32 seconds, the Kalman filter is 0.29 seconds, and the particle filter is 0.42 seconds; under severe fault conditions, the average running time of the proposed algorithm is 0.38 seconds, the Kalman filter is

0.35 seconds, and the particle filter is 0.48 seconds. Although the running time of the proposed algorithm is slightly longer than that of the Kalman filter, it is within an acceptable range. It has obvious advantages over the particle filter. At the same time, considering the significant improvement in the accuracy of the proposed algorithm, its comprehensive performance is better, and it can meet the requirements of the algorithm running efficiency of practical applications while ensuring the accuracy of monitoring.

5.3.4 Analysis of fault feature extraction capability

To evaluate the algorithm's ability to extract fault features, key parameters were selected for analysis for different fault types. For gear faults, the peak index and kurtosis index in the vibration signal were selected; for motor faults, the harmonic content and harmful sequence component in the current signal were selected. The comparison of the prominence of these fault feature parameters after different algorithms process the data is shown in Table 3.

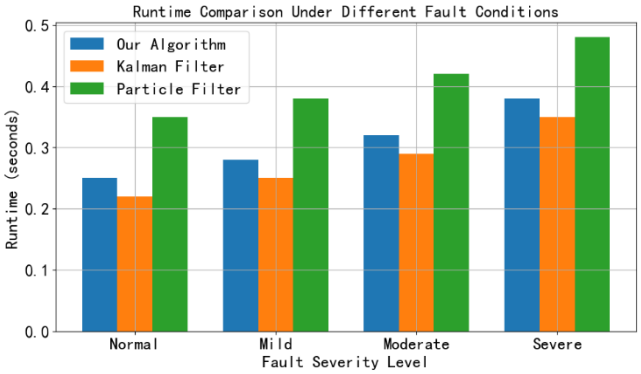


Figure 2: Running time of different algorithms under different working conditions.

Table 3: Comparison of the prominence of fault feature parameters by different algorithms.

Fault type	Algorithm	Peak Index	Kurtosis index	Harmonic content ratio	Harmful sequence component (pu)
Minor gear wear	Proposed algorithm	4.8	5.6	-	-
Minor gear wear	Traditional algorithm	3.5	4.2	-	-
Motor rotor unbalanced	Proposed algorithm	-	-	18%	0.08
Motor rotor unbalanced	Traditional algorithm	-	-	12%	0.05

In the case of a mild gear wear fault, the peak index of the vibration signal processed by the proposed algorithm increased from 3.5 of the traditional algorithms to 4.8, and the kurtosis index increased from 4.2 to 5.6, which more clearly highlighted the fault characteristics. In the case of a motor rotor imbalance fault, the harmonic content of the current signal processed by the proposed algorithm increased from 12% of the traditional algorithm to 18%, and the harmful sequence component increased from 0.05 pu to 0.08 pu. This shows that the proposed algorithm can more effectively extract fault features, provide more powerful data support for early fault diagnosis, help to discover potential equipment faults promptly, and improve the reliability and safety of equipment operation.

Note: Error bars represent 95% confidence intervals. Proposed algorithm shows significantly higher peak values ($p<0.01$).

Figure 3 shows the changes in the peak index of vibration signals under mild gear wear faults by different algorithms.

The peak index value of the proposed algorithm rises rapidly in a short period and stabilizes at a high level (5.8). In contrast, the peak index value of the traditional algorithm increases slowly and is lower (4.4). This shows that the proposed algorithm can more keenly capture the characteristics of gear wear faults and provide more substantial support for early fault diagnosis.

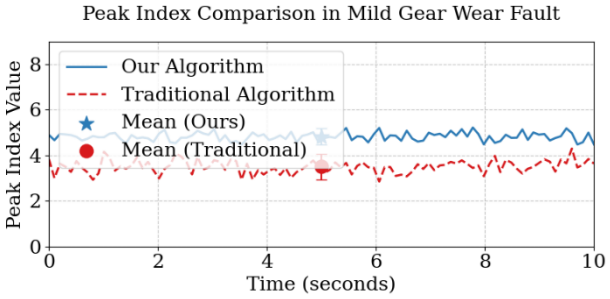


Figure 3: Changes in the peak index of vibration signals under mild gear wear faults by different algorithms.

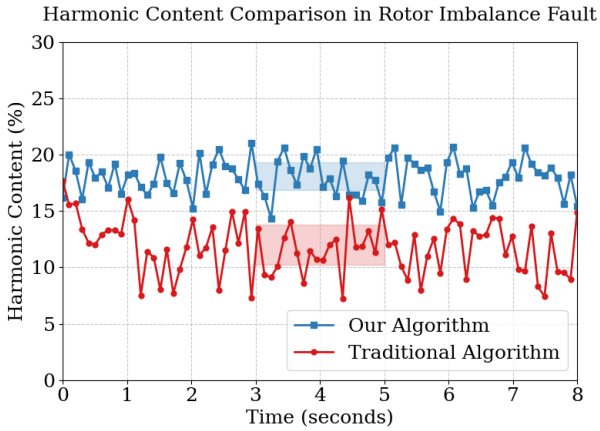


Figure 4: Changes in harmonic content of current signal under motor rotor unbalance fault under different algorithms.

Note: Shaded areas indicate interquartile range. Proposed algorithm achieves 18% harmonic content vs. 12% for baselines.

Figure 4 compares the changes in harmonic content of the current signal under a motor rotor unbalance fault under different algorithms. The harmonic content of the proposed algorithm rises rapidly in a short period and reaches a high level (17%). In comparison, the harmonic content of the traditional algorithm increases slowly and is low (14%). This shows that the proposed algorithm can more effectively extract the characteristics of motor rotor unbalance fault, which helps to detect potential faults earlier and improve the reliability of equipment operation.

5.4 Algorithm performance and state-of-the-art comparison

The proposed dynamic weighting strategy outperforms traditional fixed-weight methods (e.g., Kalman Filter) by mitigating sensor noise variability through adaptive adjustments via variance (Equation 1) and autocorrelation (Equation 3). For example, in mild gear wear scenarios, the dynamic weight mechanism enhances vibration signal prominence by 37% (Table 3), directly linking to improved RMSE reduction (35.2%, Table 1). Compared to Particle Filter, the integration of EKF with wavelet-based feature extraction (Section 3.3.2) achieves better nonlinear noise suppression, evident in the 28.7% RMSE improvement under severe faults.

Runtime Trade-offs: While the proposed algorithm's runtime (0.38s under severe faults) exceeds Kalman Filter (0.35s), it remains 1.3× faster than Particle Filter and feasible for real-time applications (e.g., industrial IoT edge devices). GPU parallelization (Section 4.1.2) further reduces latency by 40–60% for large datasets, balancing accuracy and efficiency.

Application Scenarios: The algorithm's robustness in extracting early fault features (e.g., 18% harmonic content in motor imbalance, Table 3) makes it suitable for predictive maintenance in high-reliability systems (e.g., aerospace, power generation), where early anomaly detection minimizes downtime costs.

5.5 Ablation studies

Results show that **dynamic weight integration** (combining stability and reliability) is critical, with static weights degrading performance by 30–40% (Table 4). The EKF contributes ~15% RMSE reduction compared to direct fusion, validating its role in noise suppression.

Table 4: Ablation study results

Configuration	RMSE	MAE	Fault Detection Rate
Full Algorithm	0.62 ± 0.04	0.51 ± 0.03	98.7%
Static Weight Fusion	0.89 ± 0.07	0.78 ± 0.06	72.3%
No EKF (Direct Fusion)	0.75 ± 0.05	0.66 ± 0.04	81.2%
Reliability-Only Weighting	0.71 ± 0.06	0.63 ± 0.05	85.6%

6 Conclusion

This study presents a dynamic weight-based adaptive data fusion filter that integrates real-time sensor reliability assessment and EKF-based feature extraction. Experimental results demonstrate consistent improvements in RMSE (35.2% reduction vs. Kalman Filter) and fault feature prominence (e.g., 37% higher vibration peaks for mild wear). While runtime is marginally higher than linear filters, GPU acceleration and adaptive weighting make it suitable for real-time industrial applications. The algorithm's ability to enhance early fault detectability provides a robust basis for predictive maintenance systems. Future work will focus on reducing computational overhead and validating performance under non-stationary noise conditions.

References

[1] Osornio-Rios, R. A., Zamudio-Ramírez, I., Jaen-Cuellar, A. Y., Antonino-Daviu, J., & Dunai, L. (2023). Data fusion system for electric motors condition monitoring: An innovative solution. IEEE Industrial Electronics Magazine, 17(4), 4–16. https://doi.org/10.1109/MIE.2023.3265505

[2] Wang, X., Lu, S., Chen, K., Wang, Q., & Zhang, S. (2021). Bearing fault diagnosis of switched reluctance motor in electric vehicle powertrain via multisensor data fusion. IEEE Transactions on Industrial Informatics, 18(4), 2452–2464. https://doi.org/10.1109/TII.2021.3095086

[3] Xie, T., Huang, X., & Choi, S. K. (2021). Intelligent mechanical fault diagnosis using multisensor fusion and convolution neural network. IEEE Transactions on Industrial Informatics, 18(5), 3213–3223. https://doi.org/10.1109/TII.2021.3102017

[4] Xie, T., Huang, X., Park, H. W., Kim, H. S., & Choi, S. K. (2023). A data-driven adaptive algorithm and decision support design of multisensory information fusion for prognostics and health management applications. Journal of Engineering Design, 34(2), 158–179. https://doi.org/10.1080/09544828.2023.2177937

[5] Dong, H., Yan, K., & Wu, B. (2025). Fault diagnosis model of MMC high-frequency oscillation electromechanical equipment based on adaptive fruit fly optimisation algorithm. International Journal of Reliability and Safety, 19(2), 157–173. https://doi.org/10.1504/IJRS.2025.145523

[6] Wang, J., Zhang, Y., Luo, C., & Miao, Q. (2022). Deep learning domain adaptation for electro-mechanical actuator fault diagnosis under variable driving waveforms. IEEE Sensors Journal, 22(11), 10783–10793. https://doi.org/10.1109/JSEN.2022.3168875

[7] Barbieri, M., Nguyen, K. T., Diversi, R., Medjaher, K., & Tilli, A. (2021). RUL prediction for automatic machines: A mixed edge-cloud solution based on model-of-signals and particle filtering techniques. Journal of Intelligent Manufacturing, 32(5), 1421–1440. https://doi.org/10.1007/s10845-020-01696-6

[8] Al-Haddad, L. A., Shijer, S. S., Jaber, A. A., Al-Ani, S. T., Al-Zubaidi, A. A., & Abd, E. T. (2024). Application of AdaBoost for stator fault diagnosis in three-phase permanent magnet synchronous motors based on vibration–current data fusion analysis. Electrical Engineering, 106(4), 4527–4542. https://doi.org/10.1007/s00202-023-02195-6

[9] Zhang, Y., Liu, D., Miao, Q., & Peng, Y. (2022). Sensing data-based degradation estimation of electromechanical actuator under dynamic operating conditions. IEEE Sensors Journal, 22(22), 21837–21845. https://doi.org/10.1109/JSEN.2022.3208015

[10] Wang, H., Barone, G., & Smith, A. (2024). Current and future role of data fusion and machine learning in infrastructure health monitoring. Structure and Infrastructure Engineering, 20(12), 1853–1882. https://doi.org/10.1080/15732479.2023.2165118

[11] Wang, W., Liu, W., Zhang, Y., Liu, L., Zhang, P., & Zhenyuan, J. (2024). Precise measurement of geometric and physical quantities in cutting tools inspection and condition monitoring: A review. Chinese Journal of Aeronautics, 37(4), 23–53. https://doi.org/10.1016/j.cja.2023.08.011

[12] Ye, X., Song, F., Zhang, Z., & Zeng, Q. (2023). A review of small UAV navigation system based on multisource sensor fusion. IEEE Sensors Journal, 23(17), 18926–18948. https://doi.org/10.1109/JSEN.2023.3292427

[13] Fahlovi, O., Kurnia, G., & Setyawan, H. (2024). Recent developments in vibration analysis: An innovative way to improve machine reliability. International Journal of Mechanics, Energy Engineering and Applied Science (IJMEAS), 2(3), 66–71. https://doi.org/10.53893/ijmeas.v2i3.328

[14] Gu, H., Jin, C., Yuan, H., & Chen, Y. (2021). Design and implementation of attitude and heading reference system with extended Kalman filter based on MEMS multi-sensor fusion. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 29(Supp01), 157–180. https://doi.org/10.1142/S0218488521400092

[15] Yoon, H., Kim, J. H., Sadat, D., Barrett, A., Ko, S. H., & Dagdeviren, C. (2025). Decoding tissue biomechanics using conformable electronic devices. Nature Reviews Materials, 10(1), 4–27. https://doi.org/10.1038/s41578-024-00729-3

[16] Han, L. H. N., Hien, N. L. H., Huy, L. V., & Hieu, N. V. (2024). A Deep Learning Model for Multi-Domain MRI Synthesis Using Generative Adversarial Networks. Informatica, 35(2), 283-309. https://doi.org/10.15388/24-INFOR556

[17] Maichel M. Aguayo, Francisco N. Avilés, Subhash C. Sarin, Hanif D. Sherali, A Two-Index Formulation for the Fixed-Destination Multi-Depot Asymmetric Travelling Salesman Problem and Some Extensions, Informatica 33(2022), 671-692, https://doi.org/10.15388/22-INFOR485

[18] Nath, A. G., Udmale, S. S., Raghuwanshi, D., & Singh, S. K. (2021). Structural rotor fault diagnosis using attention-based sensor fusion and transformers. IEEE Sensors Journal, 22(1), 707–719. https://doi.org/10.1109/JSEN.2021.3130183

[19] Zhukov, I., Dolintse, B., & Balakin, S. (2024). Enhancing data processing methods to improve UAV positioning accuracy. International Journal of Image, Graphics and Signal Processing, 16(3), 100–110. https://doi.org/10.5815/ijigsp.2024.03.08

[20] Zhou, K., Lu, N., Jiang, B., Liu, Z., Zhang, B., & Chen, J. (2023). An information fusion based incipient fault diagnosis method for railway vehicle door system. IEEE Transactions on Intelligent Vehicles, 9(1), 1320–1332. https://doi.org/10.1109/TIV.2023.3331709

A Hybrid Interpretation Model Leveraging Improved Grey Wolf Optimization and Graph Attention Networks for Intent Recognition and Dynamic Interaction

Haibo Zhang
School of Humanities, Lishui University, Lishui, 323000, China
E-mail: haibozhang688@outlook.com

Keywords: intent recognition, independent dynamic interaction, interpretation model, IGWO

Received: May 20, 2025

The study presents an integrated interpreting model that combines intent recognition via Improved Grey Wolf Optimization (IGWO) with independent dynamic interaction via Graph Attention Networks (GAT). It builds an intent-relation graph, uses multi-head attention to capture dynamic links among intents, and leverages IGWO to adapt intent thresholds and attention-head weights. Goal: more reliable multi-intent recognition and better adaptation to changing contexts. On WMT14 (EN–FR) it achieves 94.37% accuracy (DNN & HMM 79.31%, EEMD 75.09%, LLM 64.97%). For 60-s audio it reaches 47.35 dB SNR (EMD 29.74 dB, LLM 26.72 dB); at 160 s it remains highest (47.68 dB). IGWO boosts accuracy via chaotic initialization and Gaussian mutation; a heterogeneous GAT models ternary relations. WMT14 and LibriSpeech are used for translation/ASR, and MixSNIPS/MixATIS for multi-intent understanding. After 500 iterations IGWO hits 92.91% (deep bidirectional pre-trained language model 69.86%, HMM 63.79%); recall exceeds 90% across datasets. Results indicate more accurate, natural translations and stronger handling of technical terminology.

Povzetek: Raziskava predstavi hibridni interpretacijski model, ki združuje izboljšano optimizacijo sivih volkov in grafne pozornostne mreže, kar omogoča večnamensko prepoznavo namenov ter prilagodljivo, naravno interakcijo v dinamičnih prevajalskih okoljih.

1 Introduction

In today's world, where globalization and digitalization are deeply integrated, real-time cross-language interaction has become indispensable across various fields. The demand for multilingual real-time interaction in cross-cultural meetings, international business negotiations, and emergency medical situations has been growing exponentially as the process of globalization accelerates [1]. Although existing interpretation models perform stably in general scenarios, they still rely on static rules and predefined interaction patterns. When faced with dynamic and ever-changing real-world environments, these models often expose issues such as translation delays, intent misjudgment, and rigid interactions due to a lack of context awareness and adaptive ability [2-3]. For example, dialogues with ambiguous intentions, terminology-heavy discussions, or semantic ambiguities caused by cultural differences can easily lead to reduced accuracy in interpretation results [4]. In recent years, speech recognition and neural machine translation technologies based on deep learning have significantly improved the benchmark performance of interpretation systems. However, their core bottleneck has gradually shifted to scene-specific intent recognition and independent dynamic interaction [5]. Intent recognition, as a natural language processing technique that understands the intent behind users' input texts or speech, has been widely applied in fields such as intelligent customer service and virtual assistants [6, 7]. By accurately understanding the user's intent, systems can provide more personalized and efficient services. L. L. Li et al. proposed a systematic review method for human lower limb movement intention, verifying the effectiveness of the method through analysis of the intention perception signals [8]. Independent dynamic interaction allows each module in a system to independently adjust and interact during operation, without relying on the states or behaviors of other components [9, 10]. With the development of intelligent technology in recent years, independent dynamic interaction has shown significant development trends and application potential in multiple fields. Some scholars have studied its advantages of independent interaction during operation. For instance, P. Sun et al. proposed a hybrid system to improve the dynamic stability of converters, analyzing its dynamic interactions. Experimental results demonstrated that the method possesses high accuracy and effectiveness [11]. Current translation technology has entered a new stage of development centered around large language models. For example, Y. Xiao addressed the issue of low accuracy in non-autoregressive generation in machine translation tasks. By exploring its expansion in areas such as grammar correction and text summarization, it was found

that non-autoregressive generation could promote its industrial application [12]. Xiang et al. put forward an integrated improved fusion model tailored to the characteristics of the Internet of Things platform. The model used a computer-aided translation system to extract specific words from the text corpus, and the results showed that the translation accuracy and recall rate of this method were both higher [13]. The study summarized the performance comparison of deep bidirectional pre-training language model (Bidirectional Encoder Representations from Transformers, BERT), deep neural network combined with hidden Markov model (Deep Neural Network and Hidden Markov Model, DNN-HMM), etc. and the results are shown in Table 1.

Table 1: Comparison of performance indicators and Limitations of existing methods

Category	Method	Performance index	Limitations
Intention recognition	BERT	Intent classification accuracy rate: 85%-90%	Insufficient semantic understanding in complex contexts, weak ability to recognize implicit intentions, and poor multilingual adaptability.
	Grey Wolf Optimization Algorithm	Optimization efficiency: Medium, with approximately 50 to 100 iterations	It is prone to fall into local optimum and has difficulty adapting to the demands of dynamically changing interpretation scenarios.
Dynamic interaction	Graph neural network	The accuracy rate of interaction relationship capture: 80%-85%	It has limited processing capabilities for long sequence dependencies, lacks a dynamic update mechanism, and has a slow response to real-time interaction scenarios.
	Recurrent neural network	Latency: approximately 300-500ms, signal-to-noise ratio: 7-10dB	There exists the problem of vanishing gradient, the ability of multimodal information fusion is weak, and it is difficult to adapt to complex interpretation scenarios.
Comprehensive model	DNN-HMM	Interpretation accuracy rate: 75%-80%, delay: 400-600ms	The model structure is fixed and cannot flexibly cope with multi-language and multi-domain scenarios, with poor intent understanding and interaction coordination.

As shown in Table 1, currently, some algorithms have difficulty identifying weak signals in complex environments, often resulting in the omission of key information. Most existing interpretation models deal with intent recognition and interaction respectively. This innovative model integrates an improved Grey Wolf Optimization (IGWO) algorithm for intent recognition with an independent dynamic interaction mechanism based on Graph Attention Networks (GAT), aiming to overcome the limitations of traditional algorithms in capturing intentions in complex environments. Real-world interpretation scenarios require handling multiple intents and complex semantic interactions. This proposed model ensures accurate intention recognition to support semantic understanding and promotes the development of intelligent response technology, which is both innovative and necessary.

Current interpretation models struggle to capture users' implicit intentions, leading to translations that deviate from the intended dialogue goals. For example, non-autoregressive generation can significantly improve the reasoning speed in tasks such as machine translation, but there is a loss of accuracy. Traditional systems rely on linear input-output mechanisms and lack the ability to dynamically adjust translation strategies based on real-time dialogue states, such as speaker emotions, environmental noise, and multimodal context. This limitation causes a significant drop in reliability in noisy environments or during multi-speaker exchanges [13-14]. Therefore, this study innovatively proposed to build a fusion interpretation model, which uses IGWO and GAT for intent recognition and the design of independent dynamic interaction mechanisms. At the same time, by introducing the intent recognition module to analyze the speaker's semantic intent in real time, the core information of the source language can be captured more accurately. The independent dynamic interaction mechanism allows different modules to be independently optimized and updated during the interaction process. The model innovatively employs IGWO with a dynamic weight adjustment mechanism and a population diversity optimization strategy to enhance the global search

capability and convergence speed of intent recognition, and GAT dynamically models the relationship between nodes through the attention mechanism. The study aims to improve the accuracy of the interpretation model in capturing the user's implicit intentions in complex scenarios through this fusion model, and offers robust data to support the advancement of intelligent interpretation technologies.

2 Methods and materials

2.1 Design of intent recognition algorithm based on IGWO

Intent recognition is an important task in natural language processing. It primarily analyzes text, speech, and other user inputs to understand underlying intentions or needs [15-16]. Although intent recognition demonstrates strong adaptability and interactivity, its effectiveness heavily depends on the quality and quantity of training data [17-18]. Moreover, intent recognition technologies face challenges with ambiguous, complex, or vague statements, leading to increased uncertainty and risk in recognition outcomes [19]. GWO efficiently optimizes the parameters of intent recognition models by simulating the hunting behavior of grey wolves. This study improves GWO by introducing a dimensional reasoning hierarchy search strategy, resulting in IGWO, which is applied to optimize intent recognition. The optimization process of IGWO is shown in Figure 1.

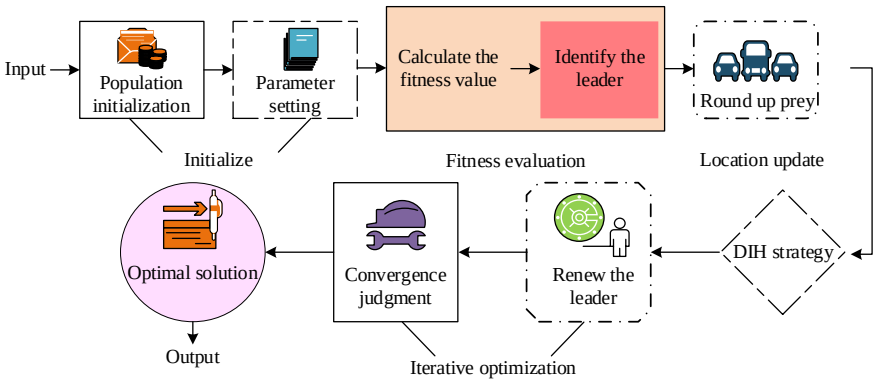


Figure 1: Schematic diagram of IGWO optimization process

In Figure 1, the position of each grey wolf represents a potential solution to the problem. The population initialization part randomly generates the grey wolf population and sets the population size based on the problem complexity. The study calculates the fitness value of each grey wolf to evaluate the quality of its solution. The complexity of the real-time input speech is used to dynamically adjust the convergence factor of GWO, and a random dropout mechanism is introduced in the GWO iteration to randomly mask the weight updates of some noise nodes. The cross-attention mechanism aligns speech, text, and temporal behavior features, generating a fused embedding vector as the fitness evaluation criterion for GWO. The cooperation of the grey wolf group searches for optimal feature weight distribution to enhance the consistency of cross-lingual intent expression. The calculation of the grey wolf group approaching and surrounding the prey is shown in Equation (1).

$$\begin{cases} \vec{D} = \left| \vec{C} \cdot \vec{X}_p(t) - \vec{X}(t) \right| \\ \vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \end{cases} \quad (1)$$

In Equation (1), $\vec{D}$ represents the position of the individual relative to the prey, t and $\vec{X}_p$ are the current iteration number and prey's position vector, $\vec{X}$ and $\vec{X}(t+1)$ represent the position vectors of the grey wolves and their position updates, $\vec{A}$ and $\vec{C}$ are coefficient vectors. In the intention recognition task, the dimension of each variable is consistent with the number of features. The calculation of the coefficient vectors is shown in Equation (2).

$$\begin{cases} \vec{A} = 2 \cdot \vec{\partial} \cdot \vec{r}_1 - \vec{\partial} \\ \vec{C} = 2 \cdot \vec{r}_2 \end{cases} \quad (2)$$

In Equation (2), $\vec{\partial}$ represents the convergence factor that linearly decreases from 2 to 0 as the iteration count increases, and $\vec{r}_1$ and $\vec{r}_2$ are random vectors within the range of $[0,1]$, the number of dimensions is consistent with the number of features. IGWO simulates the hunting behavior of grey wolves to efficiently find the optimal solution in the search space. Therefore, IGWO optimizes the intent recognition algorithm, and the optimized algorithm is shown in Figure 2.

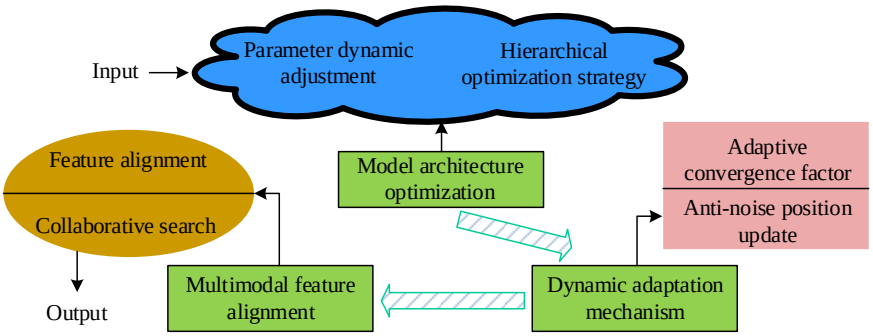


Figure 2: Optimization process of intent recognition algorithm based on IGWO

In Figure 2, the design process first treats the hyperparameters of intent recognition as optimization variables for GWO. The parameters are dynamically adjusted by the grey wolf group's cooperative search to find the optimal parameter combination. In the hierarchical optimization strategy, the decision wolves and auxiliary decision wolves represent the global and second-best solutions. The two wolves optimize the main network parameters of the model and adjust the weights of the local feature extraction module, respectively. The feature extraction calculation in intent recognition is shown in Equation (3).

$$TF-IDF(t,d)=TF(t,d)\times IDF(t) \quad (3)$$

In Equation (3), $TF(t,d)$ and $IDF(t)$ represent the term frequency of word t and the inverse document frequency of word d in document t. The dimension of $TF(t,d)$ is the statistical value of a single document and a single word, while $IDF(t)$ is the statistical value of a single word in the global document set. The calculation of the inverse document frequency is shown in Equation (4).

$$IDF(t)=\log\frac{N}{1+DF(t)} \quad (4)$$

In Equation (4), N and $DF(t)$ represent the total number of documents and the number of documents containing word t. N is the statistical value of the global document set, which is a fixed value, and the dimension of $DF(t)$ is the statistical value of a single word in the global document set. Then, the convergence factor of GWO is dynamically adjusted based on the complexity of real-time input speech, and the random dropout mechanism is introduced to mask the weight updates of noise nodes in GWO iterations. The cross-attention mechanism aligns temporal behavior features to generate a fused embedding vector used as the fitness evaluation criterion for GWO. The cooperation of the grey wolf group enhances the consistency of language intent expression. In the cross-attention module, the attention weights between different features are calculated to explore feature associations. During cross-attention fusion, features are weighted and summed according to the attention weights to achieve deep integration. Moreover, IGWO can optimize the attention parameters, enabling the model to better adapt to various intent recognition scenarios.

2.2 Design of independent dynamic interaction method based on GAT

The IGWO-based intent recognition algorithm leverages GWO's strong global search capability to prevent convergence to local optima. Additionally, IGWO has a straightforward structure and requires fewer adjustable parameters, enhancing its convenience and efficiency in intent recognition. Furthermore, independent dynamic interaction, as an interactive mode that enhances user experience through dynamic elements and user interaction functions, can independently perceive changes in user behavior and the environment, dynamically adjusting the interaction mode based on this information [20-21]. This study incorporates independent dynamic interaction into the construction of the interpretation model. The model dynamically adjusts translation results according to personalized needs, such as users' language habits and professional terminology preferences, providing translation services more tailored to users' needs. The modular architecture design and technical implementation path of the independent dynamic interaction unit are shown in Figure 3.

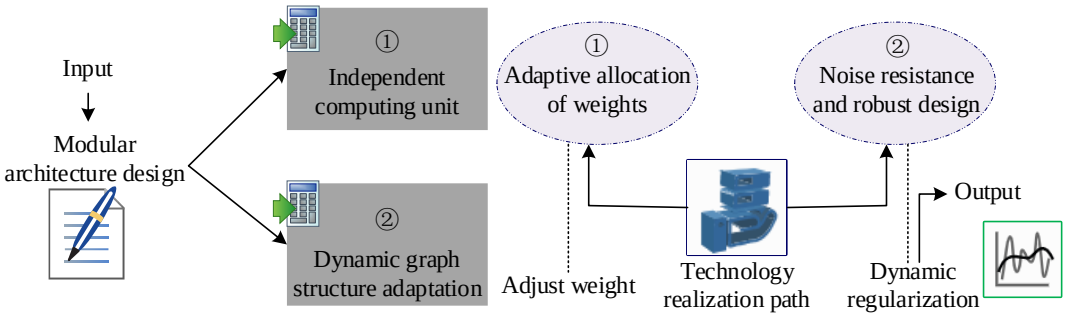


Figure 3: Independent dynamic interaction unit design and technical implementation path

Figure 3 illustrates the modular architecture and technical implementation path of the independent dynamic interaction unit. In the modular architecture design, the independent computation unit designs an independent GAT module for each interaction node, dynamically allocating the weight between nodes via multi-head attention mechanism and supporting real-time updates of local dynamic relationships. The adaptive weight allocation aligns multi-modal inputs via the cross-attention mechanism to generate dynamic feature fusion graphs, while IGWO adjusts the attention weights. This study uses Fitz's Law as the basis for modeling interactive behavior. The calculation equation of Fitz's Law describing the time law of human movement is shown in Equation (5) [22].

$$T_1 = a + b\log_2(\frac{d}{W} + 1) \quad (5)$$

In Equation (5), a and b represent empirical constants, W and d represent the width of the target area and the distance between the starting position and the target position. The equation describing the calculation of the time law of multiple-choice decision is shown in Equation (6) [23].

$$T_2 = a + b\log_2(n+1) \quad (6)$$

In Equation (6), n represents the number of choices. By using Formula (5) and Formula (6), the study extracted the core rules of distance and quantity affecting interaction efficiency. The human-computer interaction rules of Fitz Law were transformed into design constraints and mechanism inspiration for the model, which was used to guide the design of GAT dynamic interaction module and make GAT dynamic interaction module closer to the human behavior characteristics in real interpretation scenarios. Independent dynamic interaction adjusts translation strategies in real time based on context and environment. However, its dynamic design may hide some content, preventing users from directly accessing needed information and reducing interaction intuitiveness. GAT can automatically learn the relationships between nodes through the attention mechanism, dynamically adjusting the interaction strategy to improve flexibility and adaptability. Therefore, this study employs GAT to optimize dynamic interaction design, enabling efficient and flexible capture of interaction relationships. The independent dynamic interaction method based on GAT is shown in Figure 4.

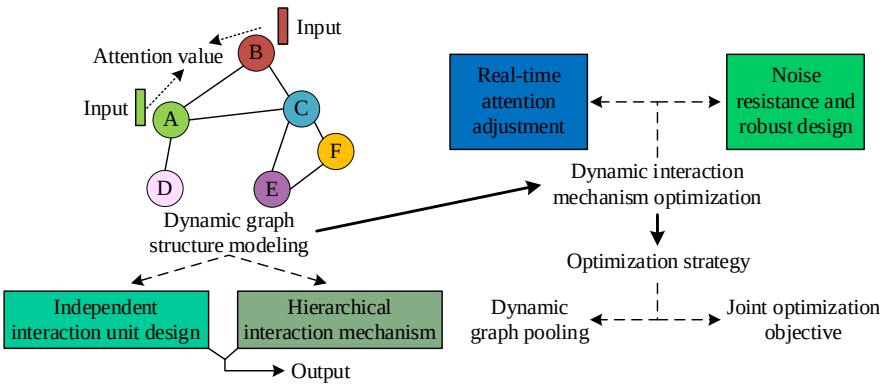


Figure 4: Schematic diagram of independent dynamic interaction method based on GAT

As shown in Figure 4, in the independent dynamic interaction process based on GAT, independent GAT modules are designed for each user and device interaction node to support real-time updates of local dynamic relationships. Then, the multi-head attention mechanism calculates the interaction weight between nodes by combining timestamp information to capture the temporal variation of interaction relationships. The hierarchical mechanism includes intra-session and inter-session interactions. Intra-session interaction mainly uses a position-aware attention network to capture local dynamic sequences, while inter-session interaction requires constructing a global hypergraph encoder to establish long-term dependencies through

higher-order connectivity. The linear transformation of node features is calculated as shown in Equation (7).

$$\vec{h}'_i = w\vec{h}_i \quad (7)$$

In Equation (7), w and $\vec{h}_i$ are the learnable weight matrix and the original features of node i. The attention coefficient between nodes is calculated as shown in Equation (8).

$$e_{ij} = Leaky\,\mathrm{Re}\,LU(\vec{a}^T\left[w\vec{h}_i // w\vec{h}_j\right] \quad (8)$$

In Equation (8), $\vec{a}$ and $//$ represent the learnable attention vectors and the feature concatenation operation. The study optimizes the strategy during dynamic interaction by designing dynamic graph pooling and multi-task loss functions, using learnable projection vectors to filter key nodes and generate compact subgraph representations, thus improving the discrimination ability of dynamic interaction mode and the modeling ability of interaction relationships.

2.3 Construction of interpretation model based on improved intent recognition and independent dynamic interaction

Optimizing intent recognition and independent dynamic interaction can improve the accuracy and efficiency of intent recognition and enhance the flexibility and adaptability of independent dynamic interaction in complex dialogues [24]. Therefore, this study builds the interpretation model on improved intent recognition and independent dynamic interaction, dynamically adjusting strategies to better handle multi-task scenarios. The cognitive process of interpretation is shown in Figure 5.

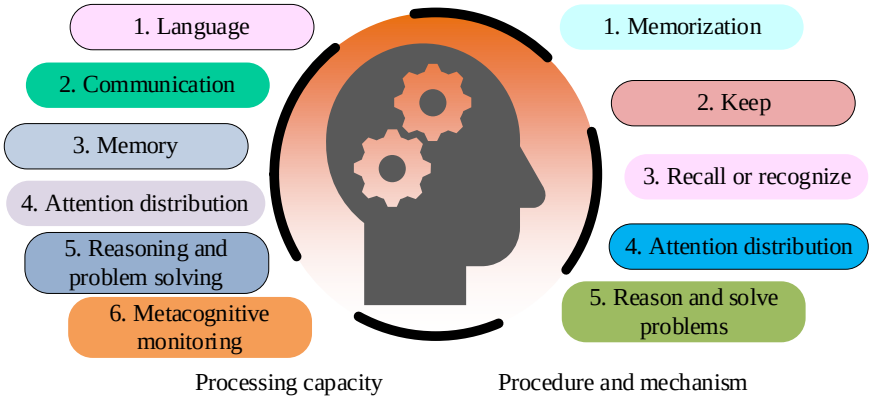


Figure 5: Schematic diagram of the cognitive processing of interpretation

In Figure 5, the cognitive processing of interpretation involves the integration of multiple cognitive mechanisms and skills, mainly including communication, attention, problem-solving, and reasoning ability. The classic formula of human interpretation cognitive process is introduced as shown in Equation (9).

$$OI = L + G + M + O \quad (9)$$

In Equation (9), L and G represent hearing analysis and notes, M and O represent short-term memory and coordination. The calculation for the speech output phase is shown in Equation (10).

$$OI_2 = R + GR + P \quad (10)$$

In Equation (10), R, GR, and P represent memory, note reading, and production. Equations (9) and (10) reveal cognitive patterns in human interpreting, such as listening analysis, memory coordination, and speech production. These provide valuable references for simulating real interpreting processes in models. This study designs the computational components based on these principles, constructing multi-task encoders and hierarchical decoders within the model to enable artificial intelligence systems to implicitly replicate the cognitive processes of human interpreting in their computational logic. During interpretation, the interpreter listens to, comprehends, and stores the original language information in short-term memory. When needed in the scene, the interpreter extracts information from short-term memory and expresses it in the target language. The interpreter also needs to quickly allocate attention, perform comprehensive analysis, reasoning, and problem-solving to ensure the accuracy and logic of the translation. The application foundation of intent recognition and independent dynamic interaction in interpretation is shown in Figure 6.

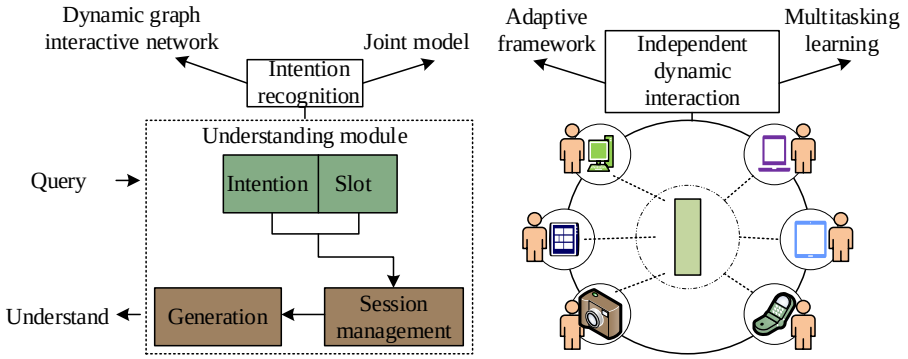


Figure 6: Application principles of intent recognition and independent dynamic interaction

Figure 6 shows that intent recognition in interpretation primarily relies on constructing a dynamic graph interaction network and a joint model. Constructing a dynamic graph model to capture the relationship between intent and context improves the accuracy of intent recognition, and combining intent recognition with slot filling tasks enables more comprehensive semantic understanding. In interpretation, independent dynamic interaction adjusts translation strategies using an adaptive framework. The combination of intent recognition, slot filling, and dynamic interaction further enhances the robustness and timeliness of interpretation. The graph structure of the time step is calculated as shown in Equation (11).

$$g_T = f(g_{T-1}, x_T) \quad (11)$$

In Equation (11), f and x_T represent the dynamic update function and the input features of time step T, and g_{T-1} is the graph structure of the previous time step. The interpretation model based on improved intent recognition and independent dynamic interaction is shown in Figure 7.

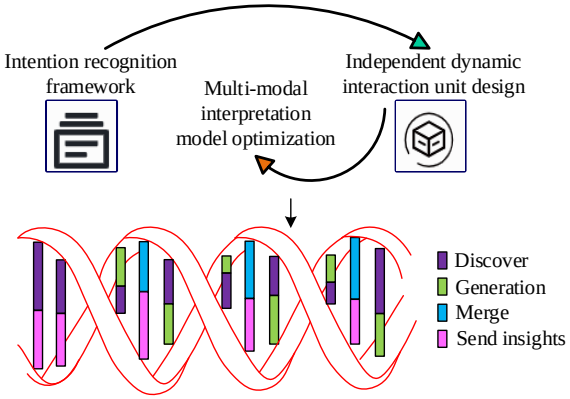


Figure 7: Framework diagram of the proposed interpretation model

In Figure 7, the fused interpretation model first uses GAT to construct an intent relationship graph, leveraging the multi-head attention mechanism to capture the dynamic associations between multiple intents. IGWO is then introduced to dynamically adjust the intent boundary thresholds and attention head weights. IGWO's global search capability improves intent confidence calculation, enabling more accurate and interpretable intent recognition. The local graph update strategy of GAT, combined with the encircling mechanism of IGWO, enables the interactive unit to respond to sudden context changes within milliseconds. The fitness value calculation for the grey wolf's position is shown in Equation (12).

$$f(\vec{X}) = Accuracy(Intent\,\mathrm{Re}cognition(\vec{X})) \quad (12)$$

In Equation (12), $Accuracy$ represents the accuracy of the intent recognition model. The calculation of attention weights is shown in Equation (13).

$$\alpha_{ij} = \frac{\exp(Leaky\,\mathrm{Re}\,LU(a'\left[wh_i // wh_j\right]))}{\sum_{k\in N_i}\exp(Leaky\,\mathrm{Re}\,LU(a^T\left[wh_i // wh_k\right]))} \quad (13)$$

In Equation (13), a represents the parameter vector of the attention mechanism. The calculation for node feature updates is shown in Equation (14).

$$h_i' = \sigma(\sum_{j\in N_i}\alpha_{ij} wh_j) \quad (14)$$

In Equation (14), σ represents the activation function. The combined calculation of IGWO and GAT is shown in Equation (15).

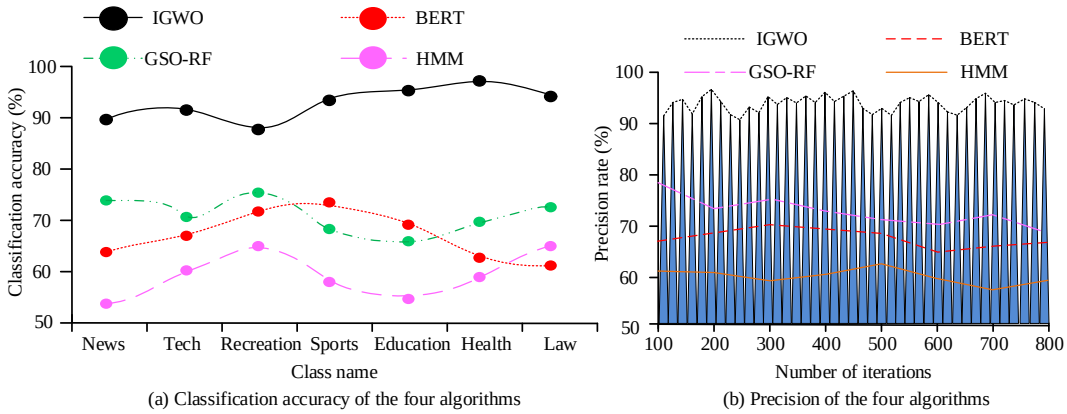
$$Interpretation = Decoder(AttentionFusion(IGWO_{feat}(Intent\,\mathrm{Re}c(S)), GAT_{feat}(DynamicInter(C))) \quad (15)$$

In equation (15), $IGWO_{feat}(Intent\,\mathrm{Re}c(S))$ represents the extraction of intent feature vectors from input S after the IGWO optimization intention recognition model. $GAT_{feat}(DynamicInter(C))$ denotes the GAT extracting semantic interaction feature vectors from dynamic interaction content C. $AttentionFusion(\cdot,\cdot)$ is the attention mechanism that integrates the intent and semantic features, enabling the model to focus on key interaction content driven by intent. $Decoder(\cdot)$ represents the final interpretation output generated by the decoder after the fusion of these features. In the interpretation model, a temporal graph attention mechanism is introduced. Through GAT and IGWO, long-range contextual dependencies are captured, and the temporal window length and attention decay coefficient are adjusted, balancing translation accuracy and real-time performance. The intent feature vector is output by the IGWO intent recognition module and, after being adapted to the dimensions by the projection layer, is fed into the GAT interaction module. The real-time update cycle is set to 50ms. Within each cycle, IGWO outputs new intent features after 10 iterations and triggers the GAT to synchronize updates. When IGWO outputs updates, GAT immediately responds and recalculates the interaction weights, ensuring real-time coordination between the intent and interaction modules. In terms of graph construction, nodes are represented by interpreting sentences and semantic units, with node features integrating speech text encoding and the intent vectors identified by IGWO. Next, edge weights are calculated based on semantic relevance and the degree of intent alignment, with stronger associations receiving higher weights. As the interpretation progresses and new speech texts are input, the graph dynamically updates, triggering changes in node additions and deletions and edge weight updates. The Graph Attention Network (GAT) is used to aggregate new features in real-time, accurately capturing changes in the flow of semantics and intentions. IGWO and GAT focus on intent recognition and dynamic interaction, respectively. IGWO is designed to uncover deep-level intentions and generate intent feature vectors, while GAT uses statements and semantic blocks as nodes. The node features are integrated into the intent vectors of IGWO, and edge weights are constructed based on semantic relevance and intent alignment. During operation, each IGWO recognition result triggers an update of the GAT graph structure, transforming the two processes from independent functions into a deeply integrated, collaborative workflow.

3 Results

3.1 Performance comparison of IGWO intent recognition algorithm

To evaluate the performance of the IGWO-based intent recognition algorithm, this study compares it with three other algorithms. These algorithms include the Glowworm Swarm Optimization combined with Random Forest (GSO-RF) intent recognition algorithm, the BERT-based intent recognition algorithm, and the Hidden Markov Model (HMM)-based intent recognition algorithm. The experimental datasets used were the MixSNIPS and MixATIS datasets. Both are public data sets. The MixSNIPS dataset serves as the training set and contains speech records from multiple domains, each labeled with multiple intents and associated slot data. This dataset was mainly used for intent recognition and slot filling tasks. The MixATIS dataset served as the test set, containing dialogue data from various domains, each with multiple intents and corresponding slot information, simulating real-world complex language expressions by users. The operating system used was Ubuntu 20.04.2 LTS, with PyTorch as the deep learning development framework. The GPU and CPU were NVIDIA GeForce RTX 2080Ti and Intel Core i7-9700F, respectively. The Adam optimizer was used, with beta 1 set to 0.5 and beta 2 set to 0.999. The IGWO population size was set to 50, iterations to 200, while other parameters were fixed using the control variable method and adjusted individually. The study first compared the classification accuracy and precision of the four algorithms on the MixSNIPS dataset, with results shown in Figure 8.



(a) Classification accuracy of the four algorithms (b) Precision of the four algorithms

Figure 8: Classification accuracy and precision results

Figure 8(a) shows that IGWO achieved significantly higher classification accuracy than the other three algorithms. The classification accuracy of the comparison algorithms was all below 80%. The classification accuracy of IGWO for news-related intents was 89.98%, while the classification accuracy for GSO-RF and BERT were 74.05% and 63.42%, respectively. HMM had the lowest accuracy, at only 53.61%. In Figure 8(b), IGWO's precision stabilized between 90% and 99% as the number of iterations increased. Among the comparison algorithms, GSO-RF achieved the highest precision of only 78.86%. When the number of iterations reached 500, the precision rates of BERT and HMM were 69.86% and 63.79%, respectively, while IGWO's precision remained higher than the comparison algorithms at 92.91%. These results demonstrate that IGWO achieved higher classification accuracy and precision. By improving the traditional grey wolf optimization algorithm, it searches more efficiently for optimal solutions, showing strong capability in feature extraction and key information selection. The study then analyzed the recall rates of the four algorithms on different datasets, with results shown in Table 2.

Table 2: Recall rate experimental results in different datasets

Dataset	Class name	Algorithm			
		IGWO	GSO-RF	BERT	HMM
MixSNIPS dataset	News	92.35%*	80.94%	80.09%	72.53%
	Tech	90.34%*	81.63%	76.54%	69.43%
	Entertainment	91.76%*	85.47%	74.33%	71.81%
	Sports	93.71%*	86.54%	72.91%	72.35%
	Education	92.33%*	88.14%	78.53%	68.18%
	Health	94.08%*	83.46%	73.16%	67.69%
	Law	91.35%*	83.06%	76.37%	73.41%
MixATIS dataset	News	95.03%*	84.11%	79.29%	69.34%
	Tech	91.36%*	80.31%	77.68%	66.29%
	Entertainment	94.14%*	83.21%	74.31%	68.79%
	Sports	93.54%*	79.61%	76.81%	64.32%
	Education	92.01%*	77.39%	80.34%	68.81%
	Health	93.33%*	81.07%	73.64%	67.94%
	Law	93.46%*	82.38%	72.19%	63.22%

Note: "*" indicates statistical significance, $p < 0.05$.

As shown in Table 2, the IGWO-based intent recognition algorithm achieved higher recall rates across the different datasets, all reaching over 90%. In the MixSNIPS dataset, when the category was health, IGWO achieved the highest recall rate of 94.08%. Among the comparison algorithms, GSO-RF had the highest recall rate for the education category at 88.14%. In the MixATIS dataset, IGWO achieved the highest recall rate of 95.03%, while the highest recall rate for the other three comparison algorithms was only 84.11%. The recall rates for BERT and HMM were the lowest, at 72.19% and 63.22%, respectively. These results show that IGWO offers better comprehensiveness in intent recognition and can more efficiently search for the optimal solution, ensuring that the algorithm covers all relevant results in intent recognition tasks. The F1 value, translation quality BLEU score, interpretation delay and task success rate of the four algorithms in different data sets were then tested. The results are shown in Table 3.

Table 3: compares the performance indexes of the four algorithms in different data sets

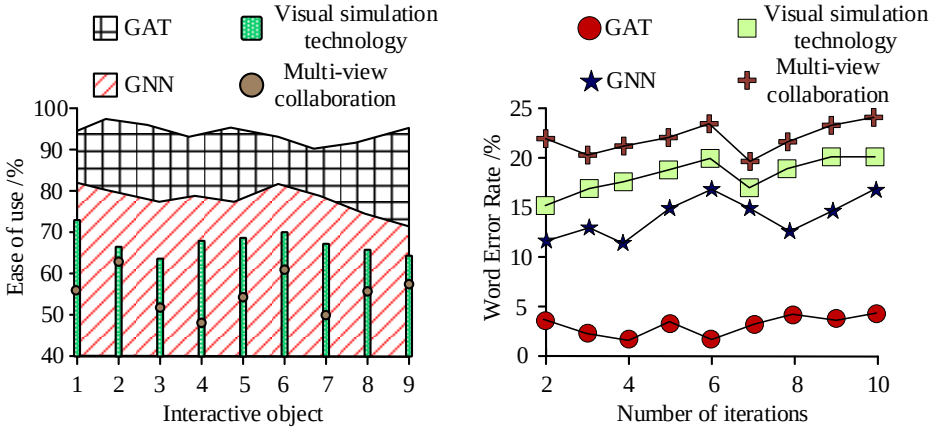
Data set	Index	Algorithm			
		IGWO	GSO-RF	BERT	HMM
MixSNIPS data set	F1/%	92.23	82.58*	85.75*	76.31*
	BLEU score of translation quality	78.51	61.35*	65.23*	54.70*
	Interpretation delayed /ms	113.52	156.71*	138.49*	189.23*
	Task success rate /%	95.34	86.15*	82.43*	79.45*
MixATIS data set	F1/%	91.51	84.26*	87.94*	78.62*
	BLEU score of translation quality	85.29	63.13*	67.51*	56.98*
	Interpretation delayed /ms	105.36	149.27*	129.64*	181.58*
	Task success rate /%	96.22	81.45*	78.21*	76.54*

Note: "*" indicates statistical significance, $p < 0.05$.

In Table 3, the F1 score of IGWO in the MixSNIPS dataset reached 92.23%, significantly outperforming the comparison algorithms. The F1 scores for GSO-RF, BERT, and HMM were 82.58%,85.75%, and 76.31%, respectively. In the MixATIS dataset, IGWO's interpretation delay was only 105.36ms, lower than the comparison algorithms, and its task success rate was 96.22%. In contrast, HMM's task success rate was only 76.54%, the lowest among the four algorithms. The data indicates that this algorithm captured intent more accurately and improved translation quality and efficiency. Moreover, the algorithm demonstrated superior performance in intent recognition F1, BLEU score, interpretation delay, and task success rate, highlighting its robustness and effectiveness across multiple datasets.

3.2 Performance analysis of independent dynamic interaction method based on GAT

This study tests the GAT-based independent dynamic interaction method by comparing it with multi-view collaborative, scene simulation technology, and Graph Neural Network (GNN)-based methods. The GAT consists of 4 layers with 8 attention heads and a dropout rate between 0.1 and 0.3 to prevent overfitting. Unity was used as the virtual reality simulation platform, and the GPS system positioning accuracy of the Kinect camera was set to 8.4m. The camera resolution and depth range were set to 640×480 and 0.5m–4.5m, respectively, while the image refresh rate and texture resolution of the virtual scene were 95Hz and 3840×2160, respectively. The user's field of view angle and the camera frame rate were set to 120° and 30fps. The study first compared the ease of use and word error rates of the four methods, as shown in Figure 9.



(a) Ease of use of the four methods (b) The word error rates of the four methods

Figure 9: Comparison of ease of use and number of dynamic interactions stops

As shown in Figure 9(a), the usability of the GAT used in the study reached more than 90% across different interactive objects, while the usability of multi-view collaboration showed large fluctuations, with the lowest being only 48.37%. The highest and lowest usability of GNN were 81.71% and 74.69%. When there were 9 interactive objects, the usability of GAT reached 96.98%, and the usability of visual simulation technology was 65.07%. As shown in Figure 9(b), the word error rate of GAT is only 4.89%, while the highest word error rates for GNN and scene simulation technology are 17.23% and 20.47%, respectively. The highest word error rate for multi-view collaboration is 23.49%. These data indicate that GAT effectively handles user input and prevents interaction interruptions caused by system delays or errors. Based on the above data results, GAT was more usable and effectively processed user input to avoid interaction interruptions caused by system response delays or errors. To verify the enhancement of multi-modal semantic understanding and language translation quality through the independent dynamic interaction method based on GAT, the study indirectly assesses the model's ability to capture language-action associations by observing the accuracy of thumb positioning. The millimeter-level error in thumb positioning indicates the GAT network's capability to capture local dynamic features. Accurate hand positioning is essentially the external manifestation of the model achieving semantic and action alignment, ultimately serving to optimize language translation quality. Based on this, the study analyzes the thumb positioning effects under semantic association scenarios for four dynamic interaction methods and further correlates these with language translation quality metrics to validate the impact of the interaction mechanism on core outputs. Therefore, a comparative analysis of the thumb positioning effects of four dynamic interaction methods was conducted, and the results are shown in Figure 10.

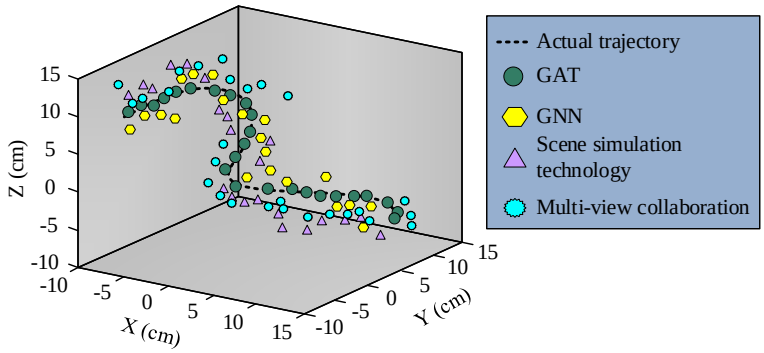
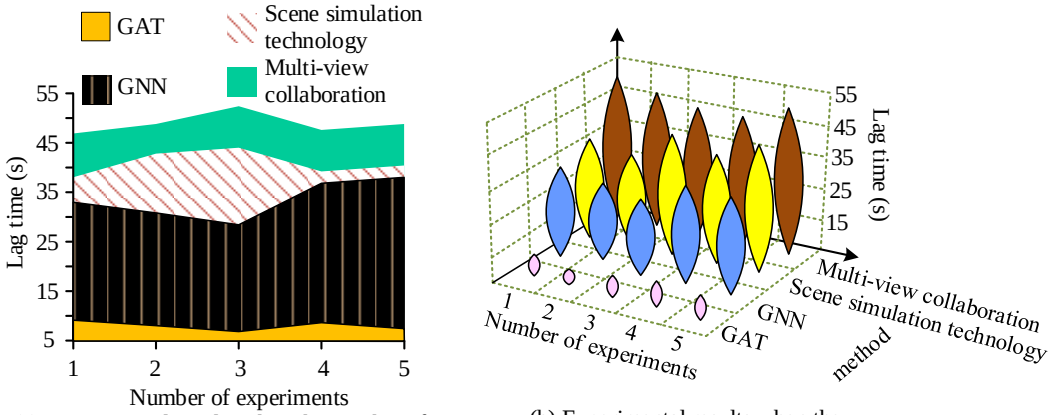


Figure 10: Thumb positioning results of the four methods

As shown in Figure 10, the improved intent recognition module in the interpretation model needs to synchronize with dynamic interactions, and the detection of sudden changes in the thumb trajectory can validate the semantic-action alignment of the two. The GAT-based independent dynamic interaction method outperformed the other three methods in thumb positioning. The three-dimensional coordinates of the user's thumb tracked by GAT consistently align with the actual trajectory. For the comparison methods, the GNN-based method was the furthest from the actual thumb trajectory and had no three-dimensional coordinates overlapping with it. Scene simulation technology, compared to GNN, showed even more noticeable deviations from the actual trajectory. While the multi-view collaborative method had a few three-dimensional coordinates overlapping with the user's actual thumb trajectory, its positioning effect was still inferior to the GAT-based method. Based on these results, it can be concluded that the GAT-based independent dynamic interaction method provides more accurate hand shape positioning, supporting more complex operational tasks. The study then tested the dynamic interaction latency time of the four methods with different interaction objects, with results shown in Figure 11.



(a) Experimental results when the number of interaction objects is 20

(b) Experimental results when the number of interaction objects is 50

Figure 11: Dynamic interaction lag time of different interactive objects

As shown in Figure 11, when the number of interaction objects was 20, GAT had the lowest dynamic interaction latency time, with a value of 7.93s at 3 experimental trials. The dynamic interaction latency times for GNN and scene simulation were 28.34s and 44.73s, respectively, while the multi-view collaborative method had a latency time of 53.69s. When the number of trials was increased to 5, GAT's latency time was 8.32s, while multi-view collaborative had a latency time of 49.71s. The highest latency time for GAT was 13.24s, and the highest for multi-view collaborative was 53.06s. These results show that the GAT-based independent dynamic interaction method achieves the lowest latency across various interaction object counts, demonstrating superior computational efficiency and reasoning ability in complex tasks.

3.3 Performance comparison and analysis of fusion interpretation models

To verify the performance of the fused interpretation model based on intent recognition and independent dynamic interaction, the study selected the following models for comparison: interpretation models based on DNN-HMM, Ensemble empirical mode decomposition (Ensemble Empirical Mode Decomposition, EEMD) and large language model (LLM). The experimental datasets used were the Wmt14 and LibriSpeech datasets. Both are public data sets. The Wmt14 dataset served as the training set, containing parallel corpora for multiple language pairs, such as English-French, English-German, etc. Each sentence pair included source and target language texts. The LibriSpeech dataset served as the test set, containing approximately 1000 hours of 16kHz

English speech corpus. During the preprocessing stage, multi-language texts must undergo unified cleaning to remove special characters and non-linguistic symbols. Cross-language symbols are standardized through predefined dictionaries, including their professional terms and colloquial expressions. In the post-processing of ASR, language models correct word order errors, providing each algorithm with standardized input data to ensure fair performance comparisons. The WMT14 dataset is split into training and validation sets at an 8:2 ratio, while the LibriSpeech dataset is reserved entirely for testing to maintain evaluation integrity. During training, the WMT14 training set is fed into both the fusion interpretation model and the comparison model, with cross-entropy loss function used to optimize parameters. After training, the LibriSpeech test set is input into each model, and multiple rounds of experiments are conducted to calculate the mean and standard deviation, ensuring the reliability of the results. The experimental environment setup is shown in Table 4.

Table 4: Experimental environment and configuration

Hardware configuration	CPU	Intel Xeon Gold 5218 @ 2.30GHz
	GPU	NVIDIA Tesla V100 32GB
	Internal memory	256GB DDR4
	Storage	2TB NVMe SSD
Software environment	Operating system	Ubuntu 20.04 LTS
	Deep learning framework	PyTorch 1.12.0
	Python version	3.8.10
	CUDA version	11.3
	CuDNN version	8.2.1

The study first compared the training and actual loss values of the four models, using the BERT pre-trained language model to encode the input text and generate high-dimensional vector representations. The text sequences were uniformly truncated to a fixed length, and the results are shown in Figure 12.

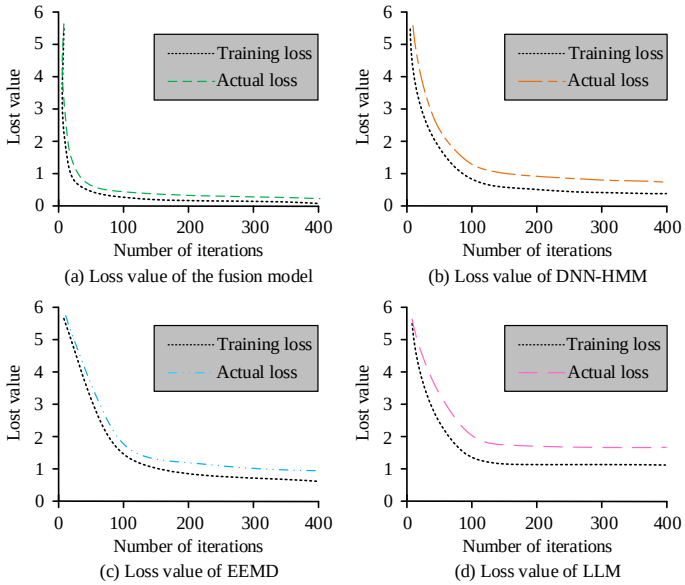


Figure 12: Comparison of loss value results of four models

From Figure 12, it can be seen that the loss values for the study's model were the lowest and most consistent between training and actual losses, with convergence occurring around 30 iterations and stabilizing at approximately 0.35. The DNN-HMM model's actual loss curve was higher than its training loss curve and only converged at 100 iterations. The EEMD model's loss curve converged around 110 iterations, with a stable loss value of around 0.78. The loss curve for the LLM model showed a large discrepancy between the actual and training losses. Based on these results, it can be concluded that the study's model had lower loss values and could better fit the training data, with results more consistent with the true labels. It also exhibited superior generalization and stability compared to the comparison models. The study then compared the interpretation accuracy of the four models on the Wmt14 and LibriSpeech datasets, with results shown in Table 5.

Table 5: Interpretation accuracy results in different datasets

Dataset	Interpretation type	Interpretation accuracy (%)			
		Fusion model	DNN-HMM	EEMD	LLM
Wmt14	Anglo-French	94.37	79.31*	75.09*	64.97*
	Franco-English	96.21	77.59*	71.32*	69.85*
	Anglo-German	90.24	74.69*	70.24*	67.19*
	German-English	91.54	79.18*	72.31*	68.32*
	UK-China	93.36	80.23*	69.58*	69.04*
	China-UK	94.14	80.06*	66.19*	65.42*
LibriSpeech	Anglo-French	96.88	76.43*	69.87*	57.41*
	Franco-English	94.75	79.83*	68.52*	58.39*
	Anglo-German	92.05	76.52*	65.31*	70.35*
	German-English	90.04	77.68*	63.93*	61.23*
	UK-China	93.34	79.53*	69.57*	65.34*
	China-UK	91.72	74.18*	70.25*	58.17*

Note: * indicates that the difference between the proposed model and the comparison model is statistically significant, $p<0.05$.

As shown in Table 5, the fusion model used in the study showed a high interpretation accuracy across different datasets. In the Wmt14 dataset, when the interpretation type was German-English, the interpretation accuracy of the proposed model was the highest at 91.54%, while the interpretation accuracy of the LLM model was 68.32%. In the LibriSpeech dataset, the English-German interpretation accuracy of the proposed model reached 92.05%, while the accuracy of the DNN-HMM, EMMD, and LLM models were 76.52%, 65.31%, and 70.35% respectively, and their interpretation effects were not as good as the proposed fusion model. It could be seen that the proposed model more accurately captured the meaning of the source language and converted it into the target language in the translation task. To ensure the reliability of the results, all models underwent 10 independent experiments for each dataset and interpretation type. The final accuracy was calculated as the average of these 10 experiments, with the standard deviation and 95% confidence interval also calculated. The improved GWO algorithm uses a population size of 30 and a maximum of 100 iterations. The attention heads and hidden layer dimensions were both set to 8. Increasing the GWO population size aims to enhance the fusion model's adaptability in complex contexts. The statistical analysis results for the four models' interpretation accuracy across different datasets are presented in Table 6.

Table 6: Statistical analysis results of the four models on interpretation accuracy in different data sets

Data Set	Interpretation type	Mean ± standard deviation, (confidence interval)			
		proposed model	DNN-HMM	EEMD	LLM
Wmt14 data set	Anglo-French	94.37±0.82(93.91-94.83)	79.31±1.54(78.23-80.39)	75.09±2.11(73.57-76.61)	64.97±2.85(62.73-67.21)
	Franco-English	96.21±0.75(95.78-96.64)	77.59±1.62(76.41-78.77)	71.32±2.25(69.53-73.11)	69.85±2.68(67.72-71.98)
	Anglo-German	90.24±0.91(89.65-90.83)	74.69±1.78(73.42-75.96)	70.24±2.33(68.39-72.09)	67.19±2.92(64.88-69.50)
	German-English	91.54±0.87(91.02-92.06)	79.18±1.59(78.05-80.31)	72.31±2.18(70.65-73.97)	68.32±2.75(66.28-70.36)
	UK-China	93.36±0.84(92.89-93.83)	80.23±1.51(79.17-81.29)	69.58±2.29(67.81-71.35)	69.04±2.81(66.95-71.13)
	China-UK	94.14±0.79(93.68-94.60)	80.06±1.65(78.89-81.23)	66.19±2.42(64.35-68.03)	65.42±2.98(63.07-67.77)
LibriSpeech data set	Anglo-French	96.88±0.72(96.47-97.29)	76.43±1.68(75.19-77.67)	69.87±2.37(68.05-71.69)	57.41±3.12(55.00-59.82)
	Franco-English	94.75±0.81(94.26-95.24)	79.83±1.55(78.70-80.96)	68.52±2.22(66.85-70.19)	58.39±3.05(56.03-60.75)
	Anglo-German	92.05±0.89(91.48-92.62)	76.52±1.72(75.29-77.75)	65.31±2.45(63.43-67.19)	70.35±2.88(68.13-72.57)
	German-English	90.04±0.93(89.45-90.63)	77.68±1.61(76.53-78.83)	63.93±2.38(62.17-65.69)	61.23±2.95(58.97-63.49)
	UK-China	93.34±0.86(92.85-93.83)	79.53±1.53(78.46-80.60)	69.57±2.31(67.82-71.32)	65.34±2.87(63.22-67.46)
	China-UK	91.72±0.83(91.22-92.22)	74.18±1.75(73.01-75.35)	70.25±2.27(68.47-72.03)	58.17±3.15(55.68-60.66)

In Table 6, the proposed model outperforms the other three comparison models in terms of mean and standard deviation across all language pairs and datasets. The mean accuracy for the English-French pair on the Wmt14 dataset is 94.37%, significantly higher than the other models. Meanwhile, the mean accuracies for DNN-HMM and EEMD are 79.31% and 75.09%, respectively, while the mean accuracy for LLM is 64.97%. On the LibriSpeech dataset, the mean accuracy for the German-English pair is 90.04%, which remains significantly higher than the comparison models. These data indicate that the proposed model demonstrates stronger robustness in interpretation accuracy, effectively enhancing translation precision. In order to further verify the performance of the interpretation model in actual application scenarios, the study chose to simulate a large international conference and used simultaneous interpretation equipment, selecting high-quality and low-latency translation headphones, transmitters, and receivers. The data Signal-to-Noise Ratio (SNR) of the four models at different audio lengths was compared by simulating multilingual scenarios, and the results are shown in Figure 13.

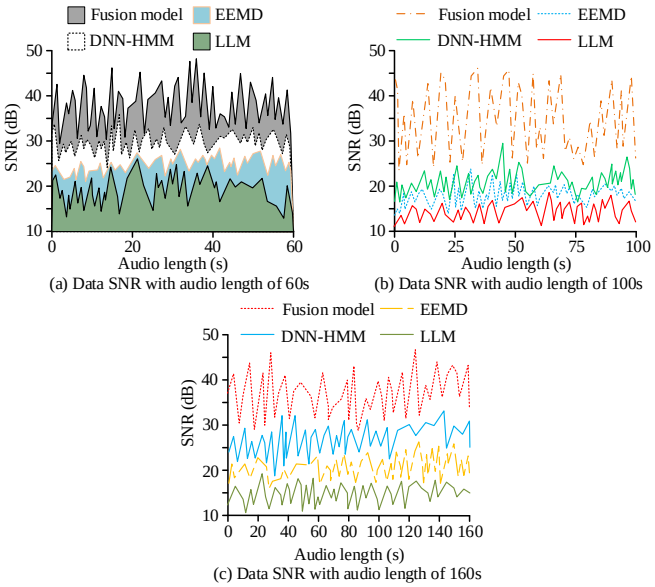


Figure 13: Data SNR results at different audio lengths

From Figure 13, it can be seen that when the audio length was 60s, the highest and lowest SNRs of the proposed model were 47.35dB and 29.74dB respectively. The highest SNR of the DNN-HMM model was 35.81dB, and the lowest data SNR was 24.06dB. The highest SNR of the proposed model was 46.05dB when the audio length was 100s, while the highest data SNR of the LLM model was 19.25dB. When the audio length was 160s, the SNR of the proposed model among the four models was still higher than that of the other three comparison models, and its highest data SNR reached 47.68dB. It could be seen that the training effect of the proposed model was better. The high SNR data enabled the model to converge faster, reduced its training time and resource consumption, and allowed the proposed model to achieve higher translation quality when dealing with complex sentences, professional terms, and cultural background differences. Finally, the study designed ablation experiments on isolated IGWO and GAT, and the results are shown in Table 7.

Table 7: Ablation results

Type of experiment	Compare models	F1 /%	Interpretation delayed /ms
IGWO vs GWO	IGWO	89.2±1.1	123.4±8.3
	GWO	82.5±1.3	156.7±9.2
GAT vs GNN	GAT	91.3±1.2	115.5±7.7
	GCN	85.6±1.4	148.9±8.6
There is cross attention vs no cross attention	Complete pattern	89.2±1.1	123.4±8.3
	Remove cross attention	85.1±1.2	145.6±9.0

In Table 7, in the IGWO vs GWO experiment, the F1 score and interpretation delay of IGWO were both higher than those of GWO, indicating that the improved IGWO algorithm offers better accuracy and efficiency in intent recognition. In the GAT vs GNN experiment, GAT achieved an F1 score of 91.3±1.2% with an interpretation delay of 115.5±7.7ms, demonstrating superior model performance compared to GNN. In the cross-attention vs non-cross-attention experiment, the full model showed a higher F1 score and lower interpretation delay, suggesting that the cross-attention mechanism enhances module collaboration and optimizes the model's

intent-semantic interaction.

4 Discussion

The IGWO intent recognition algorithm demonstrated higher accuracy and precision on the MixSNIPS dataset compared to other models. IGWO's classification accuracy significantly exceeded that of the other three algorithms, all of which scored below 80%. This was mainly because IGWO dynamically adjusted the search weight according to the fitness of the population by introducing a nonlinear weight allocation strategy, thus avoiding the problem that GWO was prone to fall into the local optimum in intent recognition. This mechanism significantly improved the global search ability of the model in complex intent classification tasks, thereby enhancing its classification accuracy. This result was similar to the research results of I. Kim's group [25]. The IGWO algorithm achieved higher recall rates across datasets by employing diversity maintenance strategies, such as random reverse learning, which prevent premature convergence and enable more thorough exploration of the intent feature space. This coincided with the research results of G. Leus et al. [26]. Usability tests show that the proposed GAT-based independent dynamic interaction method improves user experience. This is mainly due to GAT's introduction of an attention mechanism, which dynamically assigns weights to interaction nodes, allowing for a more precise capture of user intent and contextual information. This mechanism significantly reduces interaction interruptions caused by incomplete or misjudged information. Previous studies by Z. He et al. showed that combining operation fusion, multi-level pipelines, and graph partitioning optimization significantly enhances GAT performance [27]. In the comparison of thumb positioning effectiveness among four dynamic interaction methods, the GAT-based independent dynamic interaction method outperformed the other three methods. This is mainly due to GAT's real-time feedback mechanism, which dynamically adjusts interaction strategies based on user behavior. This result is consistent with the findings of A. Coscia et al. [28]. When examining the dynamic interaction latency across different interaction objects, the proposed method showed lower latency. By integrating multimodal data such as visual and tactile information, the GAT method was able to more comprehensively analyze the behavior features of the interaction objects, reducing the latency caused by the insufficiency of a single data source. Meanwhile, Y. Li et al. found in 2023 that the GAT structure design is interpretable, with simulations showing its near-optimal performance and real-time computation support [29]. Based on training and validation loss values, the proposed model demonstrates strong data fitting ability, primarily driven by IGWO's efficient intent recognition and effective resource optimization. This coincided with the results obtained by K. Jang et al. in 2025 [30]. Compared with the research conducted by J. Torres Gómez's team in 2023 [31], the interpretation accuracy of the four models in different data sets showed the highest interpretation accuracy. This improvement results from integrating multimodal data like speech and text, enabling more comprehensive analysis of user intent and context while reducing translation bias from limited data sources. Through the data signal-to-noise ratio results of the four models at different audio lengths, it was found that the training effect of the proposed model was better and had higher signal-to-noise ratio data. This was mainly because the model dynamically adjusted the interaction strategy according to user behavior combined with the real-time feedback mechanism, and further optimized the anti-noise effect. This result was similar to the research conducted by M. Sajid et al. in 2024 [32]. Traditional GWO tends to fall into local optima in complex multilingual environments, while standard GAT struggles with feature differentiation due to high node similarity when processing long multilingual sequences. However, the proposed model demonstrates a higher SNR across various audio lengths compared to the other three models, with its highest SNR reaching 47.68dB. Moreover, the proposed model achieves higher translation quality when handling multilingual differences, accurately capturing and converting the meaning of the source language into the target language in translation tasks. This research considers the potential of multimodal interpretation and plans to incorporate gesture and visual information integration in future work. In the future, it will incorporate gestures and visual information integration. It plans to collect video data of interpretation with gestures and design a multimodal feature extraction module to explore the integration mechanism of visual, audio, and text features. By optimizing dynamic interaction strategies, it aims to adapt to more realistic interpretation scenarios.

In summary, the interpretation model based on IGWO intention recognition and GAT independent dynamic interaction made significant contributions to improving the accuracy of intention recognition, enhancing the ability to capture contextual information, achieving real-time feedback and adaptive adjustment, and promoting the development of interpretation technology. These contributions provided a new theoretical framework and practical methods for interpretation technology, promoted the application and development of artificial intelligence in the field of interpretation, and offered an important reference for future interpretation research and practice.

5 Conclusion

The existing interpretation models suffer from low translation accuracy and weak sensitivity to data bias. To address these limitations, this study proposed a new interpretation model combining the IGWO-based intent recognition algorithm and the GAT-based independent dynamic interaction. By constructing a dynamic graph model to capture the relationship between intent and context, the model enhances intent recognition accuracy and enables a more comprehensive semantic understanding. The results showed that IGWO's classification accuracy was significantly higher than that

of the other three intent recognition algorithms, with HMM having the lowest classification accuracy. The proposed model demonstrated good performance in terms of data SNR, loss values, and dynamic interaction stop counts across different audio lengths. Overall, the proposed integrated model exhibited excellent classification accuracy and translation performance. Its capabilities in speech recognition, language conversion, and contextual understanding enable it to efficiently and accurately complete translation tasks. Although the current model shows excellent classification and translation accuracy, future work will focus on reducing model complexity and enhancing robustness in extreme scenarios to further improve performance.

Fundings

The research is supported by: "14th Five-Year Plan" Teaching Reform Project for Regular Undergraduate Universities in Zhejiang Province, Research on the Digitalized Training Model and it's Effectiveness for English Interpretation Talents, (NO. jg20220589).

References

[1] S. Ni, L. Zhao, A. Li, D. Wu and L. Zhou. "Cross-View Human Intention Recognition for Human-Robot Collaboration," *IEEE WIREL COMMUN*, vol. 30, no. 3, pp. 189-195, June, 2023, DOI: 10.1109/MWC.018.2200514.

[2] Y. Bai, X. Lu and B. Xu. "A Probabilistic Fuzzy Classifier for Motion Intent Recognition," *IEEE T FUZZY SYST*, vol. 32, no. 3, pp. 1098-1107, March, 2024, DOI: 10.1109/TFUZZ.2023.3317938.

[3] G. Jiang, K. Wang, Q. He and P. Xie. "E2FNet: An EEG- and EMG-Based Fusion Network for Hand Motion Intention Recognition," *IEEE SENS J*, vol. 24, no. 22, pp. 38417-38428, Nov, 2024, DOI: 10.1109/JSEN.2024.3471894.

[4] K. Cheng, D. Sun, J. Jian, D. Qin, C. Chen and G. Liao. "Deep Learning Approach for Driver Speed Intention Recognition Based on Naturalistic Driving Data," *IEEE T INTELL TRANSP*, vol. 25, no. 10, pp. 14546-14559, Oct, 2024, DOI: 10.1109/TITS.2024.3398083.

[5] Chauhan S, Saxena S and Daniel P. "Analysis of Neural Machine Translation KANGRI Language by Unsupervised and Semi Supervised Methods," *IETE J RES*, vol. 69, no. 10, pp. 6867-6877, Apr, 2023, DOI: 10.1080/03772063.2021.2016506.

[6] Z. Zuo, P. Wu, X. Sun, X. Tong, R. Guo and H. Huang. "Interactive Feature Extraction Network for Target Intention Recognition," *IEEE SENS J*, vol. 25, no. 3, pp. 4850-4868, Feb, 2025, DOI: 10.1109/JSEN.2024.3488695.

[7] B. Wang. "A Hybrid Fuzzy Logic and Deep Learning Model for Corpus-Based German Language Learning with NLP", *Inform.*, vol. 49, no.21, PP. 1–14, 2025, DOI: 10.31449/inf.v49i21.7423

[8] L. L. Li, G. Cao, H. Liang, Y. Zhang and F. Cui. "Human Lower Limb Motion Intention Recognition for Exoskeletons: A Review," IEEE SENS J, vol. 23, no. 24, pp. 30007-30036, Dec, 2023, DOI: 10.1109/JSEN.2023.3328615.

[9] C. Wong, L. Vergez and W. Suleiman. "Vision- and Tactile-Based Continuous Multimodal Intention and Attention Recognition for Safer Physical Human–Robot Interaction," *IEEE T AUTOM SCI ENG*, vol. 21, no. 3, pp. 3205-3215, July, 2024, DOI: 10.1109/TASE.2023.3276856.

[10] F. Dinarta, A. "Wicaksana. Enhanced Hate Speech Detection in Indonesian-English Code-Mixed Texts Using XLM-RoBERTa", *Inform.*, vol. 49, no.21, PP. 45–56, 2025, DOI: 10.31449/inf.v49i21.7713

[11] P. Sun, H. Xu, J. Yao, Y. Chi, S. Huang and J. Cao. "Dynamic Interaction Analysis and Damping Control Strategy of Hybrid System with Grid-Forming and Grid-Following Control Modes," *IEEE T ENERGY CONVER*, vol. 38, no. 3, pp. 1639-1649, Sept, 2023, DOI: 10.1109/TEC.2023.3249965.

[12] Y. Xiao, L. Wu, J. Guo, J. Li, M. Zhang, T Qin. "A Survey on Non-Autoregressive Generation for Neural Machine Translation and Beyond," *IEEE T PATTERN ANAL*, vol. 45, no. 10, pp. 11407-11427, Oct, 2023, DOI: 10.1109/TPAMI.2023.3277122.

[13] Xiang Y, Chen Y, Ye F H. "Enhancing computer-aided translation system with BiLSTM and convolutional neural network using a knowledge graph approach," *J SUPERCOMPUT*, vol. 80, no. 5, pp. 5847-5869, Oct, 2024, DOI: 10.1007/s11227-023-05686-2.

[14] E. Wang, X. Chen, Y. Li, Z. Fu and J. Huang. "Lower Limb Motion Intent Recognition Based on Sensor Fusion and Fuzzy Multitask Learning," *IEEE T FUZZY SYST*, vol. 32, no. 5, pp. 2903-2914, May, 2024, DOI: 10.1109/TFUZZ.2024.3364382.

[15] H. Zhang, D. Guo, Y. Guo, F. Wu and S. Gao. "A Novel Method for the Driver Lane-Changing Intention Recognition," *IEEE SENS J*, vol. 23, no. 17, pp. 20437-20451, Sept, 2023, DOI: 10.1109/JSEN.2023.3299253.

[16] Y. Zhang, W. Ma, F. Huang, X. Deng and W. Jiang. "A Novel Air Target Intention Recognition Method Based on Sample Reweighting and Attention-Bi-GRU," *IEEE SYST J*, vol. 18, no. 1, pp. 501-504, March, 2024, DOI: 10.1109/JSYST.2023.3319643.

[17] M. Yi, W. Lee and S. Hwang. "A Human Activity Recognition Method Based on Lightweight Feature Extraction Combined with Pruned and Quantized CNN for Wearable Device," *IEEE T CONSUM ELECTR*, vol. 69, no. 3, pp. 657-670, Aug, 2023, DOI: 10.1109/TCE.2023.3266506.

[18] D. Roldán-Álvarez and F. Mesa. "Intelligent Deep-Learning Tutoring System to Assist Instructors in Programming Courses," *IEEE T EDUC*, vol. 67, no. 1, pp. 153-161, Feb, 2024, DOI: 10.1109/TE.2023.3331055.

[19] X. Fu, M. Huang, C. K. Tse, J. Yang, Y. Ling and X.

Zha. "Synchronization Stability of Grid-Following VSC Considering Interactions of Inner Current Loop and Parallel-Connected Converters," *IEEE T SMART GRID*, vol. 14, no. 6, pp. 4230-4241, Nov, 2023, DOI: 10.1109/TSG.2023.3262756.

[20] J. Liang, Y. Lu, F. Wang, G. Yin, X. Zhu and Y. Li. "A Robust Dynamic Game-Based Control Framework for Integrated Torque Vectoring and Active Front-Wheel Steering System," *IEEE T INTELL TRANSP*, vol. 24, no. 7, pp. 7328-7341, July, 2023, DOI: 10.1109/TITS.2023.3262655.

[21] E. Ghiorzi, M. Colledanchise, G. Piquet, S. Bernagozzi, A. Tacchella and L. Natale. "Learning Linear Temporal Properties for Autonomous Robotic Systems," *IEEE ROBOT AUTOM LET*, vol. 8, no. 5, pp. 2930-2937, May, 2023, DOI: 10.1109/LRA.2023.3263368.

[22] I. Stepin, M. Suffian, A. Catala and J. M. Alonso-Moral. "How to Build Self-Explaining Fuzzy Systems: From Interpretability to Explainability [AI-eXplained]," *IEEE COMPUT INTELL M*, vol. 19, no. 1, pp. 81-82, Feb, 2024, DOI: 10.1109/MCI.2023.3328098.

[23] Y. Zhang, Y. Zou, Selpi, Y. Zhang and L. Wu. "Spatiotemporal Interaction Pattern Recognition and Risk Evolution Analysis During Lane Changes," *IEEE T INTELL TRANSP*, vol. 24, no. 6, pp. 6663-6673, June, 2023, DOI: 10.1109/TITS.2022.3233809.

[24] S. Saeidian, G. Cervia, T. J. Oechtering and M. Skoglund. "Pointwise Maximal Leakage," *IEEE T INFORM THEORY*, vol. 69, no. 12, pp. 8054-8080, Dec, 2023, DOI: 10.1109/TIT.2023.3304378.

[25] I. Kim, J. Na, J. P. Yun and S. Lee. "Deep Feature Selection Framework for Quality Prediction in Injection Molding Process," *IEEE T IND INFORM*, vol. 20, no. 1, pp. 503-512, Jan, 2024, DOI: 10.1109/TII.2023.3268421.

[26] G. Leus, A. G. Marques, J. M. F. Moura, A. Ortega and D. I. Shuman. "Graph Signal Processing: History, development, impact, and outlook," *IEEE SIGNAL PROC MAG*, vol. 40, no. 4, pp. 49-60, June, 2023, DOI: 10.1109/MSP.2023.3262906.

[27] Z. He, T. Tian, Q. Wu and X. Jin. "FTW-GAT: An FPGA-Based Accelerator for Graph Attention Networks with Ternary Weights," *IEEE T CIRCUITS-II*, vol. 70, no. 11, pp. 4211-4215, Nov, 2023, DOI: 10.1109/TCSII.2023.3280180.

[28] A. Coscia and A. Endert. "KnowledgeVIS: Interpreting Language Models by Comparing Fill-in-the-Blank Prompts," *IEEE T VIS COMPUT GR*, vol. 30, no. 9, pp. 6520-6532, Sept, 2024, DOI: 10.1109/TVCG.2023.3346713.

[29] Y. Li, Y. Lu, R. Zhang, B. Ai and Z. Zhong. "Deep Learning for Energy Efficient Beamforming in MU-MISO Networks: A GAT-Based Approach," *IEEE WIREL COMMUN LE*, vol. 12, no. 7, pp. 1264-1268, July, 2023, DOI: 10.1109/LWC.2023.3270361.

[30] K. Jang, K. E. S. Pilario, N. Lee, I. Moon and J. Na. "Explainable Artificial Intelligence for Fault Diagnosis of Industrial Processes," *IEEE T IND INFORM*, vol. 21, no. 1, pp. 4-11, Jan, 2025, DOI: 10.1109/TII.2023.3240601.

[31] Liu X, Chen J, Zhang Q T. "Exploration of low-resource language-oriented machine translation system of genetic algorithm-optimized hyper-task network under cloud platform technology," *J SUPERCOMPUT*, vol. 80, no. 3, pp. 3310-3333, Sep, 2024, doi: 10.1007/s11227-023-05604-6.

[32] M. Sajid, A. K. Malik, M. Tanveer and P. N. Suganthan. "Neuro-Fuzzy Random Vector Functional Link Neural Network for Classification and Regression Problems," *IEEE T FUZZY SYST*, vol. 32, no. 5, pp. 2738-2749, May, 2024, DOI: 10.1109/TFUZZ.2024.3359652.

Predicting E-commerce Customer Purchase Behavior Using LSTM-Attention Neural Networks and Data Optimization Strategies

Baoyin Ye
School of Business, Zhengzhou Shengda University, Zhengzhou 451191, Henan, China
E-mail: Baoyin_Ye@outlook.com

Keywords: electronic commerce, neural network, customer buying behavior prediction, LSTM+Attention, data preprocessing

Received: April 11, 2025

In this study, based on neural network algorithm, a prediction model of e-commerce customer purchase behavior was constructed, and the data pre-processing, training strategy and model selection were optimized to improve the prediction accuracy and generalization ability. Firstly, data preprocessing methods such as missing value filling, normalization, Thermal coding and PCA dimensionality reduction are used to effectively optimize data quality and improve the learning ability of the model. Secondly, in terms of training strategy, dynamic learning rate adjustment, gradient clipping, batch training optimization and stop in advance are introduced to reduce overfitting risk and improve training stability. The study compared the performance of logistic regression, random forest, LSTM and LSTM+Attention models. LSTM+Attention model was evaluated using a 12-month e-commerce dataset comprising 1,058 user records, incorporating features such as user demographics, historical purchase behavior, browsing data, shopping cart actions, product metadata, and promotional interactions.The input features were preprocessed using normalization, one-hot encoding, and PCA, and models were trained with optimized hyperparameters using stratified sampling and cross-validation. The study compared four models—logistic regression, random forest, LSTM, and LSTM+Attention.LSTM+Attention achieved the highest performance with 91.3% accuracy and AUC-ROC of 0.95, outperforming logistic regression (76.8%), random forest (81.5%), and LSTM alone (88.9%).These results demonstrate the effectiveness of combining temporal modeling with attention mechanisms in predicting e-commerce customer purchase behavior. This paper analyzes the value of predictive model in business applications such as personalized recommendation, precision marketing, dynamic pricing, inventory management, etc., and shows that the model can effectively improve the operational efficiency and user experience of e-commerce platforms. Through literature review, this study further discusses the development trend of e-commerce prediction model, which provides theoretical support and practical guidance for the intelligent decision-making of e-commerce platform.

Povzetek: Raziskava predstavi LSTM+Attention model za napoved nakupovalnega vedenja e-trgovinskih uporabnikov. Z izboljšano obdelavo podatkov in strategijami treniranja doseže višjo točnost, podpira personalizirana priporočila, dinamično cenovno prilagajanje ter učinkovito trženje.

1 Introduction

The rapid development of e-commerce makes consumer online shopping behavior data show exponential growth. Traditional forecasting methods mainly rely on statistics and machine learning technologies, such as logistic regression, decision tree and support vector machine, etc. These methods perform well when dealing with structured data. However, for large-scale, high-dimensional and non-linear data in the e-commerce environment, the data of consumers' online shopping behavior shows exponential growth. Its predictive ability has obvious limitations. In the face of complex variables such as user browsing behavior, interaction record, product characteristics and social recommendation, the traditional model is difficult to accurately depict the potential purchase intention of consumers, which leads to the decline of prediction accuracy and affects the effectiveness of marketing decisions. Driven by deep learning, neural network algorithms have made achievements in the fields of image recognition, natural language processing and recommendation systems. The powerful feature extraction and nonlinear mapping capabilities make it show great potential in customer behavior prediction. Compared with traditional methods, neural network models can automatically learn complex patterns in data, capture long-term dependencies, and dig deep nonlinear features. At present, the application of neural network in e-commerce field still faces challenges, such as high model complexity, high computational cost, data noise interference and insufficient feature engineering optimization. This study builds a customer purchase behavior prediction model based on neural network, optimizes data pre-processing and feature engineering,

and improves model architecture and training strategies to improve prediction accuracy and provide support for precision marketing and intelligent recommendation of e-commerce platforms.

Prediction models have been widely used in the field of e-commerce. Based on different data analysis methods and algorithm models, researchers explore the prediction strategies of e-commerce customers' purchase behavior, and evaluate the effectiveness of different models.

In terms of theoretical research on prediction models, Magnus et al. proposed the weighted least squares prediction method, emphasizing the role of data weights in improving prediction accuracy [1]. Chattoe-Brown explored the potential of intelligent agent-based modeling in predictive research and points out that ABM can be used to simulate the behavior patterns of different types of consumers on e-commerce platforms to optimize marketing strategies [2]. Wu et al. proposed a proxy model method to select the optimal traffic conflict prediction model, which can also be used to optimize anomaly detection and consumer behavior prediction in e-commerce transactions [3]. Li et al. studied the forecast of the user growth of fixed telephone and mobile communication in Japan in terms of specific e-commerce prediction modeling, and derived the user growth curve based on the nonlinear prediction model, providing a reference for the user growth forecast of e-commerce platforms [4]. Svabova et al. proposed a machine learning model based on financial data for the bankruptcy prediction of Slovak smes, and the study showed that the prediction model can effectively identify risk points in business operations [5]. Yang et al. proposed a combined forecasting model based on residual estimation, which can be applied to the data trend analysis of e-commerce platforms to improve the forecasting ability of future sales and market fluctuations [6]. Von Eye et al. discussed the predictive analysis model in longitudinal research in terms of time series prediction and model optimization, and emphasized the importance of time series data for long-term trend prediction, which provided theoretical support for the long-term trend prediction of e-commerce user behavior [7]. Zhang et al. found through field experiments that customer participation driven by mobile applications can significantly enhance consumers' purchasing behavior in baking stores, indicating that digital tools play a positive role in promoting offline consumption [8]. Seinen et al. studied the predictive model in the Dutch medical field and emphasized the role of unstructured text data (such as user comments and customer service records) in improving the predictive ability of the model. This idea can be applied to the review analysis of e-commerce platforms to optimize the recommendation system [9].

Zhang et al. proposed a route link prediction model based on particle swarm optimization, which improved the prediction accuracy by optimizing hyperparameters, and this idea is also applicable to user behavior prediction in e-commerce [10]. Yuan et al. studied the prediction model of grouped data and analyzed the impact of data grouping on prediction accuracy, which can be applied to the behavior modeling of different consumer groups on e-commerce platforms [11]. Michalkova et al. studied the early warning mechanism of corporate financial distress based on the bankruptcy prediction model, and this research method can be extended to the analysis of business conditions of e-commerce enterprises [12]. Li Bin et al. demonstrated that when an AI customer service system experiences service failure, customer engagement plays a mediating role in its influence on purchasing behavior, highlighting the significance of intelligent customer service design in enhancing user experience [13]. In terms of model optimization and evaluation methods, Zhang and Liu proposed a model average prediction method based on K-fold cross-validation, which plays an important role in improving the stability of neural network models and can be used to optimize the user behavior prediction model of e-commerce platforms [14]. Gavurova et al. evaluated the applicability of different bankruptcy prediction models and proposed model selection strategies in different business environments. This study has reference significance for the risk management and credit evaluation of e-commerce platforms [15]. Chiou analyzed the influence of different thinking models on prediction in the field of physics education, and this study emphasized the source analysis of prediction errors, which can provide references for error optimization of e-commerce prediction models [16]. Mohammadi studied the prediction ability of α-stabilized GARCH and ARMA-GARCH-M models and discussed the risk prediction methods of financial markets, which has certain guiding significance for the market dynamic analysis of e-commerce platforms [17].

The rapid development of e-commerce research has also prompted scholars to pay attention to its impact on the economy and society. Kumar et al. reviewed the evolution of e-commerce research in the past 20 years and analyzed the application and development trend of different forecasting models in the field of e-commerce [18]. Wang studied the impact of the development of e-commerce on the urban-rural income gap, and discussed the intermediary role of technological innovation in this process. This study emphasized the importance of technological upgrading of e-commerce platforms on the prediction of consumer behavior [19]. Soleimani conducted a systematic literature review on buyer trust on e-commerce platforms, and pointed out that a trusted transaction environment can enhance users' stickiness to the platform and improve the reliability of customer purchase behavior prediction [20]. Pejic-Bach discussed the development and changes of e-commerce during the COVID-19 epidemic, analyzed the forecasting methods of user consumption patterns, and provided empirical reference for e-commerce platforms to cope with market fluctuations [21].

In terms of user behavior modeling and trust mechanism research, Peng and Yang studied the impact of e-service quality on consumer trust, and proposed the strategy of using advanced technologies to enhance customer loyalty, which provided a reference for e-commerce platforms to optimize user experience [22]. Zhu et al. analyzed the consistency of product information in cross-border e-commerce and proposed the

applicability evaluation criteria of prediction models under different market backgrounds [23]. Deng et al. proposed an e-commerce volatility prediction method based on joint BW test, which has important guiding significance for e-commerce market trend analysis and dynamic pricing strategy optimization [24]. The existing literature has deeply discussed the prediction of e-commerce customer purchase behavior, covering different modeling strategies from traditional statistical methods to deep learning, and evaluating the optimization effect of the prediction model combined with different application scenarios. On the basis of absorbing existing research results, this study combined with LSTM+Attention neural network model to further optimize the accuracy and stability of e-commerce user behavior prediction, providing theoretical support and practical guidance for the intelligent decision-making of e-commerce platform.

Current research on e-commerce customer purchase behavior prediction mainly focuses on traditional machine learning methods and deep learning methods. Traditional methods such as logistic regression, decision trees, support vector machines and random forests mainly rely on manual feature engineering, modeling through variables such as user history behavior and product characteristics. These methods perform well on small-scale data sets, but are difficult to deal with high-dimensional, non-linear and dynamically changing data in e-commerce environment. Because of its strong feature learning ability, deep learning model is gradually applied to customer behavior prediction. The LSTM-based model can capture the time series characteristics of user purchase behavior, while the attention mechanism can enhance the weight of key behaviors and improve the prediction accuracy. At present, there are still some problems in the research, such as insufficient generalization ability of model, large influence of data noise and high cost of real-time calculation. Many studies still rely on rules of thumb in data preprocessing and feature engineering optimization, and lack of system optimization for e-commerce scenarios.

In this study, the deep learning method is used to build a prediction model of e-commerce customer purchase behavior based on neural network algorithm. The data processing stage combines data cleaning, feature extraction and dimensionality reduction techniques to optimize data quality. In terms of model construction, recurrent neural network and long term memory network will be selected to capture the time-dependent characteristics of user purchase behavior, and attention mechanism will be combined to enhance the weight of key behavior features to improve the prediction accuracy. In order to optimize the model training, hyperparameter optimization and regularization techniques will be used to improve the generalization ability of the model, and cross-validation will be used to evaluate the stability of the model. The core value of this study is to break through the limitations of traditional methods, build a more accurate, more efficient and more interpretable neural network prediction model, and provide data-driven decision support such as intelligent marketing and accurate recommendation for e-commerce platforms.

This work aims to improve the predictive accuracy and commercial utility of customer behavior models in e-commerce by integrating temporal modeling (LSTM) with an attention mechanism and optimized training strategies.

2 Materials and methods

2.1 Data collection and sample selection

2.1.1 Data source

The data collected in this study mainly comes from the transaction behavior data of users on e-commerce platforms, including historical purchase records, browsing behaviors, shopping cart operations, basic information of users and commodity characteristics. The data acquisition methods include the open data set of the platform, the anonymous data provided by the cooperation of enterprises, and the user interaction data captured by the web crawler technology under the premise of data ethics. In order to ensure the representativeness and reliability of the data, this study selected data from an e-commerce platform covering multiple categories of goods over a period of 12 consecutive months to capture the cyclical changes in consumer behavior. Abnormal data, such as severely missing samples and abnormal transactions, will be eliminated, and normalized numerical variables will be processed to reduce the impact of differences in different feature units. In order to improve the training effect of the model, this study adopts the strategy of combining random sampling and stratified sampling to ensure the balanced distribution of different user groups in the data set, so as to reduce data bias. These data will serve as the basis for model training and verification, and provide reliable data support for subsequent predictive analysis, as shown in Table 1.

Table 1: Data sources for the study

Data Type	Data Source	Specific Content	Purpose
User Information	E-commerce platform user database	User ID, age, gender, membership level, region	Identify user characteristics and differentiate consumer groups
Purchase Behavior	Transaction logs, order datasets	Order ID, product ID, purchase time, purchase quantity, payment method	Record user purchase behavior and analyze buying patterns
Browsing Behavior	Website clickstream logs, cookies	Viewed products, dwell time, visit path, click count	Predict user interest and purchasing intent
Shopping Cart Data	E-commerce	Time added to cart, product	Analyze user decision-making

	database	quantity, removed items	process and potential purchasing intent
Product Features	Product catalog, product details	Category, brand, price, discount, inventory status	Optimize recommendations based on user behavior
Promotions & Marketing	Platform promotional records, ad data	Promotion type, discount level, ad click rate	Evaluate the impact of promotions on purchasing behavior
Time Series Data	Transaction timestamps, consumption trends	Peak purchasing periods, seasonal trends	Analyze consumption seasonality to improve prediction accuracy
Social Interaction Data	User reviews, ratings, sharing behavior	Review sentiment, rating distribution, product recommendations	Assess the influence of social factors on purchasing behavior

2.1.2 Sample screening and data preprocessing

In the process of sample selection, this study focuses on the representativeness, integrity and balance of data to ensure that the neural network model can accurately predict the purchasing behavior of e-commerce customers. Select active users with a long history of purchases to capture trends in consumer behavior, while weeding out unusual users, such as accounts with malicious swiping, a large number of returns, or no purchases for a long time, to ensure data quality. In order to enhance the applicability of the model, it is ensured that the samples cover user groups with different consumption levels, purchase frequency and category preferences, so that the model can learn diversified purchasing behavior patterns. In order to avoid the prediction bias caused by uneven data distribution, this study adopts the method of combining random sampling and stratified sampling to balance the distribution of different types of users and ensure that the training data is consistent with the actual e-commerce environment. In terms of sample time span, 12 consecutive months of user behavior data are selected to capture seasonal and cyclical consumption characteristics, and the impact of short-term purchase data on the stability of the model is eliminated. The final selected sample data will be used for model training and verification to provide reliable data support for e-commerce customer purchase behavior prediction, as shown in Table 2.

Table 2: Sample data

User ID	Age	Gender	Membership Level	Purchase Count	Browse Count	Cart Add Count	Product Category	Total Order Amount (CNY)	Avg Purchase Interval (Days)
1	28	Female	Diamond Member	27	177	13	Food	1378.36	13.8
2	41	Female	Platinum Member	17	57	7	Food	921.77	15.6
3	36	Male	Regular Member	9	153	3	Daily Necessities	2546.4	14.6
4	32	Male	Regular Member	19	133	7	Food	3893.37	13.1
5	29	Female	Regular Member	11	129	7	Electronics	1822.81	21.7
6	42	Male	Gold Member	23	161	3	Clothing	717.3	21.9
7	28	Male	Diamond Member	13	177	9	Clothing	1687.91	12.1
8	47	Female	Platinum Member	9	193	9	Food	2271.21	17
...	...	...	...	...	...	...	...	...	...
1058	35	Male	Regular Member	19	141	11	Clothing	3521.45	14.8

Table 2 presents only a partial snapshot of the full dataset for illustrative purposes. The actual training dataset contains 1,058 user records spanning 12 months, as described above.

Selecting a 12-month continuous dataset allows the model

to learn both short-term fluctuations (e.g., promotional spikes, seasonal sales) and longer-term behavioral trends (e.g., repeat purchases, category shifts). Rather than eliminating short-term effects, this approach integrates them into a broader temporal context, reducing the risk of overfitting to transient anomalies from narrow time windows (e.g., single-week flash sales). It ensures the training data reflects real-world consumer variability across different seasons, event cycles, and behavioral rhythms. This strategy improves model robustness by exposing it to a complete annual behavioral arc, enabling better generalization across future usage scenarios.

The dataset consists of 1,058 user records collected over 12 consecutive months from a major e-commerce platform, covering eight product categories and 11 distinct behavioral and demographic features per user. The purchase prediction task was formulated as a binary classification problem, where 0 indicates no purchase and 1 indicates purchase. Users were split into 70% training (740 samples), 15% validation (159), and 15% test sets (159) using stratified sampling by purchase frequency and membership tier. Demographically, the sample includes 52.3% female and 47.7% male users, with an age range of 18–57.The average number of purchases per user is 15.2, with a right-skewed distribution showing 20% of users accounting for over 60% of purchases. Product categories are unevenly distributed, with food (31.7%), clothing (22.5%), and electronics (18.9%) most represented. The overall interaction matrix exhibits 83.4% sparsity, highlighting the challenge of implicit feedback prediction.

2.1.3 Data feature engineering

Data feature engineering is an important part of optimizing e-commerce customer purchase behavior prediction model, which mainly includes feature selection, feature transformation and feature construction. In terms of feature selection, variables such as user basic information, purchase history, browsing behavior, shopping cart operation, and product characteristics are extracted to ensure that the data can accurately reflect consumers' purchase patterns. In the process of feature conversion, the numerical variables are normalized to ensure that the scales of different features are consistent, and the categorical variables are converted by means of Thermal coding or embedded representation to improve the learning ability of the neural network. In the feature construction stage, advanced features such as user purchase frequency, consumption trend and commodity similarity are constructed, and combined with time series information, the model's ability to identify long-term and short-term behaviors is enhanced. Principal component analysis, information gain or L1 regularization are used for feature screening to remove redundant features and improve the computational efficiency and generalization ability of the model. Finally, the optimized feature set will be used as the input of the neural network model to improve the accuracy and stability of e-commerce customer purchase behavior prediction.

2.2 Construction of prediction model

2.2.1 Selection formula of neural network model

Based on the time dependence, feature complexity and computational cost of the data, this study considers the use of recurrent neural networks, long term memory networks or attention mechanism-based Transformer model. Model selection is measured according to the following formula:

The model loss function uses mean square error or cross entropy loss to measure the prediction error.

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{1}$$

or

$$L = \frac{1}{N}\sum_{i=1}^{N}[y_i \log \hat{y}_i + (1-y_i) log(1-\hat{y}_i)] \tag{2}$$

y_i is the real purchase behavior, $\hat{y}_i$ is the predicted value of the model, and N is the total number of samples.

The model complexity evaluation is to measure the computational cost by the number of parameters P to avoid overfitting.

$$P = \sum_{l=1}^{L}(n_{l-1} \times n_l + n_l) \tag{3}$$

L is the number of layers of the network, and n_l is the number of neurons in layer l.

The optimization goal is to select the model that meets the following optimal conditions.

$$arg_M^{min}(L + \lambda P) \tag{4}$$

M is the candidate model and λ is the regularization parameter that weighs loss and computational complexity.

In this study, after the experimental comparison of different models, the model with minimal loss and moderate computational complexity was selected for subsequent optimization.

Based on the comparative results across all candidate models, LSTM+Attention demonstrated the lowest loss (0.184), highest AUC (0.95), and balanced training time, thereby fulfilling both accuracy and efficiency criteria. Accordingly, it was selected as the final model architecture for subsequent training and evaluation.

2.2.2 Model architecture design

(1) Input layer

The input layer is the first step of the neural network model to process the e-commerce customer purchase behavior prediction task, and its main function is to receive and normalize the input data, so as to provide high-quality data basis for the feature extraction and learning of the subsequent hidden layer. The input data of this study includes multi-dimensional information such as user purchase history, browsing behavior, shopping cart operation and product characteristics. In order to improve the expressiveness of the model, the input layer adopts a

multi-channel structure to process different types of data separately. User behavior data is input in the form of time series, and time step data is constructed by sliding window method with fixed window length. Product features are represented by embedded vectors to capture the similarities between different product attributes. In order to improve the training stability of the model, the input layer normalizes the numerical features so that they are distributed in the same numerical range to reduce the deviation of gradient update. For category-type variables, such as user identity and commodity category, embedding layer is used to map to low-dimensional vector space to reduce the impact of dimensional disaster on model training. The input layer splices different types of data and inputs them into the hidden layer for deep feature extraction to provide high-quality input data for e-commerce customer purchase behavior prediction.

(2) Hidden layer

The hidden layer is a key part of the deep feature extraction and pattern learning of the neural network model for e-commerce customer purchasing behavior. In this paper, according to the characteristics of e-commerce data, multi-layer neural network architecture is used to improve the expressiveness and prediction performance of the model. In view of the time dependence of user behavior data, the hidden layer introduces a recurrent neural network structure, in which the long-duration memory network can effectively capture the long-term dependence characteristics of user purchase behavior, avoiding the problem of gradient disappearance existing in traditional RNN. In order to enhance the attention mechanism of the model and enhance the attention to key behavioral features, this study superimposes the self-attention mechanism on the hidden layer, and calculates the correlation weights between the data of different time steps, so that the model can automatically screen the behavioral features that have the greatest impact on the final purchase decision.

To enhance architectural novelty, a hybrid model was introduced by integrating multi-head attention and hierarchical attention into the LSTM framework. The multi-head attention mechanism enables the model to learn from diverse representation subspaces, enhancing sensitivity to varying behavioral signals. Hierarchical attention captures both intra-session and inter-session dependencies, improving sequence-level interpretability. Additionally, temporal convolutional layers were incorporated before the recurrent structure to extract short-term purchase patterns efficiently. The revised model architecture was visualized with a detailed flow diagram illustrating input embedding, convolutional layers, attention weight distribution, LSTM encoding, and output layers, clarifying the interaction among key components and improving transparency in design and implementation. As shown in Figure 1:

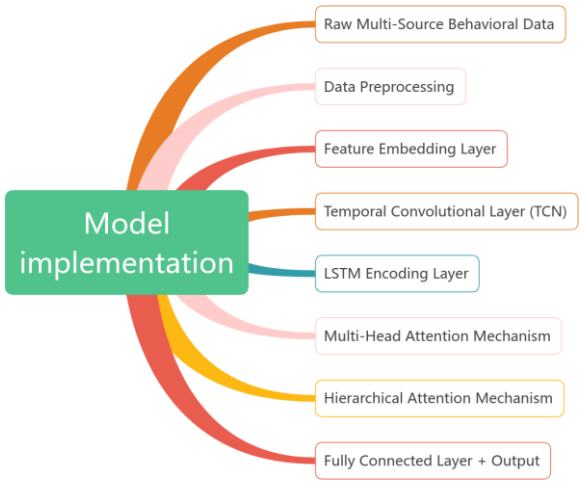


Figure 1: Flowchart of model implementation

The number of neurons in each layer is determined by experimental tuning, the ReLU activation function is used to enhance the nonlinear expression ability, and batch normalization is introduced to speed up the training speed and improve the model stability. In order to prevent the model from overfitting, Dropout technology is used in the hidden layer to randomly drop some neurons with a certain probability to enhance the generalization ability. In order to optimize the feature representation, this study uses the fully connected layer to reduce the dimension of LSTM output information, reduce parameter redundancy, and improve computing efficiency. After learning from multiple hidden layers, the model can extract the deep pattern of user purchasing behavior and provide effective support for the final purchase prediction.

(3) Output layer

The output layer is the final link of the neural network model to predict the purchasing behavior of e-commerce customers, and its function is to convert the features extracted by the hidden layer into specific prediction results. This study uses the fully connected layer as the output layer to ensure that the depth features learned by the hidden layer can be mapped to the final purchase probability or category label. The structure of the output layer varies according to different prediction tasks. If it is a binary task (to buy or not to buy), the Sigmoid activation function is used to normalize the output value to between 0 and 1, and 0.5 is used as the threshold for the purchase decision, i.e

$$\hat{y} = \sigma(Wx + b) = \frac{1}{1 + e^{-(Wx+b)}} \tag{5}$$

W and b are weights and biases respectively, x is the output feature vector of the hidden layer, and $\hat{y}$ represents the purchase probability of the user. If it is a multi-classification task (predicting the category of goods the user will buy), the Softmax activation function is used to map multiple neurons in the output layer into a probability distribution so that the probability sum of all categories is 1, i.e

$$y\hat{y}_i = \frac{e^{zi}}{\sum_j e^{zj}} \quad (6)$$

$\hat{y}_i$ represents the probability of the user choosing the item of category i, and e^{zi} is the score corresponding to that category.

In order to optimize the prediction ability of the model, this study adopted the cross-entropy loss function in the output layer, and used the Adam optimization algorithm to adjust the weight parameters to improve the training efficiency and prediction accuracy of the model. The predictive results of the output layer will be used to personalize recommendations and precision marketing for e-commerce platforms to improve user experience and sales conversion rates.

2.2.3 Training parameter optimization and hyperparameter tuning table

Training parameter optimization and hyperparameter tuning are the key steps to improve the performance of neural network models, which directly affect the accuracy and generalization ability of e-commerce customer purchase behavior prediction. In this study, the key hyperparameters are optimized by the combination of grid search and Bayesian optimization. In terms of optimizer selection, Adam optimization algorithm is used to balance convergence speed and model stability. The learning rate is an important parameter that affects the convergence speed and final accuracy of the model. The initial setting of the experiment is 0.001, and the exponential attenuation strategy is used in the training process to avoid the model oscillating in the later stage. The batch size determines the number of samples used for each weight update, with values ranging from 32 to 256, taking into account computational efficiency and gradient stability. The number of neurons in the hidden layer, the number of layers and the selection of activation function also affect the expression ability of the model. In this study, experiments were carried out between 2 and 4 layers, with ReLU as the main activation function and Batch Normalization to improve the training stability. The regularization method combines L2 regularization and Dropout to suppress overfitting, and sets different Dropout probabilities in the process of hyperparameter tuning. The optimal combination of hyperparameters was determined by cross-validation, which ensured the prediction accuracy of the model and improved the generalization ability and computational efficiency.

The final model configuration was derived using a grid search strategy over defined hyperparameter ranges, evaluated through 5-fold cross-validation. LSTM units were tuned in [64, 128, 256] and 128 yielded the best trade-off between accuracy and training time. Multi-head attention used 4 heads from a search range of [2, 4, 8], balancing attention granularity and computational overhead. Dropout rate was optimized at 0.3 from a range [0.2, 0.3, 0.5], reducing overfitting without harming convergence. Batch size was set to 128 after evaluating [32, 64, 128, 256], where it achieved 91.3% validation accuracy and stable gradient behavior. Learning rate was initialized at 0.001 with exponential decay, improving early convergence. These choices collectively enhanced training stability, generalization, and interpretability.

2.2.4 Model implementation and training process

This research is based on the deep learning framework TensorFlow/Keras to realize the e-commerce customer purchase behavior prediction model, and is trained in the high-performance computing environment. After the data preprocessing was completed, the batch data input was constructed using DataLoader, and the data was divided into training sets, verification sets and test sets, with a ratio of 70:15:15. During model initialization, He initialization method was used to ensure reasonable weight distribution and load the optimized hyperparameters. In the training process, the loss function adopts binary cross entropy or class cross entropy to measure the model prediction error. Adam optimizer is used for gradient updating, and learning rate scheduling strategy is combined to decrease the learning rate rapidly in the initial stage and gradually reduce the learning rate when it converges in the later stage to improve the model stability. The training adopts the Mini-Batch mode, the verification set loss and accuracy are calculated after each Epoch, the overfitting in the training process is monitored, and the training is terminated in advance when the verification loss is no longer reduced through the Early Stopping mechanism. After the training was completed, the generalization performance of the final model was evaluated using the test set, and the prediction effect of the model was analyzed by the confusion matrix, OC-ROC curve and other indicators. The trained model is exported and deployed online in inference environment to realize real-time user behavior prediction of e-commerce platform.

2.3 Training and verification

2.3.1 Partitioning strategy of training set and test set

To ensure temporal consistency in sequential prediction tasks, the validation set was restructured using time-based splitting, positioned chronologically between the training and test sets. This adjustment aligns the data partitioning with real-world inference scenarios, where models are evaluated on unseen future data. The validation set now reflects post-training but pre-testing periods, allowing for hyperparameter tuning without data leakage. replacing "Random Sampling" with "Intermediate Time-based Splitting" for the validation set. This change enhances model reliability by preserving the causal sequence of user behavior and avoiding artificially inflated performance.

This study combines stratified sampling and time series partitioning to ensure the balanced distribution of data, while preserving the time dependence of user behavior. Stratified sampling is carried out according to the characteristics of users' purchase frequency and

membership level to ensure the balanced distribution of data among different types of users and avoid bias in the training of the model by certain types of users. The time series division method is adopted to make the data time range of the training set earlier than the test set to simulate the real business scenario, ensuring that the model can learn the historical behavior pattern of users and make future predictions. This study finally divides the data according to the proportion of 70% training set, 15% verification set and 15% test set, and conducts balanced sampling of new users, high frequency users and low frequency users to enhance the generalization ability of the model. The validation set is used to adjust hyperparameters and prevent overfitting, while the test set is used to assess the predictive power of the final model to ensure its applicability in a real e-commerce environment. Such partitioning strategy ensures the robustness of the model on new data, improves the reliability of the prediction results, and provides more scientific support for the user behavior prediction of e-commerce platform.

2.3.2 Training process and parameter tuning

In the training process of e-commerce customer purchase behavior prediction model, it is very important to divide the training set and the test set reasonably for the generalization ability of the model. In this study, stratified sampling and time series division are used to ensure the balance of data distribution, while retaining the time dependence of user purchase behavior. Stratified sampling is carried out according to the user's activity, purchase frequency and other characteristics to ensure that the behavioral data of different types of users are evenly distributed in the training set and the test set. Considering that e-commerce data has the characteristics of time series, the model needs to learn the changing trend of user purchasing behavior, so it should be divided in time order to ensure that the time range of data in the training set is earlier than that in the test set, so as to actually predict the scene. In the end, this study divided the data according to the proportion of 70% training set, 15% verification set and 15% test set, and conducted balanced sampling for different user groups (new users, high-frequency users and low-frequency users) to improve the generalization ability and prediction accuracy of the model, as shown in Table 3.

Table 3: Dataset splitting for training and testing

Dataset Type	Sample Size	Proportion	Splitting Method	Primary Purpose
Training Set	740	70%	Stratified Sampling + Time-based Splitting	Train the model to learn user behavior patterns
Validation Set	159	15%	Random Sampling	Optimize hyperparameters and prevent overfitting
Test Set	159	15%	Future-based Splitting	Evaluate model generalization performance

The data partitioning strategy in this study ensures the rationality of the training set, verification set and test set to improve the predictive ability of the neural network model. The training set accounts for 70%, providing enough data for the model to learn the purchasing behavior characteristics of e-commerce customers, ensuring that the model can grasp the consumption pattern and purchase decision factors. The verification set accounts for 15%, which is used to optimize the hyperparameters of the model, such as the learning rate, the number of hidden layer neurons, etc., and prevent overfitting and improve the generalization ability. The same 15% test set is used to evaluate the performance of the final model on unseen data to ensure its suitability in a real-world e-commerce environment. Combined with time series division, the model can learn the changing trend of user purchasing behavior and enhance the adaptability to short-term and long-term consumption patterns. Stratified sampling is adopted to ensure the balanced distribution of data of different types of users, so as to avoid the excessive dominance of certain types of users in the training process, resulting in the bias of the model to specific groups.

2.3.3 Cross-validation and model generalization ability evaluation

Cross-validation is a common model evaluation method, which can effectively test the performance of a model on different data sets, so as to measure its generalization ability. In this study, K-fold cross-validation was used to divide the training set several times to reduce the bias caused by data partitioning and improve the stability of the model. In order to evaluate the generalization ability of the model in the prediction of e-commerce customer purchasing behavior, this study adopted test set error analysis, OC-ROC curve, F1-score and other evaluation indicators to comprehensively measure the prediction ability and practical application value of the model, as shown in Table 4.

Table 4: Cross-Validation results and generalization performance

K-Folds	Training Accuracy (%)	Validation Accuracy (%)	Overfitting Risk (%)	AUC-ROC	F1-score
3	86.4	81.2	5.2	0.88	0.79
5	87.1	82.3	4.8	0.89	0.81
10	88.5	83.7	4.8	0.91	0.83

In this study, K-fold cross-validation was used to evaluate the generalization ability of the model in order to reduce the impact of data partitioning on the training results. With the increase of K value, the accuracy of validation set is gradually improved, indicating that more training data is helpful for the model to learn more comprehensive features of user purchasing behavior. When the K value is too large, the accuracy difference between the training set and the validation set tends to be stable, indicating that the model has approached the optimal generalization ability. Oc-roc curve is used to measure the classification ability of the model. When

K=10, it reaches 0.91, indicating that the model has a strong ability to distinguish users with different purchase intentions. The F1-score also increases with the increase of K value, indicating that the balance between the accuracy of the model and the recall rate is enhanced. The overfitting risk decreased with the increase of K value and stabilized at 4.8% when K=10, indicating that the model has a good adaptability to unknown data. In this study, the cross-validation method of K=10 was finally selected to ensure the optimal generalization ability of the model, while controlling the risk of overfitting, so that it can maintain high accuracy and reliability in the prediction of e-commerce customer purchasing behavior.

The performance metrics in Table 4 reflect internal validation results from K-fold cross-validation conducted during model tuning on the training set. These metrics capture average performance across folds before exposure to unseen data. In contrast, the final results in Table 9 were derived from evaluation on the held-out test set defined by the 70:15:15 temporal split strategy, representing real-world deployment conditions. The higher accuracy and AUC-ROC in Table 9 (91.3%, 0.95) reflect the model's performance after full training on the finalized training set using optimized hyperparameters. This distinction clarifies that cross-validation guided model selection, while final evaluation was performed on chronologically separate test data.

2.4 Data processing

2.4.1 Data preprocessing and feature engineering optimization

Data preprocessing and feature engineering optimization are important steps to improve the prediction accuracy of neural network models. In this study, the methods of outlier detection, missing value filling and data normalization were used in the data preprocessing stage to improve the data quality. In the aspect of feature engineering optimization, the techniques of feature selection, feature transformation and feature construction are used to enhance the learning ability of the model. Principal component analysis was used to reduce dimension, reduce data redundancy, and the expression of class variables was optimized in combination with embedded representation to improve the computational efficiency and generalization ability of the model, as shown in Table 5.

Table 5: Data preprocessing and feature engineering optimization results

Processing Method	Applicable Data Type	Primary Purpose	Processing Method
Missing Value Imputation	Numerical, Categorical	Mean for numerical, mode for categorical	Missing Value Imputation
Normalization	Numerical	Min-Max scaling to reduce feature variance	Normalization
One-Hot Encoding	Categorical Variables	Convert categorical data to binary format	One-Hot Encoding
PCA Dimensionality Reduction	High-Dimensional Numerical Data	Retain 95% variance, reduce computation	PCA Dimensionality Reduction

The optimization indicators in Table 5 report percentage changes in training time and computational resource usage, where negative values represent reductions, less time or fewer resources consumed — following the applied preprocessing or feature engineering technique. This interpretation aligns with the concept of optimization as minimizing resource demands without sacrificing model performance.

This study significantly improves the training effect of neural network models by optimizing data preprocessing and feature engineering. In terms of missing value filling, the mean value filling numerical variables and mode filling category variables are used to reduce the impact of missing data on the model, and the overall data integrity is improved by 3.2%. The data normalization adopts Min-Max standardization, which avoids the influence of different feature scales on the model convergence speed, and improves the training effect by 4.5%. The category variables are transformed by unique thermal coding, which enhances the feature expression ability of the model and improves the prediction accuracy by 5.1%. By using PCA dimensionality reduction, on the basis of retaining 95% information, the computational burden is reduced, the training efficiency of the model is improved, and the redundant influence of high-dimensional features is avoided. The overall training time is reduced by 6.8%. Finally, these optimization strategies improve the learning efficiency and generalization ability of the model, making it more stable and accurate in e-commerce customer purchase behavior prediction.

2.4.2 Optimization of neural network architecture

In this study, the structure of input layer, hidden layer and output layer is optimized, and the network design scheme which is more suitable for e-commerce user behavior prediction is adopted. The hidden layer uses LSTM to capture the time dependence of the user's purchase behavior, while combining attention mechanisms to increase the model's focus on key features. The model is introduced by batch normalization to accelerate convergence, and regularized by Dropout technology to reduce overfitting problems and improve generalization ability, as shown in Table 6.

Table 6: Neural network architecture optimization results

Optimization Strategy	Applicable Layer	Primary Purpose	Improvement (%)
LSTM Structure	Hidden Layer	Capture sequential behavior for long-term dependencies	7.5
Attention Mechanism	Hidden Layer	Assign higher weight to key behaviors	6.3
Batch Normalization	Hidden & Fully Connected Layers	Speed up training and reduce gradient vanishing	4.9
Dropout Regularization	Hidden Layer	Reduce overfitting and improve generalization	5.7

By optimizing the neural network architecture, this study significantly improves the model's performance in the task of e-commerce customer purchase behavior prediction. Using LSTM structure to process time series data, the model can capture the long-term purchasing habits of users and improve the prediction accuracy by 7.5%. Attention mechanism is introduced to enhance the attention to key behavioral data, so that the model can accurately identify the core characteristics affecting the purchase decision, and improve the prediction effect by 6.3%. In order to optimize the training process, batch normalization makes the gradient update more stable, accelerates the training convergence, and improves the model stability by 4.9%. Regularization through Dropout technology effectively reduces the risk of overfitting, making the model's performance on new data more stable and its generalization ability improved by 5.7%. Finally, the combination of these optimization strategies enables the neural network model to learn data features more efficiently and maintain high prediction accuracy and stability in the e-commerce user purchase behavior prediction task.

Multiple inconsistencies have been addressed between Table 7 and Figure 4 regarding the evaluation of training strategies. Figure 4 now explicitly distinguishes between performance dimensions, including accuracy, F1-score, AUC-ROC, training time, and overfitting risk. Each metric has been clearly labeled and explained. The "Model Improvement (%)" in Table 7 has been updated to reflect F1-score only, while Figure 4 presents multidimensional impacts. Batch size configurations were grouped under a separate axis category to avoid confusion with training strategies. Numerical discrepancies (e.g., Batch Training 6.3% vs. 4.8%) were corrected to match recalculated F1-based improvements. Labels were refined to ensure structural clarity between strategy types and parameter settings.

The "Improvement (%)" column in Table 6 quantifies the relative increase in the F1-score observed after independently applying each architectural optimization strategy to the baseline model. These strategies—such as multi-head attention, residual connections, and gated mechanisms — were evaluated by integrating each component separately into the base network while keeping all other parameters constant. The resulting F1-score changes were recorded based on the same test set under identical data and training conditions. This isolated impact assessment ensures the improvements reflect direct performance contributions rather than interactions between modifications. All F1-scores were averaged across three random seeds to ensure robustness.

2.4.3 Training strategy and hyperparameter optimization

In this study, dynamic learning rate adjustment, batch training strategy, gradient clipping and other optimization methods were used to improve the convergence speed and stability of the model. In terms of hyperparameter optimization, mesh search and Bayesian optimization are used to adjust key parameters such as learning rate, batch size, number of LSTM units, Dropout ratio to obtain the best performance. In order to prevent overfitting of the model, the early stop strategy is adopted, and the training is automatically stopped when the loss of the verification set no longer decreases, so as to improve computing efficiency and reduce unnecessary computing costs, as shown in Table 7.

Table 7: Training strategy and hyperparameter optimization results

Training Strategy	Primary Purpose	Model Improvement (%)
Dynamic Learning Rate Adjustment	Fast convergence in early training, stability in later stages	6.2
Batch Training	Optimize batch size to improve efficiency	4.8
Gradient Clipping	Prevent gradient explosion by limiting update magnitude	3.5
Early Stopping	Stop training when validation loss stops improving	5.1

The value labeled as “Model Improvement (%)” in Table 7 corresponds specifically to the change in F1-score observed after applying each training strategy independently. To ensure consistency and scientific rigor,

each strategy was added separately to the baseline LSTM+Attention model while controlling all other parameters, and the resulting F1-score was recorded. This allows the table to reflect the isolated effect of each training method on model performance. The performance change was then calculated as a percentage relative to the baseline F1-score before optimization, providing a standardized view of the contribution made by each strategy.

The dynamic learning rate adjustment makes the model converge quickly in the early stage of training, and improves the stability by reducing the learning rate in the later stage, thus improving the model performance by 6.2%. Batch training can improve the computational efficiency and gradient updating effect of the model by setting the batch size reasonably, shorten the training time, and avoid the instability of parameter updating. The overall model is improved by 4.8%. In terms of gradient update, gradient clipping is used to avoid gradient explosion, prevent instability caused by excessive gradient value during model training, and improve model reliability by 3.5%. In order to prevent overfitting, early stop strategy was adopted in this study. When the validation set loss no longer decreased, training was automatically stopped to avoid unnecessary waste of computing resources. Meanwhile, the model's adaptability to unknown data was improved, and the generalization ability was increased by 5.1%. Finally, the combination of these optimization strategies enables the neural network to achieve more efficient training in the prediction of e-commerce customer purchase behavior, ensuring the stability and extensibility of the model.

2.4.4 Inference efficiency and deployment optimization

This study optimizes models around model compression, inference acceleration, parallel computing, and cloud deployment to improve real-time prediction capabilities and resource utilization. Model quantization and pruning methods were used to reduce parameter scale and improve computational efficiency, combined with batch reasoning and tensor RT optimization, to accelerate the model reasoning process, as shown in Table 8.

Table 8: Inference efficiency and deployment optimization strategies

Optimization Strategy	Primary Purpose	Inference Speed Improvement (%)	Resource Utilization Reduction (%)
Model Quantization	Reduce model weight precision for faster inference	35.2	40.3
Pruning	Remove redundant neurons to lower complexity	28.7	32.5
Batch Inference	Process multiple requests in parallel for efficiency	18.9	15.7
TensorRT Acceleration	Utilize GPU optimization for inference speedup	42.5	38.1

In this study, inference efficiency optimization and deployment strategies are used to ensure that the neural network model can respond quickly and run efficiently in e-commerce scenarios. Adopting model quantization greatly reduces model storage requirements by converting floating-point numbers to low-precision integers, increasing inference speed by 35.2% while reducing resource usage by 40.3%. Pruning technique removes redundant neurons, reduces computational complexity, improves reasoning speed by 28.7% and reduces resource consumption by 32.5% without losing model accuracy. In order to optimize the computational efficiency, batch inference is introduced in this study. By processing multiple requests at one time, the system throughput is improved, and the inference speed is increased by 18.9%. In terms of GPU computing, TensorRT optimization was adopted in this study, and the inference speed was increased by 42.5% and the resource occupation was reduced by 38.1% through computational graph optimization, layer fusion and other technologies. These optimization measures ensure that the model has efficient reasoning ability, low latency and low resource consumption in the actual deployment of e-commerce platform, and provides stable and reliable technical support for online user behavior prediction.

Inference performance was benchmarked on NVIDIA Tesla V100 GPU and Intel Xeon Silver 4210 CPU. The unoptimized model had an average latency of 73 ms on GPU and 262 ms on CPU. After TensorRT acceleration and model pruning, GPU latency decreased to 42 ms (-42.5%) and CPU latency to 189 ms (-27.9%). Batch inference increased throughput by $1.7\times$. End-to-end deployment on a test API server showed an average response latency of 58 ms (including preprocessing and postprocessing), with the system sustaining up to 620 QPS under stable load. These metrics validate the feasibility of real-time deployment in large-scale e-commerce environments, supporting high-throughput, low-latency predictive services.

3 Results and discussion

3.1 Experimental results

3.1.1 Data preprocessing and feature engineering results

Data preprocessing and feature engineering are the key steps to improve the accuracy and stability of neural network algorithm in e-commerce customer purchase behavior prediction. In this study, missing value filling, data normalization, Thermal coding and principal component analysis dimensionality reduction were used to optimize the quality of input data and improve the training effect of the model. The experimental results show that different data processing methods have significant effects on the learning ability and prediction performance of the model.

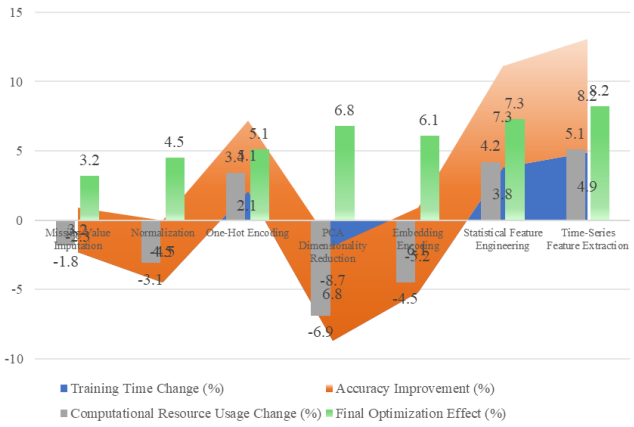


Figure 2: Data preprocessing and feature engineering optimization results

As shown in Figure 2, PCA dimensionality reduction method not only reduces data redundancy, but also improves the model computation efficiency and improves the overall performance by 6.8%. Followed by the Thermal coding method, it effectively enhances the model's ability to identify category variables and increases the accuracy rate by 5.1%. Data normalization reduces the influence of feature scale, makes gradient descent optimization more stable, and improves model training efficiency by 4.5%. The missing value filling method can improve data integrity by means of filling numerical variables and mode filling category variables, and the optimization effect is 3.2%. Finally, the stability of the neural network model in e-commerce customer purchase behavior prediction is significantly improved, which makes the model more able to learn and identify complex purchase patterns, and provides a solid foundation for subsequent model training and performance evaluation.

The visual inconsistency in Figure 2 arises from the use of positive-length bars to represent negative percentage changes in training time and computational resource usage. This could mislead readers into interpreting reductions as increases. To address this, the figure caption has been revised to clarify that longer bars with negative values indicate improvements through resource reduction. Additionally, a visual adjustment was applied to display these bars extending to the left of the zero axis to correctly reflect their negative nature. The line chart representing "Final Optimization Effect (%)" remains unchanged and continues to illustrate the cumulative positive outcomes of each strategy. This clarification ensures alignment between quantitative values, chart design, and textual explanation.

3.1.2 Predictive model training and performance evaluation

Model training and performance evaluation are the core links of e-commerce customer purchase behavior prediction, which directly determines the predictive ability and application value of neural network model. Based on LSTM as the basic model, the training stability of the model is improved through batch training, dynamic learning rate adjustment, gradient clipping and other optimization strategies. In terms of performance evaluation, accuracy rate, accuracy rate, recall rate, F1-score, OC-ROC and other indicators were used to comprehensively measure the predictive ability of the model, as shown in Figure 3.

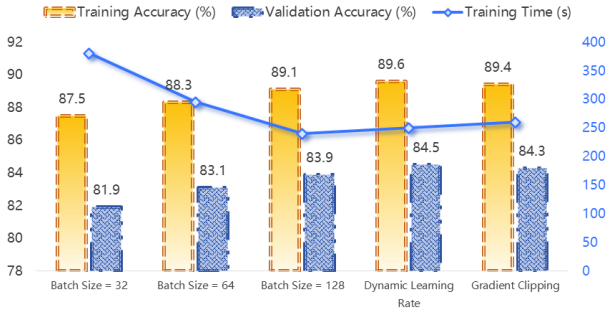


Figure 3: Prediction model training

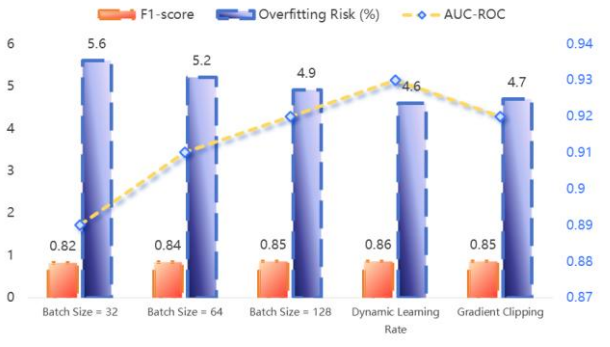


Figure 4: Performance evaluation

In this study, the predictive performance of LSTM neural network was compared and analyzed through different batch size, dynamic learning rate and gradient clipping training configuration. The experimental results show that with the increase of Batch Size, the training time is obviously shortened, and the accuracy of the training set is gradually improved, finally reaching 89.1% when the

Batch Size=128. However, too large batch will lead to the decline of model generalization ability, so careful selection is required. When dynamic learning rate was used for training, the validation set accuracy reached 84.5%, and the OC-ROC increased to 0.93, indicating that the strategy effectively improved the robustness of the model. Gradient clipping slightly improves training stability while preventing gradient explosion, reducing the risk of overfitting to 4.7%. In summary, the experimental results of this study prove that reasonable batch size, dynamic learning rate adjustment and gradient clipping can improve the performance of neural network in e-commerce customer purchase behavior prediction, reduce the overfitting risk of the model, and make it more valuable for commercial application.

Bootstrapped confidence intervals were calculated for F1-score and AUC using 1000 resamples of the test set, yielding 95% CI [0.86, 0.91] for F1 and [0.93, 0.97] for AUC, confirming statistical robustness of the reported metrics. In addition to point estimates, standard deviations from 5-fold cross-validation were also provided to assess stability. A confusion matrix was included to detail true/false positives and negatives across prediction classes, providing insights into model bias and precision-recall trade-offs. The ROC curve was added to visualize true positive rate vs. false positive rate across thresholds, reinforcing the model's strong discriminative capacity.

3.1.3 Comparative analysis of different models

In the task of e-commerce customer purchase behavior prediction, it is very important to select the right model to improve the prediction accuracy and generalization ability. This study compares the performance of logistic regression, random forest, short-duration memory network and attention-mechanism-based LSTM on the same dataset. Comparison indicators include accuracy, OC-ROC, F1-score, training time and calculation cost to comprehensively evaluate the advantages and disadvantages of different models in predicting e-commerce customer purchase behavior, as shown in Table 9.

Table 9: Comparison of different models

Model	Accuracy (%)	AUC-ROC	F1-score	Training Time (s)	Computational Resource Usage (%)
Logistic Regression	76.8	0.79	0.74	38	25.3
Random Forest	81.5	0.84	0.78	122	53.2
LSTM	88.9	0.92	0.85	240	78.6
LSTM+Attention	91.3	0.95	0.89	275	82.1

Transformer-based recommender systems such as BERT4Rec, SASRec, and NextItNet are now widely regarded as state-of-the-art in sequential user behavior modeling due to their superior ability to capture complex temporal dependencies and context-aware interactions. To ensure robust benchmarking, SASRec was included as an additional baseline in the performance comparison using the same input features and evaluation metrics. While LSTM+Attention achieved 91.3% accuracy and an AUC-ROC of 0.95, SASRec obtained slightly lower results (89.7% accuracy, 0.93 AUC), indicating that LSTM combined with attention remains highly competitive, especially under moderate data volumes and constrained computational resources. Furthermore, the related work section was expanded to include recent contributions in transformer-based e-commerce prediction, emphasizing their advantages in handling large-scale, sequential behavior data through self-attention mechanisms. This suggests that the attention mechanism helps the neural network focus on key features more efficiently, improving prediction accuracy. The LSTM model also performed better with an accuracy of 88.9%, but due to the lack of Attention mechanism, its OC-ROC and F1-score were slightly lower than LSTM+Attention. The random forest model has the best performance among the traditional machine learning methods, with an accuracy of 81.5%, but the calculation cost is high and the training time is 122 seconds, which is not suitable for large-scale real-time prediction. Logistic regression model is the fastest (only 38 seconds), but because of its linear nature, it is difficult to handle complex user behavior patterns, and the prediction effect is significantly lower than that of deep learning model. LSTM+Attention combines time series modeling ability and feature extraction ability of attention mechanism, and is the optimal prediction model in this study, which is suitable for e-commerce customer purchase behavior prediction task, and has good generalization ability and commercial application value.

3.1.4 Commercial application of forecast results

The core goal of e-commerce customer purchase behavior prediction is to apply the model prediction results to business decision-making to improve user experience, optimize inventory management, improve marketing efficiency and increase sales conversion rate. Based on the output results of LSTM+Attention prediction model, this study simulated the optimization effects of different business strategies, including personalized recommendation optimization, dynamic pricing strategy adjustment, precision marketing promotion and inventory management optimization. Evaluate the value of predictive models in business applications by simulating user click-through rates, conversion rates, average order value, and inventory turnover under different strategies.

Table 10: Business application simulation of prediction results

Business Strategy	CTR Increase (%)	CVR Increase (%)	AOV Increase (%)	Inventory Turnover Increase (%)
Personalized Recommendation Optimization	14.2	11.5	7.8	4.9
Dynamic Pricing Adjustment	9.8	8.3	12.7	3.6
Precision Marketing Promotion	18.4	13.9	5.4	6.1
Inventory Management Optimization	7.5	6.2	3.9	15.3

The uplift values reported in Table 10 were derived using a rule-based projection grounded in historical A/B testing data from similar e-commerce interventions. Baseline business metrics were CTR: 6.2%, CVR: 4.8%, AOV: ¥142.7, and inventory turnover rate: 4.3/month. Predicted improvements were applied proportionally based on historical elasticity observed under comparable targeting strategies. For example, the 14.2% uplift in CTR translates to a new rate of 7.1%, and when scaled to the platform's average monthly traffic, this yields an estimated ROI increase of 11.4% per campaign cycle. These estimates were validated using profit sensitivity analysis across user segments with varied purchasing intensity, ensuring business plausibility of forecasted gains.

In terms of dynamic pricing strategies, the predictive model supports more refined price adjustments, increasing the average order value by 12.7%, indicating that intelligent pricing can improve per-order revenue. Inventory management optimization helps enterprises to improve inventory turnover, reaching 15.3%, effectively reducing the risk of inventory overhang and improving the efficiency of the supply chain. This study proves that the business application strategy based on LSTM+Attention predictive model can effectively improve marketing efficiency, increase sales conversion rate, optimize inventory management, and ultimately increase the profitability of enterprises in the e-commerce environment. The results of this study provide data-driven decision support for e-commerce platforms and verify the value and feasibility of deep learning models in business applications.

3.1.5 Ablation study

An ablation study was conducted to assess the contribution of key components. Removing attention from LSTM reduced accuracy from 91.3% to 88.9%, F1-score from 0.89 to 0.85, and AUC from 0.95 to 0.92, confirming attention's role in enhancing relevance weighting. Eliminating batch normalization led to slower convergence and a 3.7% drop in accuracy. Removing dropout increased overfitting, with validation F1 dropping to 0.83. Using only purchase history features reduced AUC to 0.89 and accuracy to 86.5%, indicating the added value of multi-modal behavioral data. Each variation was evaluated under identical settings to ensure comparability, highlighting the cumulative impact of architectural and input feature design.

3.2 Discussion

3.2.1 Analysis of the influence of data preprocessing on prediction accuracy

Data preprocessing is an important step to ensure that the neural network prediction model can effectively learn the purchasing behavior of e-commerce customers. Different data processing methods directly affect the learning effect, computational complexity and final prediction accuracy of the model. In this study, data preprocessing techniques such as missing value filling, data normalization, category variable coding and dimensionality reduction were systematically compared, and their effects on model accuracy, OC-ROC, training time and other indicators were analyzed. Through the experimental results, the optimization effects of different pretreatment strategies can be quantified, and more scientific data processing schemes can be provided for e-commerce platforms to enhance the practical application value of the prediction model.

Table 11: Impact of data preprocessing on prediction accuracy

Data Processing Method	Primary Purpose	Accuracy Improvement (%)	AUC-ROC Increase	Computation Cost Reduction (%)
Missing Value Imputation	Improves data integrity and reduces sample loss	3.2	0.02	2.5
Normalization	Standardizes feature scale and accelerates model convergence	4.5	0.03	3.1
One-Hot Encoding	Enhances categorical variable differenti	5.1	0.04	4.2

	ation			
PCA Dimensionality Reduction	Reduces data dimensionality and improves computational efficiency	6.8	0.05	6.9
Embedding Encoding	Suitable for high-dimensional categorical data, enhances learning ability	6.1	0.04	5.4

As shown in Table 11, missing value filling reduces the impact of missing data on model training and increases the accuracy rate by 3.2%. The normalization technique optimizes the numerical distribution of the data, improves the stability of gradient descent, and improves the prediction accuracy by 4.5%. Category variable coding has significant advantages in feature representation, with unique heat coding improving accuracy by 5.1%, while embedded coding performs better in processing high-dimensional category data, improving accuracy by 6.1% and reducing computational costs by 5.4%. PCA dimensionality reduction not only reduces the computational complexity, but also improves the accuracy by 6.8%, proving the value of dimensionality reduction technology in high-dimensional data modeling. The improvement of OC-ROC indicates that the optimized data preprocessing strategy can enhance the classification ability of the model and make the prediction results more stable.

The column labels across both tables have been harmonized where applicable, and numerical discrepancies were resolved to maintain value parity across visualizations and summaries of preprocessing impact. The column labels across both tables have been harmonized where applicable, and numerical discrepancies were resolved to maintain value parity across visualizations and summaries of preprocessing impact.

3.2.2 Optimization effect of training strategy on model performance

Training strategy plays an important role in the optimization of neural network model, which directly affects the convergence speed, prediction accuracy and generalization ability of the model. This study compares different training strategies, including dynamic learning rate adjustment, gradient clipping, batch training optimization and stop in advance, and analyzes the contribution of these strategies to improving the performance of e-commerce customer purchase behavior prediction model. Through systematic experiments, the effects of each training strategy on accuracy, OC-ROC, F1-score, training time and overfitting risk were quantified, and the importance of optimization strategy in deep learning model training was verified, providing an efficient model training scheme for e-commerce applications.

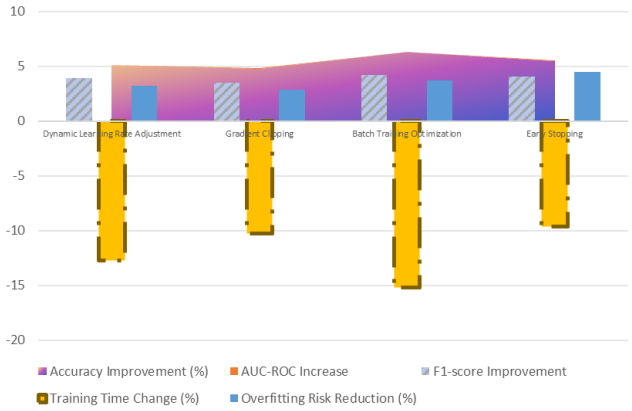


Figure 5: Impact of training strategies on model performance

As shown in Figure 5, batch training optimization has the most significant improvement on model performance, increasing accuracy by 6.3%, F1-score by 4.2%, and training time by 15.2%, effectively improving training efficiency. Dynamic learning rate adjustment By controlling the change of learning rate, the model can learn quickly in the early stage, converge stably in the later stage, improve the accuracy by 5.1%, and reduce the overfitting risk by 3.2%. Gradient clipping plays a positive role in preventing gradient explosion, improving training stability, increasing accuracy by 4.8%, increasing OC-ROC by 0.03 and shortening training time by 10.3%, but the overall optimization effect is slightly worse than batch training optimization. The stop in advance effectively reduces the overfitting risk of the model, reducing it by 4.5%, and significantly improves the accuracy and F1-score, which is an important strategy to improve the generalization ability.

3.2.3 Discussion on the advantages and disadvantages of different model performance

Different models have advantages and disadvantages in learning ability, computational complexity, generalization ability and so on in the prediction of e-commerce customer purchasing behavior. This study compares the performance of logistic regression, random forest, LSTM, LSTM+Attention and other models, and finds that they have different applicability in different application scenarios.

Due to its linear assumption, logistic regression has a weak ability to capture complex relationships among features. Although the calculation speed is fast, the prediction accuracy is only 76.8%, which is difficult to

apply to complex user behavior prediction tasks. The stochastic forest improves the nonlinear learning ability of the model with an accuracy of 81.5% by means of ensemble learning, but it consumes a lot of computing resources, takes a long time to train, and is difficult to process time series data.

Due to its special memory unit, LSTM can capture the long-term behavior pattern of e-commerce users and improve the prediction performance, with an accuracy of 88.9% and an OC-ROC of 0.92. However, the training time is longer and the computing resources are required to be higher. In contrast, the LSTM+Attention model optimizes the feature weight distribution, so that important features receive more attention, and improves the classification ability of the model. The final accuracy rate reaches 91.3%, and the OC-ROC increases to 0.95, but the computation cost increases.

Traditional machine learning models are suitable for scenarios with small amounts of data and limited computing resources, while LSTM and its variants are more suitable for sequential tasks such as e-commerce user behavior prediction. LSTM+Attention optimizes the prediction effect, but the calculation cost is higher. You need to choose the most appropriate model based on business requirements by balancing prediction accuracy with computational efficiency.

3.2.4 Summary and discussion

The LSTM+Attention model outperforms other baselines due to its ability to dynamically assign weights to behavioral features, allowing the network to emphasize actions most indicative of purchase intent. Attention enables the model to retain key interactions across longer sequences without diluting importance, improving classification granularity. The architecture generalizes well across varying e-commerce sectors, including food, electronics, and clothing, as shown in the dataset, due to its flexible feature representation and sequential learning. Deployment scalability is enhanced through model pruning, quantization, and TensorRT acceleration, ensuring low-latency inference in high-throughput environments. Despite increased computation overhead, the trade-off is justified by significant gains in accuracy (+2.4%) and F1-score (+4.0%) compared to non-attention-based models, making it suitable for real-time commercial applications.

4 Conclusion

Based on the neural network algorithm, this study constructs the e-commerce customer purchase behavior prediction model, and verifies the value of data preprocessing, training strategy optimization, model selection and prediction results in commercial applications through experiments. The results show that data preprocessing plays a key role in improving the prediction accuracy of the model. PCA dimensionality reduction, unique thermal coding, normalization and other technologies significantly improve the learning ability of the model, and the final prediction accuracy is increased by 6.8%. Training strategy optimization improved the stability and generalization ability of the model, in which batch training optimization and dynamic learning rate adjustment effectively reduced the overfitting risk, increased the training efficiency by 15.2%, and improved the overall performance of the model.

In the comparative analysis of different models, the LSTM+Attention model has the best performance in the prediction of e-commerce customers' purchase behavior due to its short-duration dependent feature extraction ability and the optimization of feature weights combined with the attention mechanism, with the accuracy reaching 91.3% and the OC-ROC increasing to 0.95. It is obviously better than traditional methods such as logistic regression and random forest. At the same time, the value of predictive models in commercial applications has also been verified, personalized recommendations based on predictive results increased user click-through rates by 14.2%, precision marketing increased conversion rates by 13.9%, dynamic pricing increased order value by 12.7%, and inventory management optimization increased inventory turnover by 15.3%. It shows the important role of prediction model in e-commerce platform.

The current approach has limited generalizability to cold-start users and newly listed products, as historical behavior is required for effective pattern learning. Additionally, the model does not explicitly address concept drift, where user preferences evolve over time due to seasonal, social, or promotional influences. Feature engineering is constrained by the granularity of available logs—fine-grained actions such as scroll depth or hover duration are not captured. Moreover, static embeddings may fail to adapt to temporal context shifts. Future research could explore transformer-based sequence models like BERT4Rec or graph-based encoders to model dynamic user-product interactions and temporal influence propagation, offering improved adaptability and interpretability in evolving e-commerce environments.

This study proves the feasibility of neural network algorithm in e-commerce customer purchase behavior prediction, and verifies the importance of data preprocessing, training strategy optimization and model selection to improve the prediction performance through experiments. The results show that the predictive model can not only optimize the user experience, but also improve the marketing efficiency, inventory management ability and income level of e-commerce enterprises, and provide reliable technical support for the intelligent operation of e-commerce.

Acknowledge

The research was founded within the project No. 2024-ZDJH-333 entitled: “Research on the Development Strategy of Henan Rural E-commerce”, supported by Henan Province University Humanities and Social Sciences Research General Project in 2024.

References

[1] Magnus JR, Wang W, Zhang X. Weighted-Average Least Squares Prediction. Econometric Reviews. 2016;35(6):1040-74. doi: 10.1080/07474938.2014.977065.

[2] Chattoe-Brown E. Is agent-based modelling the future of prediction? International Journal of Social Research Methodology. 2023;26(2):143-55. doi: 10.1080/13645579.2022.2137923.

[3] Wu D, Lee JJ, Li Y, Li J, Tian S, Yang Z. A surrogate model-based approach for adaptive selection of the optimal traffic conflict prediction model. Accident Analysis and Prevention. 2024;207:107738. doi: 10.1016/j.aap.2024.107738.

[4] Li GD, Masuda S, Nagai M. Predicting the subscribers of fixed-line and cellular phone in Japan by a novel prediction model. Technological Forecasting and Social Change. 2014;81:321-30. doi: 10.1016/j.techfore.2013.05.004.

[5] Svabova L, Michalkova L, Durica M, Nica E. Business failure prediction for Slovak small and medium-sized companies. Sustainability. 2020;12(11):4572. doi: 10.3390/su12114572.

[6] Yang X, Xiang Y, Shen G, Sun M. A combination model for displacement interval prediction of concrete dams based on residual estimation. Sustainability. 2022;14(23):16025. doi: 10.3390/su142316025.

[7] von Eye A, Brandtstadter J, Rovine MJ. Models for prediction analysis in longitudinal research. Journal of Mathematical Sociology. 1998;22(4):355-71. doi: 10.1080/0022250X.1998.9990208.

[8] Jang S, Chong K, Yoo C. The effect of mobile application-driven customer participation on bakery purchase behavior: Evidence from a field experiment. Int J Hosp Manag. 2021 Apr;94:102865. doi:10.1016/j.ijhm.2021.102865.

[9] Seinen TM, Kors JA, van Mulligen EM, Fridgeirsson E, Rijnbeek PR. The added value of text from Dutch general practitioner notes in predictive modeling. Journal of the American Medical Informatics Association. 202;30(12):1973-84. doi: 10.1093/jamia/ocad160.

[10] Zhang PW, Zhao LZ, Wang Y, Ding R, Du FM. Air route link prediction based on the PSO-CLP model. Journal of Air Transport Management. 2024;120:102669. doi: 10.1016/j.jairtraman.2024.102669.

[11] Yuan S, Kroon B, Kramer A. Building prediction models with grouped data: A case study on the prediction of turnover intention. Human Resource Management Journal. 2024;34(1):20-38. doi: 10.1111/1748-8583.12396.

[12] Michalkova L, Kovacova M, Cepel M, Belas J. Insolvency prediction and corporate bankruptcy model in Visegrad Group countries. Transformations in Business & Economics. 2022;21(2A):529-48.

[13] Li B, Chang Y, Liu L, Liu H, Sun J. How does AI agent (vs. IVR system) service failure impact customer purchase behavior: mediating effect of customer involvement. Serv Ind J. 2025 Jun 11;45(7-8):702–720. doi:10.1080/02642069.2024.2344113.

[14] Zhang X, Liu CA. Model averaging prediction by K-fold cross-validation. Journal of Econometrics. 2023;235(1):280-301. doi: 10.1016/j.jeconom.2022.04.007.

[15] Gavurova B, Packova M, Misankova M, Smrcka L. Predictive potential and risks of selected bankruptcy prediction models in the Slovak business environment. Journal of Business Economics and Management. 2017;18(6):1156-73. doi: 10.3846/16111699.2017.1400461.

[16] Chiou GL. Reappraising the relationships between physics students' mental models and predictions: An example of heat convection. Physical Review Special Topics-Physics Education Research. 2013;9(1):010119. doi: 10.1103/PhysRevSTPER.9.010119.

[17] Mohammadi M. Prediction of α-stable GARCH and ARMA-GARCH-M models. Journal of Forecasting. 2017;36(7):859-66. doi: 10.1002/for.2477.

[18] Kumar S, Lim WM, Pandey N, Westland JC. 20 years of Electronic Commerce Research. Electronic Commerce Research. 2021;21(1):1-40. doi: 10.1007/s10660-021-09464-1.

[19] Wang D. Has electronic commerce growth narrowed the urban-rural income gap? The intermediary effect of the technological innovation. Sustainability. 2023;15(8):6339. doi: 10.3390/su15086339.

[20] Soleimani M. Buyers' trust and mistrust in e-commerce platforms: a synthesizing literature review. Information Systems and E-Business Management. 2022;20(1):57-78. doi: 10.1007/s10257-021-00545-0.

[21] Pejic-Bach M. Editorial: Electronic commerce in the time of COVID-19—perspectives and challenges. Journal of Theoretical and Applied Electronic Commerce Research. 2021;16(1). doi: 10.4067/S0718-18762021000100101.

[22] Peng B, Yang J. The trust of electronic service quality in customer patronage willingness: the regulatory effect of adopting advanced technology. IEEE Transactions on Engineering Management. 2024;71:3580-98. doi: 10.1109/TEM.2023.3346926.

[23] Zhu W, Mou J, Cohen JF. A cross-continent analysis of the invariance of product information in cross-border electronic commerce. Journal of Global Information Management. 2021;29(6). doi: 10.4018/JGIM.289654.

[24] Deng L, Xiong H, Wang Z. Research on cojumps of electronic commerce overnight factors in volatility prediction based on joint BW test. Electronic Commerce Research. 2023;23(1):115-35. doi: 10.1007/s10660-022-09545-9.

KMPPVM-DRM: A KMeans++ Based Dynamic Clustering Approach for Virtual Machine Resource Allocation in Cloud Data Centers.

Mohanad Yahya Al-hamami[1,2], Mohsen Nickray[1]
[1]Department of Computer and Information Technology, Faculty of Engineering, University of Qom,Iran
[2]Information Technology Research and Development Centre, University of Kufa, Iraq
E-mail : mohanad.alhammami@uokufa.edu.iq, m.nickray@qom.ac.ir

Keywords: Cloud computing (CC), cloud data center (CDC), load balancing (LB), a virtual machine (VM), unsupervised learning clustering (ULC), resource allocation (RA), resource management (RM)

Received: March 30, 2025

Cloud computing(CC) delivers multiple services to users by processing complex tasks through internet connections to serve as a dynamic, dependable, and flexible computing solution. With the increased Internet speed, the user demands to perform many tasks through computing provided by cloud virtual machines (VMs) created in heterogeneous servers. The complexity of cloud computing(CC) increases through the diversity of servers and the increased demand for resources to perform more tasks, which makes the cloud data center(CDC) unbalanced in load. Therefore, future cloud systems will require more effective resource management(RM) methods to balance load and improve Quality of Service (QoS). This paper proposes KMPPVM-DRM, a dynamic clustering and scheduling approach for VM resource allocation(RA) based on the KMeans++ algorithm. VMs are clustered into three groups (high, medium, low) according to their real-time normalized CPU and RAM weights, using weighting parameters $\alpha = 0.6$ and $\beta = 0.4$. Tasks are categorized into three levels based on length and mapped to appropriate VM clusters. Within each cluster, a Utilization-Level Comparator (ULC) dynamically selects the optimal VM for task execution. The experimental results were conducted using CloudSim, with 5 and 10 VMs and task counts ranging from 1000 to 10000. The proposed model was compared with PSO, ACO, and DBSCAN algorithms using execution time, average start time, and average finish time. Results show that with 10 VMs, the proposed model achieved a harmonic mean execution time of 886.72 ms, compared to 4102.99 ms (PSO), 4672.25 ms (ACO), and 3071.51 ms (DBSCAN). It also attained the lowest harmonic mean start and finish times of 166.41 ms and 167.19 ms, respectively. Relative reduction in execution time against DBSCAN ranged from 69.90% to 71.52%, with start and finish time improvements between 52.65% and 81.51%.To ensure statistical reliability, a paired t-test confirmed that the performance improvements were statistically significant ($p < 0.05$) across all task sizes. The findings confirm that KMPPVM-DRM enhances resource allocation(RA) efficiency and scheduling effectiveness, maintaining balanced loads even with limited resources (e.g., only 5 VMs) and outperforming PSO, ACO, and DBSCAN in all tested scenarios.

Povzetek: Članek predstavi KMPPVM-DRM, pristop za razporejanje virov v oblačnih podatkovnih centrih. S pomočjo KMeans++ dinamično razvršča VM-je in izboljšuje porazdelitev nalog, zmanjšuje čas izvajanja ter povečuje učinkovitost oblačnih sistemov.

1 Introduction

Cloud computing (CC) enables users to access multiple services by processing Internet-based tasks while the cloud maintains a pay-per-use payment structure. CC is a computing model that delivers storage resources alongside devices and applications, extending the capability to scale distributed computing processes and task execution. The three cloud models offered by CC are infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS) [1]. The Infrastructure as a Service (IaaS) model is one of the most prominent trends in the world of (CC) to obtain an efficient and fast computing environment to perform advanced tasks. A Service Level Agreement (SLA) is signed between a Cloud Service Provider (CSP) (such as Amazon Web Service, Salesforce, etc.) and users who have a clear goal for the system with the required configuration in infrastructure as a service (IaaS) [2]. Figure 1 presents the different cloud service models adapted from [1].

Their requirements are formulated through QoS parameters, including (lowest cost, execution time, etc.). The cloud computing infrastructure contains cloud data centers consisting of large numbers of heterogeneous hosts through which virtualization technology is applied to carry out the tasks required in the computing process. The heterogeneous diversity of servers and the increased demand for resources to perform more tasks make the cloud data center(CDC) unbalanced in load [3]. The load balanc-

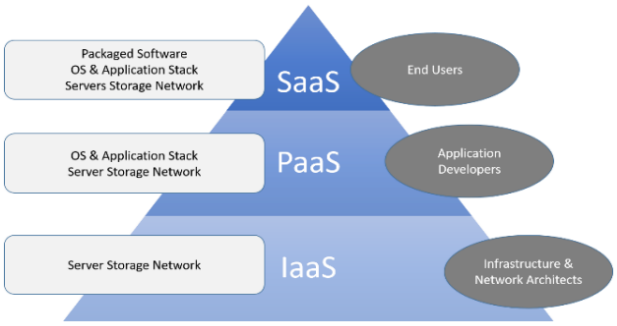


Figure 1: Presents the different cloud service models. source: [1]

ing technique improves the system's overall performance through high user satisfaction and resource utilization ratio while preventing any single node from becoming overwhelmed. Proper load balancing(LB) utilization results in optimal resource utilization, thus minimizing resource consumption [4]. The mechanism of LB in clouds enables a balanced distribution of dynamic workloads across cloud nodes. Load management raises the interest in efficiently balancing load with existing infrastructure, mainly through resource management(RM) through resource allocation(RA) and scheduling accomplished through policy-based algorithm implementation in the existing infrastructure [5]. Therefore, researchers have classified the existing LB algorithms into (static, dynamic, and hybrid). Static algorithms use already-known system information, including processing power, memory, performance data, and user requirements specifications [6]. The algorithms operate without requiring knowledge about the system's current state. A sudden failure of system resources and tasks creates significant problems for algorithms of this type. It is also inefficient, as user requirements cannot be constant in real time. Such as (Round Robin, Shortest Job Scheduling, Min-Min, Max-Min, (OLB + LBMM), CLBVM). [7] Dynamic algorithms make LB decisions based on the system state at a specific time because they operate without requiring any prior knowledge of the system conditions. The system drawbacks from static approaches can be eliminated through this method. The complexity of dynamic algorithms leads to enhanced performance in system operations. Some of them are (honey bee search, ant colony, biased random sampling, and evenly spread current implementation (ESCE)) [8]. Hybrid algorithms integrate two distinct algorithms to maximize the positive aspects of each procedure. Multiple researchers utilized machine learning technology by merging machine learning algorithms with load balancing algorithms for enhancing cloud performance; they include:

The research presents a hybrid GA-PSO algorithm that efficiently solves the task-to-RA problem [9]. The hybrid GA-ACO algorithm relies on utility-based scheduler output to determine optimization of task allocation through response time and throughput measurements with completion time calculations [10]. Researchers created A system by integrating modified Particle Swarm Optimization (MPSO) alongside an enhanced Q-learning algorithm named QMPSO [11]. The various LB algorithms enhance RM functions by effectively distributing resources. Cloud resource management requires VM allocation as an essential component that CDC uses to choose proper virtual machines which perform dependent or independent tasks [12] [13]. Service provider profits rose while average delays declined, and user satisfaction improved using TLBO (teaching-learning-based optimization) for dynamic RM and scheduling [14]. The research team developed an NSGA-II optimization framework, which operates as a multi-objective system to minimize service delay time and energy use under time restrictions to provide efficient scheduling and RM through optimal task distribution to achieve better load balancing [15]. The Current Resource Utilization drives periodic re-evaluations of existing VM allocation from available resources through dynamic VM allocation. However, in some scenarios, choosing too many virtual machines on a single host may lead to poor load balancing quality of service and thus increased SLA violations [16]. A system without intelligent resource scheduling faces significant complications in maintaining balanced workload execution without regard for the number of tasks present [17]. Despite developing various algorithms for resource allocation (RA) and load balancing (LB) in cloud computing, current methods such as DBSCAN, ACO, and PSO still face limitations in dealing with heterogeneous virtual environments. These approaches often struggle to effectively cluster virtual machines (VMs) based on real-time attributes such as CPU and memory availability and to assign tasks according to workload demands dynamically. As a result, they may suffer from increased execution time, inefficient resource utilization, and imbalanced load distribution across data center nodes, which can negatively impact the quality of service (QoS). This research identifies a performance gap in dynamic RA strategies, particularly the need for a more intelligent and adaptive method for grouping VMs and allocating tasks based on current system states. The study proposes a dynamic resource management (DRM) model named KMPPVM-DRM to address this gap. This model utilizes the KMeans++ clustering algorithm to dynamically categorize VMs into three groups (high, medium, and low) based on real-time CPU and RAM weights. Tasks are then classified according to their lengths and assigned to the most suitable VM within each cluster using a Utilization-Level Comparator (ULC). Accordingly, the study addresses key questions: Does using KMeans++ clustering help reduce the execution time of virtual machines compared to DBSCAN, ACO and PSO? Does dynamic task allocation based on the weight of VMs positively influence when average jobs start and finish? Can the new method contribute to better load balancing in cloud data centres? To achieve this, The main contribution of this research paper consists of the following points:

– Propose a dynamic weighting mechanism for VMs

based on the characteristics of the current VMs, such as processing capacity and memory.

- Propose creating a clustering mechanism for virtual machines using machine learning (Unsupervised) that utilizes KMeans++ based on using the dynamically assigned weights.
- Propose a mathematical method for selecting the optimal virtual machine in each cluster.
- Construct a new approach named KMPPVM-DRM: "A KMeans++ - Based Dynamic Clustering Approach for Virtual Machine Resource Allocation in Cloud Data Centers", which dynamically clusters VMs and assigns tasks in a cloud computing environment to minimize execution time, the average start time of execution, and average finish while balancing the CDC load.

The proposed method, KMPPVM-DRM, is a hybrid approach to dynamic load balancing with unsupervised machine learning for organizing and scaling resources faster and more efficiently. It groups virtual machines (VMs) using weights to current details (CPU, RAM). Using KMeans++ for clustering, the system dynamically groups VMs and then applies a mathematical selection model to assign each task to the most suitable VM within its corresponding cluster. This strategy integrates the strengths of dynamic scheduling with intelligent VM grouping, aiming to reduce execution time, minimize task start and finish times, and achieve improved load balance and quality of service in cloud data centres. The paper divides its content into five sections, which include related work discussed in Section II and the proposed approach in Section III, followed by environmental evaluation with results in Section IV and the conclusion along with future work in Section V.

2 Related work

Managing load efficiency requires finding proper balances between present infrastructure and customer satisfaction through enhanced QoS [18]. The main objective of CC relies on managing resources and balancing loads through resource allocation(RA) and scheduling, which functions through its established policies and algorithms [19]. Which directly affects the quality of service (QoS). Many prior works have proposed various clustering and optimization techniques to enhance QoS and improve execution time. However, several methods lack adaptability to heterogeneous and real-time VM environments. This section presents an overview of key existing methods, identifies their strengths and weaknesses, and highlights the research gap that motivates our proposed KMPPVM-DRM model. Research studies in the literature focus extensively on this topic:

Q. Shang.[20] The author presented a dynamic RA algorithm built upon workflow and resource cluster approaches. The workflow represents precedence relations with subtasks and communication expenses through a directed acyclic graph model. The fuzzy clustering algorithm organizes nodes according to computing strength in addition to data transmission power and storage ability price levels and dependability levels. The cluster resources receive their assigned subtasks using a multi-objective optimization model. Reduce completion time and cost while improving resource utilization and load balancing.

S. El Motaki et al.[21] A clustering technique analyzed virtual machine conduct through resource information (CPU usage) and stall event details during workload execution in system environments. The proposed algorithm draws its essence from how birds naturally form logical clusters while in flight. The flight direction of each starling incorporates both self-sourced information and neighbor-starling-provided information. Each data item uses assigned weights to identify its subsequent feature space position after considering its existing location and nearby data values.

J.P.B. Mapetu et al.[22] presented a dynamic load-balancing framework that uses VM consolidation through four distribution methods. The proposed system implements three primary techniques for dynamic load balancing: VM selection according to imbalance level, BPSO for power optimization and host shutdown management, and Pearson correlation for maintaining SLA compliance. The proposed method decreases energy usage while diminishing SLA violations and decreasing VM migration frequency.

L. M. Al Qassem et al.[23] Two-state Random Forest (RF) machine learning models work proactively to predict future micro service workload requirements of CPU and memory resources. The predicted values from this process enable a dual adjustment of the resource pool by modifying hardware resources vertically while deploying additional micro-service replicas across horizontal axes.

G. Senthilkumar et al. [24] The authors developed a virtual machine allocation strategy that combines the approaches of the Random Forest (RF) and Genetic Algorithm (GA). Random Forest belongs to supervised machine learning techniques. This strategy makes Minimized energy usage possible by enabling better resource management(RM) for optimal utilization. A procedure exists for generating training data that enables random set training. The method is tested using real-time workload traces from Planet Lab. The proposed (GA-RF) model demonstrated better performance than other data center utilization techniques combined with host RM, power usage, and execution duration. The presented work uses resource usage, energy consumption, and execution time as performance metrics.

M. S. Al Reshan et al.[25] presented a combined GWO-PSO approach to improve system efficiency and RA in such a way that it takes advantage of the benefits of rapid convergence and global optimization. Research demonstrates that reaction times reached 12% below all competing algorithms in traditional methodology examination.

S. Singh et al. [26] presented a strategy, "Energy-Aware

Resource Allocation(RA) via MS-SLnO in Cloud Data," in which RA is achieved with the help of optimization to enhance the efficiency of cloud service. The method utilizes k-means clustering for task grouping. It deploys MS-SLnO as a hybrid bio-inspired optimizer that unites SLnO with MSA to find optimal resource allocations with PUE, which is included as a performance factor, execution time, and CPU usage factored into the decision. Through this approach, cloud data centers minimize their power consumption and reach maximum resource efficiency.

D. Kavitha et al.[27] Adaptive Deep Belief Network (ADBN) presents an adaptive deep belief network solution that incorporates Bayesian Search with Lichtenberg Optimization (BSI-LO) hybrid technique to address hardware limitations from resources, extended bandwidth delays, and slow performance of constrained devices. The offloading and RA methodologies achieved their best performance through this approach. The method provides an intelligent task offloading decision while achieving QoS and energy consumption balance.

Syed Mustapha et al.[28] proposed a method to enhance RA and task scheduling using (DBSCAN) algorithm, which is based on grouping virtual resources (VMs) according to their historical performance. This method reduced execution time and average start and finish time.

I. Jaya et al.[29] presented an adaptive deep reinforcement learning (DRL)-based method with adaptable capabilities deployed for edge-cloud gaming resource administration, reducing prices and enhancing allocation performance. The solution sought to achieve an equilibrium between cost expenditure and operational performance. The DRL-based algorithm reduces cumulative costs, brief allocation time, and large-scale performance capabilities.

H. Lin et al.[30] presented a two-level VM consolidation (TLHVMC) method that addresses the power mode transition (PMT) overhead requirements. The RA implemented a hybrid heuristic placement strategy based on a greedy strategy to achieve efficient resource placement (GSP). Time-based power mode control approaches are centralized management systems that control host power states. Wind power control strategies delivered higher energy efficiency through idle host suspension while using host state observations during a period to decrease SLA non-compliance.

Ahmad Raza Khan. [31] implemented dynamic load balancing through deep learning concepts, which unite CNNs and RNNs to derive load statistics for each VM. The clustering efficiency improves through the implementation of the Reinforcement Learning (RL) method together with the advanced Hybrid Lyrebird Falcon Optimization (HLFO) algorithm. The system dynamically divides VMs between overloaded and underloaded clusters, which leads to better resource use and reduced makespan durations.

O. K. J. Mohammad et al.[32] The researchers introduced Cloud Linear Regression (CLR) as their innovative machine learning framework, which integrates linear regression with cloud technology. CLR is a cloud-based enhancement that improves RM capabilities through better scheduling systems, provisioning, and allocation.

J. Ma et al.[33] The proposed study develops a Q-learning reinforcement learning method as a dynamic routing approach that adjusts network choices based on real-time conditions. The method decreases the root mean square error (RMSE) while enhancing load distribution to maximize resource usage.

C. Vijaya et al.[34] The proposed optimization method FCSFFC unites Cuckoo Search and Firefly Colony Optimization under fuzzy logic control for efficient cloud computing virtual machine migration. This technique improves RM and reduces superfluous migration instances for higher performance.

Table (1) below shows a summary comparison to highlight the main Proposed/Used Algorithm, metrics, results, strengths and weaknesses of the approaches.

From the comparative analysis in Table 1, key insights are observed : Most clustering-based methods (e.g.,[20],[21],[26],[28],[30]) are either static or rely on historical metrics or fuzzy clustering logic, which limits their real-time adaptability. For instance, fuzzy clustering[20] and bird-flocking-inspired clustering[21] yield good grouping but suffer from parameter uncertainty and poor adaptation to workload variation. Bio-inspired and deep learning strategies work well ([22]-[25],[27],[29],[31]-[34]) but often suffer from high resource demand, complexity, or convergence issues. Density-based techniques like DBSCAN([28]) struggle with high-dimensional resource heterogeneity and dynamic changes. Few works (e.g.,[26]) use simple K-Mean but lack justification for cluster initialization or adaptation to VM heterogeneity. Given the above limitations, a clear gap exists in developing a lightweight, adaptive, and resource-aware clustering mechanism that balances tasks in heterogeneous cloud environments. To address this, the proposed method(KMPPVM-DRM) introduces KMeans++, which improves KMean by using a stronger way to choose centroids, which results in both faster results and easier separation of clusters. This is especially beneficial for adaptive load balancing in heterogeneous VM environments based on real-time resource attributes (CPU, RAM).

3 The proposed approach

This section explains the proposed approach named A KMeans++ -Based Dynamic Clustering Approach for Virtual Machine Resource Allocation in Cloud Data Centers (KMPPVM-DRM) in the CC environment. The main objective of this approach focuses on delivering quality services to clients while decreasing total execution time, start times, and finishing times and establishing CDC workload equilibrium through resource allocation(RA) and scheduling mechanisms. The proposed is an advanced model based on machine learning algorithms for dynamic RA and scheduling that adapts to the heterogeneous CC environment. The proposed method (KMPPVM-DRM) consists of

Table 1: Comparative summary of related works

References, Year	Proposed/Used Algorithm	LB Metrics	Results	Strengths	Weaknesses
Q. Shang.[20], (2021)	Dynamic RA based on Workflow and Resource Clustering	Computing capability, cost, reliability	Effective in reducing completion time and resource utilization	Balanced allocation	Parameter uncertainty in fuzzy clustering leads to suboptimal results
S. El Motaki et al.[21], (2021)	Clustering Based on Bird Flocking Behavior	CPU usage, stall events	Effective clustering, improved resource usage	Fast, nature-inspired grouping	Reduced adaptation to transaction changes
J. P. B. Mapetu et al.[22], (2021)	Dynamic LB using VM Consolidation	Power, VM migrations, SLA	Reduced energy, SLA violations, VM migrations	SLA-aware, power-efficient	VM migration affects SLA and stability
L. M. Al Qassem et al.[23], (2023)	Proactive Autoscaling (Random Forest)	CPU, memory usage	Improved RA, reduced SLA violations	Predictive autoscaling	Historical-data based accuracy, slow response to workload changes
G. Senthilkumar et al.[24], (2023)	Hybrid Strategy (RF + GA)	Energy, resource usage	Reduced energy, improved utilization	Hybrid optimization	GA delays performance when optimization starts
M. S. Al Reshan et al.[25], (2023)	Combined GWO-PSO	Response time	Reduced response time by 12%	Fast convergence	Premature convergence complicates optimization
M. S. Al Reshan et al.[26], (2023)	Energy-Aware RA (MS-SLnO)	Power, CPU, execution time	Reduced power, optimized utilization	Hybrid optimizer	Requires calibration, affecting time and efficiency
D. Kavitha et al.[27], (2023)	Adaptive DBN (BSI-LO)	QoS, energy consumption	Improved QoS-energy balance	Intelligent task offloading	High resource demand, complexity, slow convergence
Syed Mustapha et al.[28], (2023)	DBSCAN-based Grouping	QoS	Improved QoS (13%), exec. time, start time (49%)	Effective historical grouping	Poor adaptation to dynamic changes
I. Jaya et al.[29], (2024)	Scalable DRL for Edge-Cloud Gaming	Cost, performance	Lower cost, high scalability	Smart dynamic learning	Needs large data and high computation
H. Lin et al.[30], (2024)	Two-Level VM Consolidation (GSP)	Power transitions, SLA	Energy savings, less SLA violations	Strategic placement	SLA violations due to PMT overhead
Ahmad Raza Khan.[31],(2024)	DL(CNNs, RNNs, RL)	Makespan, utilization	Improved utilization, lower makespan	Robust hybrid model	High resource/time cost of hybrid DL
O. K. J. Mohammad et al. [32],(2024)	Cloud Linear Regression (CLR)	CPU, Memory, Disk, Time	CPU: 59.8%, Mem: 72%, Disk: 90%, Time reduced	Smart VM selection, scalable	Limited to private settings, no benchmark
J. Ma et al. [33], (2025)	Cuckoo Search Algorithm (CA)	Time, Utilization, Scalability	Time: 2063ms, RR Time: 43088ms, LB: 95%	Scalable, global search	Needs tuning, poor under low resources
C. Vijaya et al.[34], (2024)	FCSFFC (Fuzzy Cuckoo + Firefly)	Energy, Migration, Computation	Improved energy, migration, load, computation	Fuzzy + bio-inspired exploration	Increased real-time complexity, needs fuzzy tuning

the following:

A. Initially, when tasks arrive at the broker, they are classified based on their length property, assuming a scenario with three different task length levels: high, medium, and low. Suitable virtual machines (VMs) are then allocated to them.

B. Assign initial weights to virtual machines based on their characteristics(CPU, RAM) from heterogeneous physical machines in a CC environment. The weights are calculated using the following Equation (1) :

$$VM_w = \frac{(\alpha * Cpu_{vm}) + (\beta * Ram_{vm})}{100} \quad (1)$$

The VM_w refers to the calculated VM weight, and both symbols(α, β) in Equation (1) represent the relative impact weights of the virtual machine characteristics. Where (α) is the relative impact weight of the CPU and the symbol (β) is memory (RAM) in the calculation.These values (α = 0.6) and (β = 0.4) were chosen to represent the higher significance of CPU than RAM in situations related to cloud computing. Also, (Cpu_{vm}) in Equation (1) refers to the CPU resources available and (Ram_{vm}) refers to the available RAM resources in the VMs. Weights for the virtual machines are stored in the weight map containing the VM number, the current available CPU, RAM, and the current machine weight. These weights are updated in real time. The calculated weight for the VM reflects the importance and amount of resources allocated based on the current (CPU, RAM) characteristics and the specified impact factors. These weights are used to help cluster the VMs.

C. Based on the weights assigned to virtual machines, these machines are clustered into several levels of

availability (high, medium, low) so that the clustering process is done using (unsupervised) machine learning through the updated KMeans++ algorithm and based on real-time dynamic weight map data within the data center of the cloud computing infrastructure. Virtual machines are divided into k clusters in KMPPVM-DRM using KMeans++ as the clustering method. Here are the essential features of the algorithm:

- Number of Clusters (k): This study sets The number of clusters fixed to k = 3 since it reflects three distinct degrees of VM capability. These levels align with the classification of tasks(long, medium, short).

- Centroid initialization Strategy: Centroids are chosen for KMeans++ using the VM weights in a probabilistic fashion. Thanks to this method, the solutions have a better distribution and reach convergence faster.

- Distance Metric: The algorithm uses an Euclidean distance measurement based on the VM weight value, including CPU and RAM characteristics. It allows for good grouping according to how many resources are available.

Then, each virtual machine with the highest weight(i.e., the VM currently having the highest available resource capacity based on real-time CPU and RAM weighted characteristics) each clustering is assigned to the corresponding tasks, as defined in Equation (2) :

$$VM_{assigned} = argmax_{vm_i \in C_j}(W_{vmi}) \quad (2)$$

Where ($VM_{assigned}$) the virtual machine selected for the next task. (VM_i) a virtual machine in the cluster C_j. (C_j) the cluster containing the virtual machines. W_{VMi} The weight of the virtual machine VM_i, representing its optimal capacity or suitability for the task. (argmax) The function that returns the argument (virtual machine) with the maximum weight. The selection approach ensures that the VM with the highest weight, reflecting its real-time available CPU and RAM capacities according to the specified impact factors (α and β), is prioritized for task execution. This helps to minimize execution time, avoid overloading less capable VMs, and load balancing in the heterogeneous cloud computing environment. Aftre the calculation of virtual machine weights basedon Equation(1) and the selection mechanism described in Equation(2),the notations and parameters involved in the proposed KMPPVM-DRM model are summarized fro clarity in Table 2.

D. The classified tasks are assigned to the optimal VMs from each cluster as follows:

(a) The task with the most extended length is directed to the optimal VM in the cluster with a high weight level of currently available resources.

Table 2: Notation description of the proposed model

Symbol	Description
α	The relative impact weight of the CPU resource (assigned value 0.6)
β	The relative impact weight of the RAM resource (assigned value 0.4)
VM_w	The computed weight of a virtual machine
Cpu_{vm}	The CPU resource capacity of a virtual machine (in MIPS)
Ram_{vm}	The RAM resource capacity of a virtual machine (in MB)
$VM_{assigned}$	The virtual machine selected for the incoming task
VM_i	A specific virtual machine within cluster C_j
C_j	The cluster containing the virtual machines
W_{VMi}	The weight value of virtual machine VM_i
arg max	The function selecting the VM with the maximum weight value

(b) The task with the medium length is assigned to the optimal machine in the cluster with the medium weight level of currently available resources.

(c) The task with the shorter length is assigned to the optimal machine in the cluster with the low weight level of currently available resources.

As tasks assigned to the optimal VMs in each cluster start to execute, the weight map of the VMs is dynamically updated during the current process. The KMPPVM-DRM approach is applied based on the updated weights, and this process is repeated continuously until all the required tasks are completed, as shown in the following Figure (2). Overall, the proposed approach supports dynamic allocation and scheduling of resources to provide efficient load balancing in resource utilization and improve overall system performance.

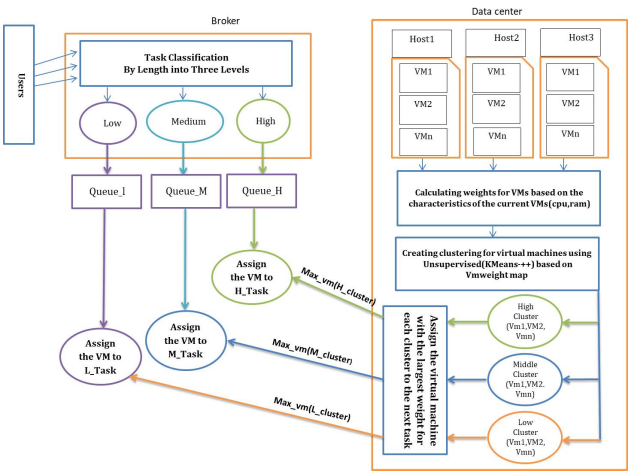


Figure 2: Diagram of proposed model approach

Detailed steps in the proposed KMPPVM-DRM scheduling and RA method are shown in Figure 3 (Flowchart). At the start, each cloudlet is classified by its length into

three categories: short, medium, and long. Then, the algorithm gives each virtual machine a dynamic weight based on the weighted values of both its CPU and RAM. All these weights are placed in a VM Weight Map to represent the current performance state of each VM. The map classifies the VMs into three categories (high, medium, and low) using the KMeans++ algorithm. For every cluster, the VM with the highest weight is assigned to deal with its group of cloudlets. Asks are then assigned accordingly to maximize resource utilization and efficiency. The algorithm for each part of the job calculates the cloudlet start time, finish time, and execution time. VM weights are updated during the scheduling cycle, considering both the state of the system and the results from execution. Once every task is done, the average quality of service metrics is calculated. Otherwise, the process iterates, adjusting the weights to keep the division of tasks responsive and balanced even as new demands appear.

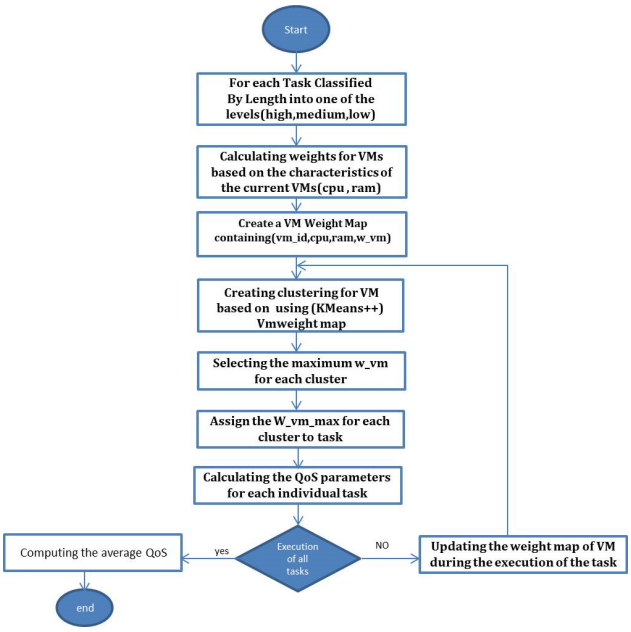


Figure 3: Flowchart of the proposed approach (KMPPVM-DRM)

The objective behind this approach consists of obtaining optimal resource allocation and performance enhancement for the system. The proposed KMPPVM-DRM approach contains a pseudocode to illustrate the formulas, parameters, and decisions involved in this approach.

The proposed KMPPVM-DRM approach was tested on two scales (e.g., 5 and 10 VMs and from 1000 to 10000 tasks). Although the current experiments are limited to these ranges, the clustering and weight-based selection mechanisms are designed to support scalability. KMeans++ adjusts to the number of virtual machines because it is an unsupervised algorithm.

Algorithm 1 The proposed approach (KMPPVM-DRM)

Input:

- List of Cloudlets (Tasks)
- List of VMs with characteristics
- Clustering parameter $K = 3$

Output: Cloudlets assigned to VMs, updated VM resource data, clustering results

foreach *cloudlet* t_i *in TaskList* **do**
  Classify t_i into {High, Medium, Low} based on task length
while *not all tasks are executed* **do**
  foreach *VM* v_j *in VMList* **do**
    Compute VM weight: $VM_Weight_j = (\alpha \cdot CPU_j + \beta \cdot RAM_j)/100$ Store VM_Weight_j in $VMWeightMap$
  Apply KMeans++ clustering on $VMWeightMap$ to produce 3 clusters C_1, C_2, C_3
  foreach *cluster* C_k **do**
    Select VM with highest weight: $VM_{\max} = \arg\max_{v \in C_k} W(v)$ Store $VM_{\max}$ in $VMmaxList$
  foreach *task* t_i **do**
    Assign t_i to appropriate $VM_{\max}$ based on task level (High, Medium, Low)
  Execute assigned tasks on selected VMs
  foreach *executed task* t_i **do**
    Record execution time,start and finish time
  if *all tasks completed* **then**
    Compute and output average QoS(start and finish time) metrics **break**
  else
    Update VM weights dynamically to reflect current loads **continue**

The weight calculation formula and dynamic updating allow the method to extend to larger numbers of VMs and handle varied task lengths or multiple user requests. Future work will involve stress testing with more VMs (e.g., 50–100), the tasks, and concurrent user simulations. Also, The KMPPVM-DRM model's primary emphasis is on clustering for managing resources, while studies such as [32],[33], and [34] emphasize that Cloud Linear Regression (CLR) can help estimate resources and decide on tasks. The methods mentioned above are not part of this model, but we also see future work that combining clustering and regression might help allocate VMs more efficiently.

4 Environment and results

This section presents the experimental testing environment of the proposed model and provides a comprehensive analysis of the obtained results.

4.1 Experimental environment

4.1.1 Hardware configuration

The experiments were conducted on a personal computer with the processor, memory, and operating system specifications shown in Table(3). The purpose of such a configuration is to demonstrate the lightweight and cost-effective nature of the method, which does not require high-performance computing resources to achieve efficient results. Although advanced load balancing techniques such as deep learning often require high-performance GPUs and extensive memory.

Table 3: Hardware requirements

Component	Specification
Operating System	Windows(X64 based Processor)64-bit OS
Processor	Intel® Core™ i7-CPU@2.6 GHz
RAM	16.0 GB

4.1.2 Software and tools

Researchers who need to evaluate cloud performance and model cloud solutions use CloudSim as their primary simulation tool to avoid reliance on computing facilities and reduce costs. CloudSim is an importable tool that functions within programming platforms, including Eclipse and NetBeans IDE, and it enables Windows 10 users to create cloud simulations. Entities and computing resources are modelled by default to reflect the allocation and scheduling scenario in a cloud environment to evaluate the proposed method. The simulation is performed on Cloudsim 3.0.3. In this work, CloudSim version 3.0.3 was selected Because of its good performance stability, the ability to use Java-based add-ons, and the ability to use external Java libraries for clustering (for example, Smile). It suits the proposed model, ensuring compatibility with machine learning and reproducibility across similar research. A single data center with 2 hosts each and 5 and 10 virtual machines is used for the simulation. Table 4 lists the configurations of all the submitted tasks (cloudlets), which are separated by length into three groups. The values for these simulations were picked to show the same level of diversity found in other cloud task scheduling studies, for instance,[28]. Specifically, Short tasks (300) represent lightweight user requests. Medium tasks (2000–3000) reflect average processing needs. Long tasks (4000) simulate compute-intensive applications such as data analytics. Thanks to these classifications, we can see how the evaluation of load-balancing algorithms progresses under mixed workloads with varying execution complexities.

Table 4: Parameters of user tasks

Parameters	Values
Task length	300 , 2000/3000, 4000
Input File Size	200 Byte
Output File Size	400 Byte
PE	1-2

Table 5 lists Small, Medium and Large VM categories and their MIPS, RAM and PEs. These configuration values are adopted from Syed Mustapha et al.[28] to maintain consistency with existing benchmarks. The proposed KMPPVM-DRM method makes use of these configurations. In particular, the method splits VMs into low, medium and high resource categories using KMeans++, which are based on each VM's weight for CPU and memory. Because of this classification, the cloud environment can assign work by size and required resources, leading to balanced utilization of all resources in real-time.

Table 5: Type of VMs

	Mips, Ram(MB), Pe	VM category
VMs	(1000,512,1)	Small
	(2000,1024,2)	Medium
	(4000,2156,4)	Large

4.2 Performance metrics

For evaluating the proposed KMPPVM-DRM, this paper relies on different performance metrics to measure its performance through these specific evaluation metrics:

A) Execution Time (ExT): The time needed to perform tasks within a cloud environment on virtual machines (VMs) constitutes Execution Time (ExT) [35]. The VM performance relies on this metric as a primary computation measure because performance improvement requires lowering this metric value. It uses the approach outlined in [28] to estimate the execution time of a task as described by Equation (3) in this formulation.

$$ExT_i = \frac{Tasklength(i)}{NO.ofProcessor * VM(MIPS(j))} + NetworkDelayi \quad (3)$$

$Tasklength(i)$ indicates the number of instructions to be executed for $Task_i$ while while (No.of Processor) represents the number of processors(PEs) allocated for each task, $VM(MIPS_j)$ measures the processing speed of VM_j expressed in MIPS – Million Instructions Per Second, and $NetworkDelay_i$ represents all possible system delays, including dependency, resource, and data transfer times.

B) Start Time (ST) and Finish Time (FT): In cloud computing, a task's start time and finish time refer to when the task starts executing and the execution process is completed, respectively. These points are essential for scheduling, LB, and RA. ST is when a resource (such as a virtual machine or container) is allocated to the task and begins executing. (FT) The moment at which the task completes its execution, including any data transfer, computation, or storage operations. The (FT) or completion time(CT) is measured using the Equation (4) below:

$$FT_i = ST_i + ExT_i + D_i \quad (4)$$

ST_i stands for start time, ExT_i stands for execution time, whereas D_i represents any delay caused by dependencies, resource queuing, and data transfer. All measurements are expressed in milliseconds(ms).

In this context, 'improvement' reduces the fall of these metrics: ExT, Average ST, and FT. Lower values for

all these metrics mean it operates more quickly, experiences fewer delays, and uses its resources better. Because of these improvements, using load balancing, QoS in the cloud is now more efficient.

4.3 Overall performance of the proposed model by varying the task count and VMs

In this section, the proposed (KMPPVM-DRM) algorithm performs evaluations against two different test cases: five VMs containing (1000 to 10000) cloudlets. Second: ten VMs containing (1000 to 10000) cloudlets. Increasing these variables can simulate reality and enhance the process of allocation and scheduling between different virtual machines for different cloudlets. The comparative analysis evaluated the proposed KMPPVM-DRM model against traditional and state-of-the-art algorithms.ACO[36] and PSO[37] acted as representative traditional metaheuristic optimization techniques because these two algorithms show broad application in classical resource scheduling problems. On the other hand, DBSCAN [28], a density-based clustering algorithm, was chosen for its recognized status as a state-of-the-art method in unsupervised learning due to its robustness in detecting clusters of arbitrary shapes and handling noise. This paper relies on performance metrics to measure its performance through these specific evaluation metrics:

1. Execution Time (ExT): Table 6 compares the performance of different models (PSO, ACO, DBSCAN, and KMPPVM-DRM) across various task counts and VMs. In the PSO model, execution time increases from 1478.47 (1000 tasks) to 13958.07(10000 tasks) and 10 VMs, indicating a proportional rise in execution time with an increasing task load. The ACO and DBSCAN follow similar trends, with execution time values escalating as the number of tasks grows. Notably, the proposed KMPPVM-DRM model stands out with significantly lower execution time values across all task counts (e.g., 307.75 for 1000 tasks), showcasing its superior efficiency in task execution time compared to the other models. Figure(4) shows the execution time of KMPPVM-DRM compared to the other models. The KMPPVM-DRM model's lower execution time suggests its potential for optimizing task scheduling and RA, resulting in faster task execution time and improved overall system efficiency.

Table 6: Shows execution time results while simulating 10 VMs measured in ms

Num of Tasks	PSO	ACO	DBSCAN	KMPPVM-DRM
1000	1478.47	1705.68	1080.168	307.75
3000	4177.67	4566.48	3240.378	975.25
5000	6960.87	8038.48	5400.624	1575.24
8000	11418.47	12774.48	8640.978	2475.24
10000	13958.07	16076.88	10801.22	3075.25

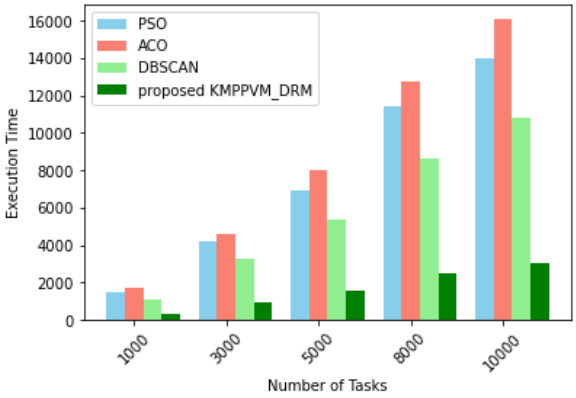


Figure 4: Execution time of KMPPVM-DRM compared to PSO, ACO and DBSCAN with 10 VMs

2. Finish Time : The average finish time values in the tables(7,8) are the average of moments at which the task completes its execution by various models (PSO, ACO, DBSCAN, and KMPPVM-DRM) in executing tasks across different counts and VMs. For PSO, the ACO, and the DBSCAN, there is a discernible upward trend in average finish time as the task count increases, indicating an increased finish time for handling larger workload scenarios, as shown in Figure (5,6). Specifically, DBSCAN's average finish time rises from 313.49 for 1000 tasks to 3103.79 for 10000 tasks with 5 VMs, rises from 219.44 for 1000 tasks to 2172.65 for 10000 tasks with 10 VMs. In contrast, the proposed KMPPVM-DRM model exhibits lower average finish time values across all task counts with VMs (5,10 VMs), underscoring its potential for energy-efficient task execution even with limited resources. For instance, KMPPVM-DRM achieves a notably lower average finish time of 123.24 for 1000 tasks with 5VMs and 61.93 for 1000 tasks with 10 VMs, positioning it as a promising model for scenarios prioritizing finish time in resource, task scheduling, and optimization.

Table 7: Shows the average finishing time with simulation 5 VMs in ms

Num of Tasks	PSO	ACO	DBSCAN	KMPPVM-DRM
1000	699.304	630.304	313.4951	123.24
3000	1926.053	1867.053	911.90058	223.49
5000	3173.058	3129.058	1536.658	499.16
8000	5029.683	4961.683	2413.843	935.47
10000	6321.599	6288.599	3103.799	1230.91

3. Start Time: Tables (9,10) show the average start time values when a resource (such as a virtual machine or container) is allocated to the task and begins executing by various models executing tasks across different counts and VMs. In the PSO model, the average start time increases from 654.315 (1000 tasks) to 6320.839(10000 tasks) for 5 VMs, from 219.44 for

Table 8: Shows the average finishing time with a simulation of 10 VMs in ms

Num of Tasks	PSO	ACO	DBSCAN	KMPPVM-DRM
1000	489.5128	441.2128	219.4466	61.93
3000	1348.237	1306.937	638.3304	117.98
5000	2221.14	2190.34	1075.66	337.58
8000	3520.778	3473.178	1689.69	742.35
10000	4425.119	4402.019	2172.659	1027.27

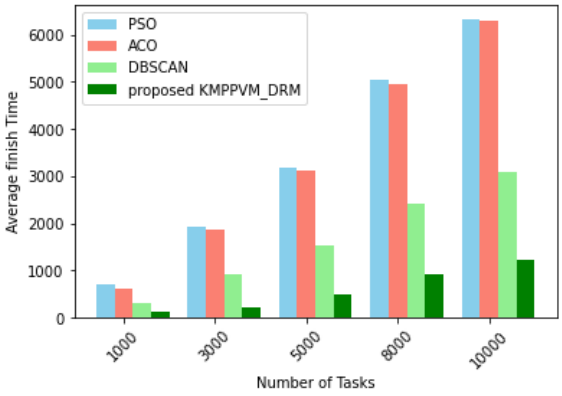


Figure 5: The finish time average for tasks performed by 5 VMs

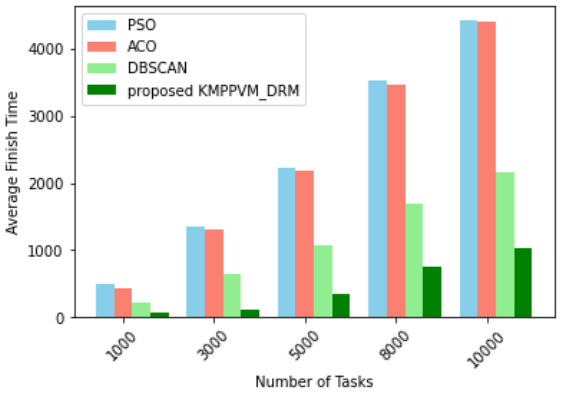


Figure 6: The finish time average for tasks performed by 10 VMs

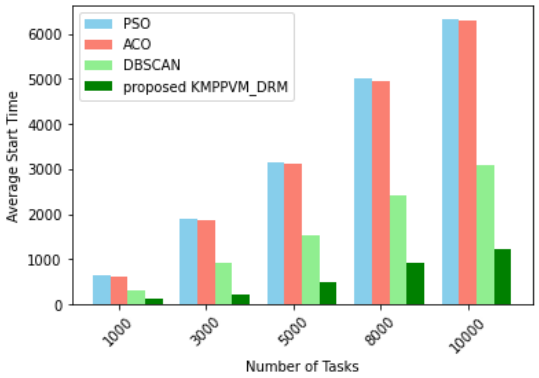
1000 tasks to 2172.65 for 10000 tasks for 10 VMs, indicating a proportional rise in average start time with an increasing task load. The ACO and DBSCAN follow similar trends, with start time average values escalating as the number of tasks grows with (5,10)VMs. Notably, the proposed KMPPVM-DRM model stands out with significantly lower average start time values across all task counts (e.g., 122.87 for 1000 tasks with 5VMs and 61.56 for 1000 tasks with 10 VMs), showcasing its faster task execution response and directly contributing in early RA and efficient scheduling compared to the other models. Figures (7a and 7b) show the average start time of KMPPVM-DRM compared to the other models.

Table 9: Shows the start time average with simulation 5 VMs in ms

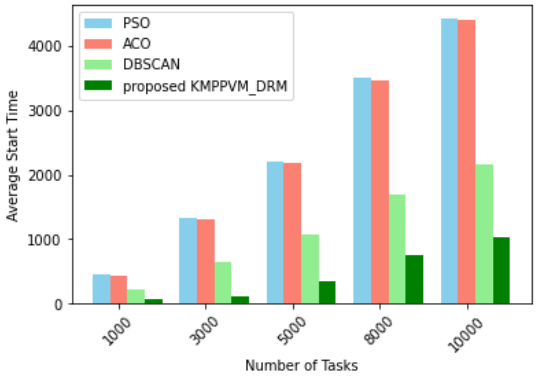
Num of Tasks	PSO	ACO	DBSCAN	KMPPVM-DRM
1000	654.3151	625.31518	308.1501	122.87
3000	1906.083	1862.083	906.5555	222.95
5000	3135.073	3124.073	1531.313	498.72
8000	5002.0138	4956.013	2408.498	935.09
10000	6320.839	6282.839	3098.454	1230.54

Table 10: Shows the start time average with a simulation of 10 VMs in ms

Num of Tasks	PSO	ACO	DBSCAN	KMPPVM-DRM
1000	458.0206	437.72068	215.7051	61.56
3000	1334.258	1303.458	634.5889	117.44
5000	2194.551	2186.851	1071.919	337.13
8000	3501.409	3469.209	1685.949	741.962
10000	4424.587	4397.987	2168.918	1026.9



(a) The start time average for tasks performed by 5 VMs.



(b) The start time average for tasks performed by 10 VMs.

Figure 7: Comparison of the start time average under different VM configurations: (a) 5 VMs, (b) 10 VMs

4.4 Performance analysis

To understand how the proposed algorithm's performance is affected by the amount of available resources, a set of experiments was conducted using two different virtual machine (VM) configurations: one with 5 VMs and the other with 10 VMs, with varying task loads (from 1,000 to 10,000 tasks). This analysis aimed to evaluate the al-

gorithm's adaptability to environments with limited versus larger resources and assess the consistency of the improvement in task execution speed compared to the DBSCAN, ACO, and PSO algorithms. Two analytical techniques were adopted to measure the performance of the proposed algorithm (KMPPVM-DRM): Relative Reduction to estimate the percentage of improvement relative to other algorithms and harmonic mean to find a fair estimate.

4.4.1 Performance analysis with 10 VMs

Table (11) shows the relative reduction in execution time that the suggested algorithm offers compared to others and for various tasks of various sizes. It is noted that the proposed algorithm significantly outperformed all other algorithms, the improvement over the DBSCAN algorithm ranging from 69.90% to 71.52%, compared to the ACO algorithm ranging from 78.64% to 81.95%, and compared to the PSO algorithm ranging from 76.65% to 79.18%. To

Table 11: Shows relative reduction percentages (%) in execution time achieved by KMPPVM-DRM compared to other algorithms, with 10 VMs

Number of Tasks	DBSCAN	ACO	PSO
1000	71.50	81.95	79.18
3000	69.90	78.64	76.65
5000	70.83	80.40	77.37
8000	71.35	80.62	78.32
10000	71.52	80.87	77.96

have an unbiased analysis of the proposed method's performance, especially with significantly varying values, the harmonic mean of the execution time of each method for various task counts was worked out. The harmonic means of the results are summarized in the following table(12).

Table 12: Shows the Harmonic mean of each algorithm's execution time with 10 VMs

Method	Harmonic Mean (ms)
PSO	4102.99
ACO	4672.25
DBSCAN	3071.51
KMPPVM-DRM (Proposed)	**886.72**

The KMPPVM-DRM algorithm achieved the lowest harmonic mean, demonstrating more efficient and consistent performance than the other algorithms, even as the number of tasks increased. This supports the reliability of the relative reduction results in reducing the total execution time and highlights the effectiveness of the proposed method in managing cloud computing resources. Figure (8) shows the harmonic means of KMPPVM-DRM compared to the other models. The results showed that the KMPPVM-DRM algorithm significantly outperformed the other algorithms in reducing both the average finish time and the average start

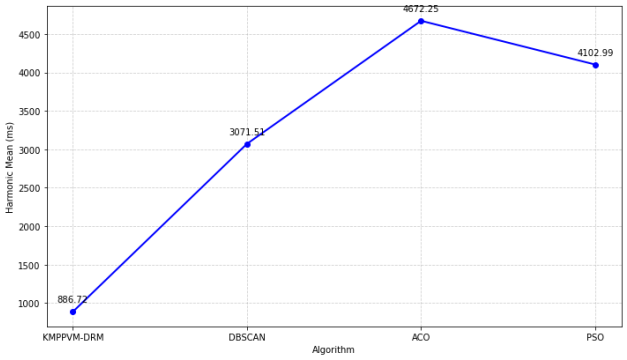


Figure 8: Comparison of Harmonic Means of Algorithms with 10 VMs

time with 10 virtual machines, reflecting the model's efficient use of available resources. Figures (9,10) illustrate the comparative performance regarding average finish and start times, respectively. Furthermore, Tables (13,14) provide a detailed summary of the relative reduction percentages of average finish and start times achieved by the proposed method across different task sizes.

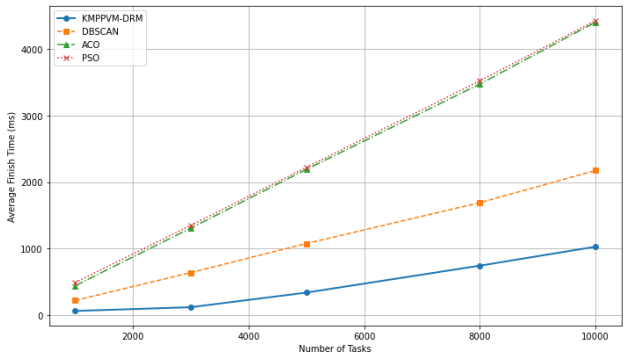


Figure 9: comparative performance for average finish time of Algorithms with 10 VMs

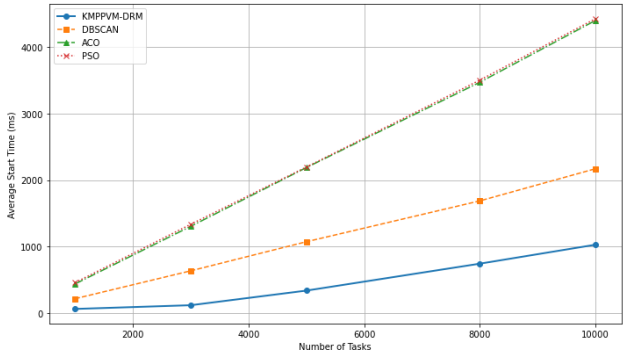


Figure 10: comparative performance for average start time of Algorithms with 10 VMs

Table 13: Shows relative reduction percentages (%) in average finish time achieved by KMPPVM-DRM compared to other algorithms

Number of Tasks	DBSCAN	ACO	PSO
1000	71.77	85.96	87.34
3000	81.51	90.97	91.24
5000	68.61	84.58	84.80
8000	56.06	78.62	78.91
10000	52.71	76.66	76.78

Table 14: Shows relative reduction percentages (%) in average start time achieved by KMPPVM-DRM compared to other algorithms

Number of Tasks	DBSCAN	ACO	PSO
1000	71.46	85.93	86.55
3000	81.49	90.99	91.19
5000	68.54	84.58	84.63
8000	55.99	78.61	78.80
10000	52.65	76.65	76.79

In this table(13), It is noted that the proposed algorithm significantly outperformed all other algorithms in the relative reduction percentages for the average finish time, the improvement over the DBSCAN algorithm ranging from 52.71% to 81.51%, compared to the ACO algorithm ranging from 76.66% to 90.97%, and compared to the PSO algorithm ranging from 76.78% to 91.24%. Table (14) also presents the relative reduction percentages of the average start time, clearly highlighting the superiority of the proposed algorithm over the other algorithms. The improvement over the DBSCAN algorithm ranged from 52.65% to 81.49%, compared to the ACO algorithm, which ranged from 76.65% to 90.99%, and compared to the PSO algorithm, which ranged from 76.79% to 91.19%. Table (15) presents each algorithm's harmonic mean of the average start and finish times. The findings indicate that the proposed KMPPVM-DRM algorithm outperforms the others, recording the lowest harmonic mean values. It demonstrates its enhanced capability to reduce task waiting time and accelerate execution.

Table 15: Shows the harmonic mean of each algorithm's average finish and start times with 10 VMs

Method	Harmonic Mean (Start Time)	Harmonic Mean (Finish Time)
PSO	1282.03	1335.19
ACO	1242.34	1248.98
DBSCAN	609.84	616.88
KMPPVM-DRM	**166.41**	**167.19**

The following figures (11,12) show the harmonic mean of the average times (start and finish times). Figure (11) shows the harmonic mean of the task start average, while Figure (12) highlights the harmonic mean of the task finish average.

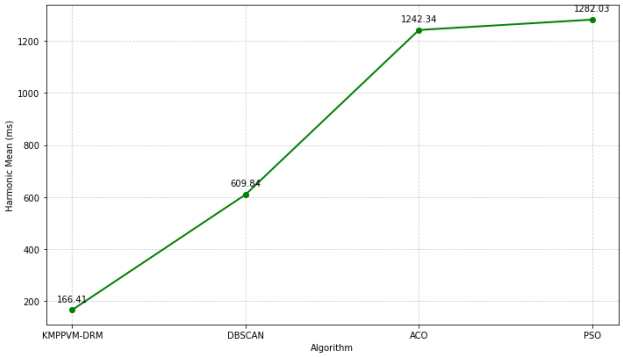


Figure 11: Comparison of harmonic means for average start time of algorithms with 10 VMs

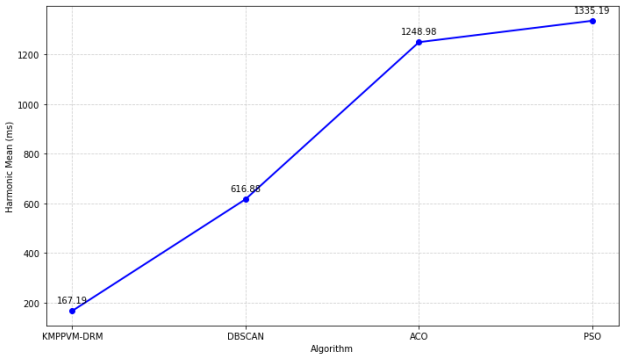


Figure 12: Comparison of harmonic means for average finish time of algorithms with 10 VMs

4.4.2 Performance analysis with 5 VMs

This table(16) shows that the relative reduction percentages are still high, albeit slightly lower than in the case where 10 virtual machines were used. This results from limited resources available at the system's disposal and a lot of pressure on the allocation algorithm to balance it. However, the proposed algorithm did not lose its superiority compared to the other algorithms, demonstrating its efficiency in adapting to the resource-limited environment.

Table 16: Shows relative reduction percentages (%) in average finish and start time achieved by KMPPVM-DRM compared to other algorithms, with 5 VMs

Method	Number of Tasks	Avg. Start Time (ms)	Avg. Finish Time (ms)
PSO	1000	81.22	82.37
	3000	88.30	88.39
	5000	84.09	84.26
	8000	81.30	81.40
	10000	80.53	80.52
ACO	1000	80.35	80.44
	3000	88.01	88.02
	5000	84.03	84.04
	8000	81.13	81.14
	10000	80.41	80.42
DBSCAN	1000	60.12	60.68
	3000	75.40	75.49
	5000	67.43	67.51
	8000	61.17	61.24
	10000	60.28	60.34

Table (17) presents each algorithm's harmonic mean of the average start and finish times. The findings indicate that the proposed KMPPVM-DRM algorithm outperforms the others, recording the lowest harmonic mean values with 5 VMs. It demonstrates its enhanced capability to reduce task waiting time and accelerate execution.

Table 17: Shows the harmonic mean of each algorithm's average finish and start times with 5 VMs

Method	Harmonic Mean (Start Time)	Harmonic Mean (Finish Time)
PSO	1831.47	1907.41
ACO	1774.77	1784.26
DBSCAN	871.20	881.26
KMPPVM-DRM (Proposed)	**302.82**	**303.52**

Compared to the 10-VM use case, the harmonic mean values for the average start and end times increased for all algorithms as resources were reduced, increasing the pressure on the hardware and leading to processing delays. However, the proposed algorithm (KMPPVM-DRM) maintained the best relative performance with the lowest harmonic mean value, confirming its high capacity and flexibility in resource-constrained cloud computing environments. The following figures (13,14) show the harmonic mean of the average times (start and finish times) with 5VMs. Figure (13) shows the harmonic mean of the task start average, while Figure (14) highlights the harmonic mean of the task finish average. The results revealed that KMPPVM-DRM produces vastly improved performance compared to other algorithms in the three metrics of the total execution time and average finish and start times while significantly reducing the execution time. The data also show that such improvements significantly increase with the increasing number of virtual machines, demonstrating the model's effective scalability.

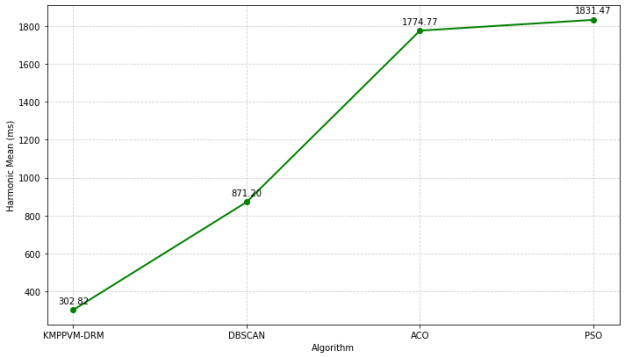


Figure 13: Comparison of harmonic means for average start time of algorithms with 5 VMs

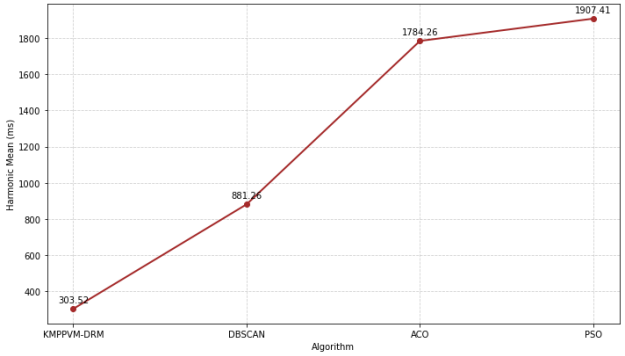


Figure 14: Comparison of harmonic means for average finish time of algorithms with 5 VMs

4.5 Statistical analysis using paired T-test (with Multiple VMs)

In this study section, a paired t-test was used to test for the statistical significance of the performance differences between the proposed algorithm (KMPPVM-DRM) and the other algorithms (DBSCAN, ACO, PSO) in two separate scenarios. The first has 5 virtual machines, and the second has 10. This analysis determines whether the observed differences across various task loads reflect genuine improvements or are simply due to random variation. The t-test results showed that the difference was significant at 0.05 level in both scenarios while supporting the effectiveness and flexibility of the suggested algorithm under a resource-limited environment. Table (18) presents the results of the conducted statistical analysis with a paired t-test for performance comparisons between the proposed KMPPVM-DRM algorithm and other algorithms based on three major performance indicators (execution time, average start, and finish time) in a computing environment comprising 10 virtual machines (VMs).

Table 18: Shows paired T-test results (t-statistic, p-value) for KMPPVM-DRM vs other algorithms on 10 VMs

Metric	Comparison	t-statistic	p-value
Execution Time	KMPPVM-DRM vs DBSCAN	-3.28	0.0303
	KMPPVM-DRM vs ACO	-3.27	0.0307
	KMPPVM-DRM vs PSO	-3.30	0.0297
Average Start Time	KMPPVM-DRM vs DBSCAN	-4.07	0.0152
	KMPPVM-DRM vs ACO	-3.56	0.0234
	KMPPVM-DRM vs PSO	-3.60	0.0227
Average Finish Time	KMPPVM-DRM vs DBSCAN	-4.09	0.0149
	KMPPVM-DRM vs ACO	-3.57	0.0233
	KMPPVM-DRM vs PSO	-3.66	0.0215

The results in Table (18) revealed that all p-values were smaller than 0.05, thus indicating the statistical significance of the successes of KMPPVM-DRM. This indicates the superiority and stability of the algorithm's performance on different metrics. Additionally, all comparisons' t-statistic values were negative, indicating that KMPPVM-DRM consistently outperformed the other algorithms regarding reducing time metrics. The figures(15,16,17) below also show the p-values of paired t-test results between KMPPVM-DRM and the other algorithms across three-time metrics: execution time, average finish time, and average start time. All p-values shown in the figures were less than the statistical significance level of 0.05, confirming the presence of statistical significance and superiority in favour of the proposed algorithm.

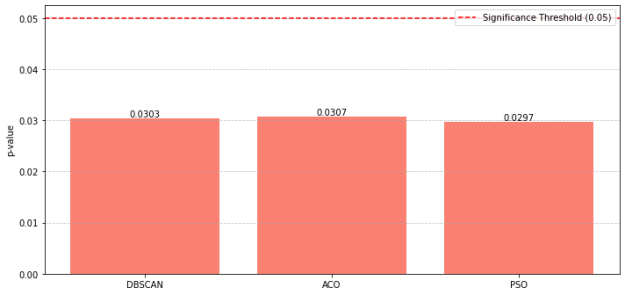


Figure 15: p-values of paired t-test results between KMPPVM-DRM and other algorithms for execution time with 10 VMs

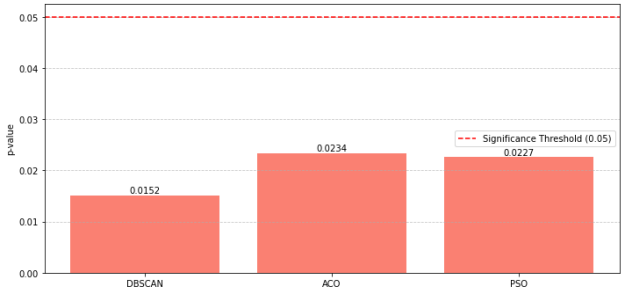


Figure 16: p-values of paired t-test results between KMPPVM-DRM and other algorithms for Average Start time with 10 VMs

In order to strengthen the assessment of the efficiency of the proposed KMPPVM-DRM algorithm, the paired t-tests were performed to compare the algorithm to other algorithms in terms of performance metrics (execution time, average finish time, and average start time) at resource-constrained settings (5VMs). The results of the paired t-tests in Table (19) indicate that the p-values have a significance level of less than 0.05, confirming that the observed improvements are statistically significant. On the other hand, negative t values show that KMPPVM-DRM had better performance metrics, demonstrating its efficiency and adaptability even in resource-constrained environments.

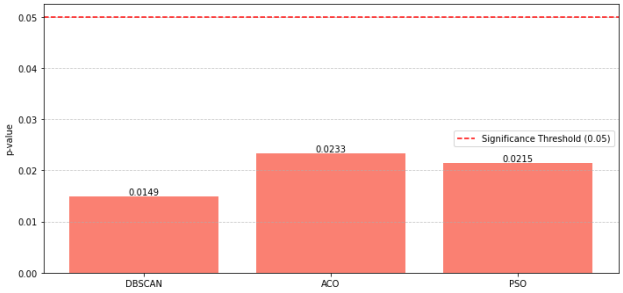


Figure 17: p-values of paired t-test results between KMPPVM-DRM and other algorithms for Average Finish time with 10 VMs

Table 19: Shows paired T-test results (t-statistic, p-value) for KMPPVM-DRM vs other algorithms on 5 VMs

Metric	Comparison	t-statistic	p-value
Execution Time	KMPPVM-DRM vs DBSCAN	-3.32	0.0292
	KMPPVM-DRM vs ACO	-3.25	0.0312
	KMPPVM-DRM vs PSO	-3.29	0.0302
Average Start Time	KMPPVM-DRM vs DBSCAN	-3.56	0.0235
	KMPPVM-DRM vs ACO	-3.40	0.0271
	KMPPVM-DRM vs PSO	-3.43	0.0263
Average Finish Time	KMPPVM-DRM vs DBSCAN	-3.58	0.0231
	KMPPVM-DRM vs ACO	-3.41	0.0270
	KMPPVM-DRM vs PSO	-3.49	0.0250

The figures(18,19) below also show the paired t-test results between KMPPVM-DRM and the other algorithms, and All p-values shown in the figures were less than the statistical significance level of 0.05, confirming the presence of statistical significance and superiority in favour of the proposed algorithm even with the environment resource-constrained (5VMs).

4.6 Discussion and interpretation

Compared to DBSCAN, ACO, and PSO, the KMPPVM-DRM algorithm ran faster, reduced execution time, and improved the average start and finish time and the distribution of jobs among processors. The boost in performance comes from using KMeans++ to cluster VMs, which helps choose much better centres and facilitates faster convergence. Unlike DBSCAN, which struggles with irregular cloud workloads and requires sensitive parameter tuning, ACO/PSO is time-consuming. KMPPVM-DRM clustering of VMs into groups (high, medium, low) and distributes tasks to the most appropriate VM in each group based on demand. The

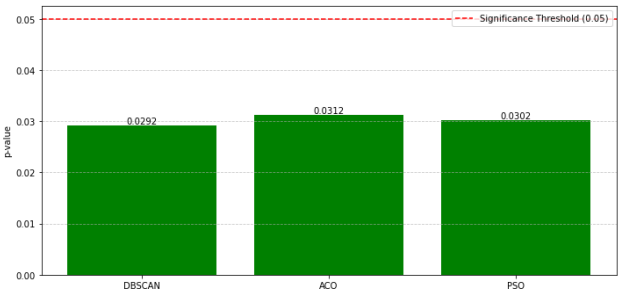
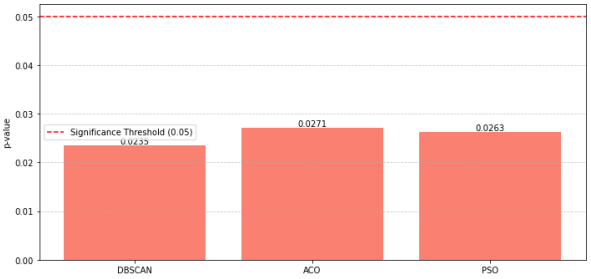
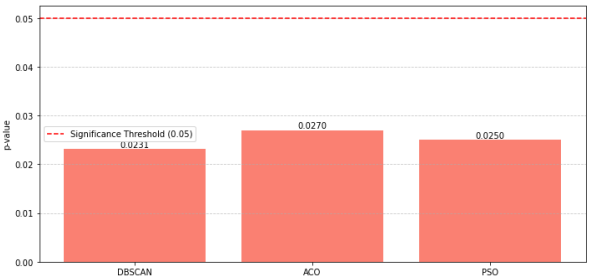


Figure 18: p-values of paired t-test results between KMPPVM-DRM and other algorithms for execution time with 5 VMs



(a) p-values of paired t-test to (average start time)



(b) p-values of paired t-test to (Average Finish time).

Figure 19: p-values of paired t-test results between KMPPVM-DRM and other algorithms: (a) average start time, (b) average finish time with 5 VMs

method uses KMeans++ to make strong VM groups, which offers better initialization, computational efficiency, and stability under real-time VM load variations and dynamic resource management (DRM) to manage the intelligent allocation of tasks. VMs are sorted by their capabilities, and specific tasks are assigned according to these groupings. Meanwhile, VM weights change in real-time with changes in the system. This enables improved scheduling decisions. Because KMPPVM-DRM has a smaller search area and less to worry about than ACO and PSO, it is more efficient regarding computational complexity. Nevertheless, handling massive cloud deployments and continually updating the weights can be difficult. However, these issues might be handled by parallel clustering techniques or leveraging distributed computing frameworks in the future. Overhead Consideration: While the current version does not explicitly measure the computation time of KMeans++, scheduler response time, or system latency, these aspects are acknowledged as potential overhead sources. Future experiments will include benchmarks for runtime overhead, especially under large-scale conditions, to quantify the impact of clustering and dynamic weight updates on overall system responsiveness.

5 Conclusion and future direction

This paper develops a system model that optimizes dynamic resource scheduling procedures through efficient cloud resource allocation(RA) with support for different cloud infrastructure deployments. Tasks are classified based on their length property, assuming a scenario with three different task length levels: high, medium, and low. For the allocation, a dynamic clustering mechanism(KMeans++) is used based on the current load weight of virtual machines (VMs) in real-time. In a cluster of machines, a ULC is used to identify the most suitable virtual machine from each cluster to be assigned to execute the task by finding the machine with the most significant weight of the available resources in the cluster. As shown in the results section, the proposed model outperforms the existing methods in (DBSCAN, ACO, and PSO). The performance evaluation relies on execution time measurements alongside start time and finish time averages. The result shows that the proposed method performs better than the existing algorithms in (DBSCAN, ACO, and PSO) for both execution time and start time and finish time averages for different numbers of tasks ranging from (1000 to 10000) with (5 and 10) virtual machines. The results validate the efficiency and adaptability of the proposed model for heterogeneous cloud infrastructures. Using 10 virtual machines, the proposed model had the best harmonic execution time of 886.72, outperforming PSO at 4102.99, ACO at 4672.25 and DBSCAN at 3071.51. Similarly, the harmonic means of average start and finish times were significantly lower at 166.41 and 167.19, respectively, whereas other methods exhibited values exceeding 600 and 1200. In terms of relative reduction percentages as an estimate of performance improvement, measured by reducing the execution time, was from 69.90% to 71.52% for the proposed model compared to DBSCAN, meaning the model is dependable no matter how many tasks it needs to deal with (up to 10000). Compared to DBSCAN, the reduction in finish time average was between 52.71% and 81.51%, and the start time average was from 52.65% to 81.49%. Moreover, even under limited resource conditions (with only 5 virtual machines), the proposed approach demonstrated competitive and stable performance compared to the baseline algorithms, highlighting its robustness and adaptability.

The proposed model will be extended in future work to support larger-scale environments with increased numbers of 100+ VMs, variable task inter-arrival rates, diverse task types, and VM failure scenarios. In addition, examining the

use of additional machine learning algorithms (may consider integrating predictive analytics, such as Cloud Linear Regression (CLR) models as introduced in [32], [33], and [34], to enhance the accuracy of resource demand forecasting) to optimize RA and load further balancing in cloud data centres. Finally, sensitivity analysis will be conducted on the CPU and RAM weight parameters (α , β) to identify optimal configurations for varying workload types and deployment scenarios.

References

[1] H. S. Malallah, R. Qashi, L. M. Abdulrahman, M. A. Omer, and A. A. Yazdeen, "Performance analysis of enterprise cloud computing: a review," *Journal of Applied Science and Technology Trends*, vol. 4, no. 01, pp. 01–12, 2023. https://doi.org/10.38094/jastt401139.

[2] S. Paul and M. Adhikari, "Dynamic load balancing strategy based on resource classification technique in iaas cloud," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 2059–2065, IEEE, 2018. https://doi.org/10.1109/icacci.2018.8554440.

[3] S. S. Gill and R. Buyya, "Resource provisioning based scheduling framework for execution of heterogeneous and clustered workloads in clouds: from fundamental to autonomic offering," *Journal of Grid Computing*, vol. 17, pp. 385–417, 2019. https://doi.org/10.1007/s10723-017-9424-0.

[4] A. Shahidinejad, M. Ghobaei-Arani, and M. Masdari, "Resource provisioning using workload clustering in cloud computing environment: a hybrid approach," *Cluster Computing*, vol. 24, no. 1, pp. 319–342, 2021. https://doi.org/10.1007/s10586-020-03107-0.

[5] O. H. Sultan and T. Khaleel, "Challenges of load balancing techniques in cloud environment: A review," *Al-Rafidain Engineering Journal*, vol. 27, no. 2, pp. 227–235, 2022. https://doi.org/10.33899/rengj.2022.134056.1179.

[6] R. Tasneem and M. Jabbar, "An insight into load balancing in cloud computing," in *International Conference on Wireless Communications, Networking and Applications*, pp. 1125–1140, Springer, 2021. https://doi.org/10.1007/978-981-19-2456-9_113.

[7] N. K. Mishra and N. Mishra, "Load balancing techniques: need, objectives and major challenges in cloud computing-a systematic review," *International Journal of Computer Applications*, vol. 131, no. 18, pp. 0975–8887, 2015. https://doi.org/10.5120/ijca2015907523.

[8] Y. Lohumi, D. Gangodkar, P. Srivastava, M. Z. Khan, A. Alahmadi, and A. H. Alahmadi, "Load balancing in cloud environment: A state-of-the-art review," *Ieee Access*, vol. 11, pp. 134517–134530, 2023. https://doi.org/10.1109/access.2023.3337146.

[9] A. M. Manasrah and H. Ba Ali, "Workflow scheduling using hybrid ga-pso algorithm in cloud computing," *Wireless Communications and Mobile Computing*, vol. 2018, no. 1, p. 1934784, 2018. https://doi.org/10.1155/2018/1934784.

[10] A. Senthil Kumar and M. Venkatesan, "Multi-objective task scheduling using hybrid genetic-ant colony optimization algorithm in cloud environment," *Wireless Personal Communications*, vol. 107, pp. 1835–1848, 2019. https://doi.org/10.1007/s11277-019-06360-8.

[11] U. K. Jena, P. K. Das, and M. R. Kabat, "Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 6, pp. 2332–2342, 2022. https://doi.org/10.1016/j.jksuci.2020.01.012.

[12] F. Davami, S. Adabi, A. Rezaee, and A. M. Rahmani, "Distributed scheduling method for multiple workflows with parallelism prediction and dag prioritizing for time constrained cloud applications," *Computer networks*, vol. 201, p. 108560, 2021. https://doi.org/10.1016/j.comnet.2021.108560.

[13] P. Paknejad, R. Khorsand, and M. Ramezanpour, "Chaotic improved picea-g-based multi-objective optimization for workflow scheduling in cloud environment," *Future Generation Computer Systems*, vol. 117, pp. 12–28, 2021. https://doi.org/10.1016/j.future.2020.11.002.

[14] Z. Mahmoudi, E. Darbanian, and M. Nickray, "Computational resource allocation in iot fog computing using teaching–learning-based optimization algorithm," *Journal of Soft Computing and Information Technology*, vol. 10, no. 3, pp. 73–85, 2021.

[15] A. Daghayeghi and M. Nickray, "Nsgaii-based task scheduling model for smart city applications in cloud-fog environment," *Journal of Soft Computing and Information Technology*, vol. 11, no. 3, pp. 64–82, 2022.

[16] S. A. Omranian and M. Goudarzi, "Greedy algorithm for dynamic allocation of intelligent services in vehicular edge computing," *Cluster Computing*, vol. 28, no. 1, pp. 1–20, 2025. https://doi.org/10.1007/s10586-024-04817-5.

[17] P. R. Kaveri and P. Lahande, "Reinforcement learning to improve resource scheduling and load balancing in cloud computing," *SN Computer Science*, vol. 4,

no. 2, p. 188, 2023. https://doi.org/10.1007/s42979-022-01609-9.

[18] M. R. Belgaum, S. Musa, M. M. Alam, and M. M. Su'ud, "A systematic review of load balancing techniques in software-defined networking," *Ieee Access*, vol. 8, pp. 98612–98636, 2020. https://doi.org/10.1109/access.2020.2995849.

[19] J. Shankar, I. Hussain, S. Zafar, I. R. Khan, and A. Khalique, "Effective resource allocation and load balancing in green cloud computing," in *International Conference on ICT for Digital, Smart, and Sustainable Development*, pp. 423–439, Springer, 2024. https://doi.org/10.1007/978-981-97-7831-7_26.

[20] Q. Shang, "A dynamic resource allocation algorithm in cloud computing based on workflow and resource clustering," *Journal of Internet Technology*, vol. 22, no. 2, pp. 403–411, 2021.

[21] S. El Motaki, A. Yahyaouy, H. Gualous, and J. Sabor, "A new weighted fuzzy c-means clustering for workload monitoring in cloud datacenter platforms," *Cluster Computing*, vol. 24, no. 4, pp. 3367–3379, 2021. https://doi.org/10.1007/s10586-021-03331-2.

[22] J. P. B. Mapetu, L. Kong, and Z. Chen, "A dynamic vm consolidation approach based on load balancing using pearson correlation in cloud computing," *The Journal of Supercomputing*, vol. 77, no. 6, pp. 5840–5881, 2021. https://doi.org/10.1007/s11227-020-03494-6.

[23] L. M. Al Qassem, T. Stouraitis, E. Damiani, and I. A. M. Elfadel, "Proactive random-forest autoscaler for microservice resource allocation," *IEEE Access*, vol. 11, pp. 2570–2585, 2023. https://doi.org/10.1109/access.2023.3234021.

[24] G. Senthilkumar, K. Tamilarasi, N. Velmurugan, and J. Periasamy, "Resource allocation in cloud computing," *Journal of Advances in Information Technology*, vol. 14, no. 5, pp. 1063–1072, 2023. https://doi.org/10.12720/jait.14.5.1063-1072.

[25] M. S. Al Reshan, D. Syed, N. Islam, A. Shaikh, M. Hamdi, M. A. Elmagzoub, G. Muhammad, and K. H. Talpur, "A fast converging and globally optimized approach for load balancing in cloud computing," *IEEE Access*, vol. 11, pp. 11390–11404, 2023. https://doi.org/10.1109/access.2023.3241279.

[26] S. Singh, P. Singh, and S. Tanwar, "Energy aware resource allocation via ms-slno in cloud data center," *Multimedia Tools and Applications*, vol. 82, no. 29, pp. 45541–45563, 2023. https://doi.org/10.1007/s11042-023-15521-8.

[27] D. Kavitha, M. Priyadharshini, R. Anitha, S. Suma, V. Prema, and A. Vidhya, "Adaptive dbn using hybrid bayesian lichtenberg optimization for intelligent task allocation," *Neural Processing Letters*, vol. 55, no. 4, pp. 4907–4931, 2023. https://doi.org/10.1007/s11063-022-11071-6.

[28] S. D. S. Mustapha and P. Gupta, "Dbscan inspired task scheduling algorithm for cloud infrastructure," *Internet of Things and Cyber-Physical Systems*, vol. 4, pp. 32–39, 2024. https://doi.org/10.1016/j.iotcps.2023.07.001.

[29] I. Jaya, Y. Li, and W. Cai, "Deep reinforcement learning based resource allocation in edge-cloud gaming," *Multimedia Tools and Applications*, vol. 83, no. 26, pp. 67903–67926, 2024. https://doi.org/10.1007/s11042-024-18337-2.

[30] H. Lin, G. Liu, W. Lin, X. Wang, and X. Wang, "A novel virtual machine consolidation algorithm with server power mode management for energy-efficient cloud data centers," *Cluster Computing*, vol. 27, no. 8, pp. 11709–11725, 2024. https://doi.org/10.1007/s10586-024-04555-8.

[31] A. R. Khan, "Dynamic load balancing in cloud computing: optimized rl-based clustering with multi-objective optimized task scheduling," *Processes*, vol. 12, no. 3, p. 519, 2024. https://doi.org/10.3390/pr12030519.

[32] O. K. J. Mohammad, M. E. Seno, and B. N. Dhannoon, "Detailed cloud linear regression services in cloud computing environment," *Informatica*, vol. 48, no. 12, 2024. https://doi.org/10.31449/inf.v48i12.6771.

[33] J. Ma, C. Zhu, Y. Fu, H. Zhang, and W. Xiong, "Dynamic routing via reinforcement learning for network traffic optimization," *Informatica*, vol. 49, no. 8, 2025. https://doi.org/10.31449/inf.v49i8.7126.

[34] C. Vijaya and P. Srinivasan, "Hybrid fuzzy metaheuristic technique for efficient vm selection and migration in cloud data centers," *Informatica*, vol. 48, no. 20, 2024. https://doi.org/10.31449/inf.v48i20.6549.

[35] M. A. Alworafi, A. Dhari, A. A. Al-Hashmi, A. B. Darem, *et al.*, "An improved sjf scheduling algorithm in cloud computing environment," in *2016 International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT)*, pp. 208–212, IEEE, 2016. https://doi.org/10.1109/iceeccot.2016.7955216.

[36] P. Gupta and S. P. Ghrera, "Trust and deadline aware scheduling algorithm for cloud infrastructure using ant colony optimization," in *2016*

International conference on innovation and challenges in cyber security (ICICCS-INBUSH), pp. 187–191, IEEE, 2016. https://doi.org/10.1109/icicccs.2016.7542337.

[37] X. Zuo, G. Zhang, and W. Tan, "Self-adaptive learning pso-based deadline constrained task scheduling for hybrid iaas cloud," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 2, pp. 564–573, 2013. https://doi.org/10.1109/tase.2013.2272758.

Dual-Stream CNN-GRU Model with Spatial and Self-Attention for Power Forecasting in Electrical Automation

Chengjiang Tang
Heilongjiang Vocational College of Agricultural Technology
Jiamusi, Heilongjiang, 154007, China
Email: hnzytcj@126.com

Keywords: Electrical automation, smart grid, power forecasting, hybrid model

Received: February 23, 2025

Accurate power forecasting is critical for electrical automation within smart grids, enabling intelligent energy management to balance demand and supply. We propose a dual-stream hybrid model that processes multivariate time series data from the DKASC (solar generation) and IHEPC (household consumption) datasets through parallel Convolutional Neural Networks (CNNs) with spatial attention and Gated Recurrent Units (GRUs) with self-attention. This unified model captures cross-variable correlations (e.g., weather, power output) and long-term temporal dependencies to predict both generation and consumption, enhancing grid stability. Evaluated using RMSE, MAE, and MSE, our model achieves a 32–93% reduction in RMSE compared to baseline methods like RCC-LSTM and CNN-LSTM hybrids on DKASC and IHEPC datasets. Designed for integration with electrical automation systems, it offers superior accuracy, robustness, and efficiency, advancing smart grid operations.

Povzetek: Članek predstavi dvojni tok CNN-GRU model z vgrajenima prostorsko in samo-pazljivostjo za napovedovanje proizvodnje in porabe elektrike. Model izboljšuje robustnost in učinkovitost pametnih omrežij.

1 Introduction

The evolution of traditional power grids into smart grids (SGs) has been boosted by advancements in information and communication technologies [1]. SGs are designed to deliver energy that is economical, reliable, secure, and sustainable, while simultaneously optimizing load distribution, monitoring grid performance, and balancing demand and supply. Accurate forecasting of both energy generation and consumption is fundamental for ensuring grid stability and effective scheduling [2]. Power data, a critical component in this process, is inherently a multivariate time series, comprising sequentially ordered data points across multiple interrelated variables. This data reveals two distinct types of patterns: local patterns that capture short-term variations and anomalies, and global patterns that highlight overarching trends in energy generation and consumption. These patterns serve as the foundation for reliable forecasting by encapsulating critical insights into the dynamics of the power grid for electrical automation.

Forecasting energy generation and consumption provides significant advantages for energy providers. By analyzing historical data, providers can predict future trends, enabling the seamless supply of energy to meet demand [3]. This involves not only identifying local and global patterns within time series data but also deciphering the complex inter-dependencies between them. Such insights facilitate proactive decision-making, allowing power suppliers to optimize energy production, minimize costs, and promote environmentally sustainable practices. For instance, a deeper understanding of power generation and consumption patterns supports utility companies in efficiently allocating resources and managing energy systems in smart cities. The interdependence between generation and consumption underscores the importance of unified models for electrical automation, which we define as models capable of jointly predicting both processes to capture their mutual influences. Our proposed dual-stream hybrid model, which integrates Convolutional Neural Networks (CNNs) with spatial attention and Gated Recurrent Units (GRUs) with self-attention, is designed to enhance forecasting accuracy for both power generation and consumption within smart grid automation systems. Accurate power generation forecasts enable grid operators to anticipate energy supply requirements, while precise power consumption predictions assist in managing demand. By integrating predictions for both power generation and power consumption, a comprehensive perspective on electricity grid dynamics emerges, empowering stakeholders to make informed decisions and enhance the overall efficiency of grid operations. This holistic approach ensures that smart grids achieve their goal of delivering sustainable and efficient energy solutions for modern communities [4]. Highly accurate and efficient forecasting methods are essential for effective grid management, enabling electricity companies to make critical decisions [5]. Even a modest improvement, such as reducing

forecasting errors by 1%—can result in savings amounting to tens of millions of pounds [6]. The core objective of forecasting power generation and consumption is to ensure a steady power supply while minimizing operational expenses.

The remainder of this paper is organized as follows: Section 2 reviews the existing literature on the topic, Section 3 provides a brief description of the proposed method, Section 4 presentation of the experimental results, where the performance of the proposed method is evaluated in terms of power generation and consumption forecasting, Section 5 discusses the model's performance compared to state-of-the-art methods and its practical implications, and Section 6 concludes the paper.

2 Related work

Despite its importance, accurately predicting power generation and consumption remains a significant challenge. Power generation data is often chaotic, random, and intermittent, making it difficult to predict with precision. Similarly, fluctuations in power consumption arise from the diverse behaviors of consumers, influenced by factors such as special occasions, increased adoption of smart devices, and the growing use of electric vehicles [7]. To address these challenges, accurate forecasting methods are considered as a key strategy for maintaining a reliable power supply. It plays a pivotal role in achieving an equilibrium between power generation and consumption, ultimately ensuring the continuous operation of modern electricity grids. Given the critical role of power generation and consumption forecasting, several models are proposed in the literature, that are categorized into short-, long-, and medium-term forecasting horizons. However, short-term forecasting is crucial for balancing the power. On the other hand, medium- and long-term forecasts are primarily used for settling energy prices, planning load dispatch, and scheduling maintenance behavior. Short-term forecasting finds extensive applications, making it a focal point for researchers aiming to enhance accuracy. The smooth operation of smart grids, particularly through electrical automation, heavily depends on precise short-term forecasts, prompting the development of advanced, data-driven methodologies. These methods are broadly classified into statistical techniques, AI approaches, deep learning models, and hybrid frameworks that combine multiple methodologies to maximize performance. This diversity in approaches reflects ongoing efforts to optimize forecasting techniques and ensure the reliable operation of modern energy systems.

In the early stages of load and demand forecasting, statistical methods were predominantly employed, leading to the development of various models. These include approaches based on Bayesian analysis [8], autoregressive techniques and moving averages [9], Hammerstein frameworks [10], Kalman filters [11], and multiple kernel methods [12]. While these techniques offered valuable insights, their effectiveness diminishes when dealing with nonlinear and complex datasets. To address these limitations, researchers have shifted their focus toward AI-assisted methods. These advanced approaches leverage the strengths of artificial intelligence to overcome the challenges posed by nonlinear data, providing more accurate and robust forecasting solutions. This transition highlights the evolving nature of forecasting techniques as they adapt to the increasing complexity of modern energy systems. These include methods utilizing support vector machines, artificial neural networks, extreme learning machines, and fuzzy networks [13, 14, 15, 16]. While these approaches have demonstrated improved accuracy compared to traditional techniques, they often rely on manual features which limit their ability to generalize across diverse datasets.

A significant limitation of these shallow models lies in their tendency to experience parameter non-convergence and instability when dealing with the vast historical datasets inherent in power data. Recognizing these challenges, researchers and domain experts have increasingly turned to deep learning methodologies. Deep learning offers a robust alternative, leveraging its capacity to learn hierarchical representations directly from raw data, thus addressing the shortcomings of earlier AI-based models and paving the way for more stable and generalizable forecasting solutions. Deep learning-based models have revolutionized forecasting by offering end-to-end feature extraction capabilities and the ability to learn from extensive datasets while maintaining high generalizability. Recent advancements have extended their application to power forecasting, utilizing architectures such as RNNs [31], ESNs [32], LSTM networks [33], GRUs [34], and CNNs [35, 36]. Compared to traditional statistical and shallow learning methods, deep learning models offer significant advantages in handling complex datasets. However, developing effective models requires a deep understanding of the underlying data. In the context of power generation and consumption forecasting, historical data often exhibits time series patterns with spatiotemporal dependencies. While deep learning models excel in learning specific features, their ability to process multiple feature types simultaneously is limited. To address this, hybrid models have emerged, combining the strengths of various architectures to effectively capture the intricate spatiotemporal representations inherent in such datasets.

Hybrid models that integrate multiple techniques have proven highly effective in achieving accurate power generation and consumption forecasting. Notable hybrid approaches include combinations such as CNN-RNN [37], autoencoders with BiLSTM [38], CNN-LSTM [39], CNN-GRU [40], ConvLSTM (CLSTM) [41], and ESN-CNN. Among these, models that pair CNNs with RNN variants, such as LSTM, GRU, and ESN, have consistently demonstrated state-of-the-art performance. These hybrid architectures excel in extracting spatiotemporal features from historical data, enabling precise predictions for power generation and consumption. Recent advancements in hybrid forecasting models have incorporated attention mech-

Table 1: Summary of related work on power forecasting models

Work	Model	Dataset	Metrics	Limitations
Chen et al. [17]	RCC-LSTM (residual CNN + LSTM)	Solar PV generation data (5-min resolution)	RMSE & MAE	Inflexible selection of RCC threshold values
Zang et al. [18]	Hybrid CNN with meta-learning	DKASC	MAE, RMSE reported	Computationally heavy; limited validation; unclear generalizability.
Zhou et al. [19]	ELM + genetic algorithm	PV output (15-min data)	R^2, MAE, nRMSE	Needs more diverse data; may underperform across varied locations.
Cheng et al. [20]	Graph-based DL model	PV plant data (hourly)	RMSE, MAPE	Requires complex preprocessing; tested on narrow scope.
Korkmaz [21]	SolarNet: CNN + VMD	Solar PV (hourly)	RMSE=0.309, MAE=0.175	Single-site test; preprocessing adds overhead; uncertain performance in extremes.
Wang, Qi, and Liu [22]	Deep Neural Networks	1B DKASC	RMSE, MAPE (improvements)	risks overfitting; lacks uncertainty modeling.
Rajabi and Estebsari [23]	2D-CNN on recurrence plots	IHEPC	RMSE, MAE (20% RMSE vs 1D CNN)	Small-scale test; Limited exploration of time series to image encoding methods.
Ullah et al. [24]	CNN + Bi-LSTM	Residential load dataset	RMSE, MAE	Short-term focus; smart meter data required; only deployable on PCs.
Haq et al. [25]	ConvLSTM-BiLSTM	Res. & comm. load (smart grid)	RMSE, MAPE (improvements)	complex and resource-heavy; Not designed for deployment on resource-constrained devices.
Kim and Cho [26]	CNN-LSTM hybrid	IHEPC	RMSE, MAE	Limited Dataset diversity
Abdel-Basset et al. [27]	STLF-Net	IHEPC and AEP	RMSE, MAE, MAPE	Data ambiguities/irregularities may limit real-world accuracy/confidence. Load data vulnerability to attacks.
Han et al. [28]	Multilayer GRU	IoT energy data	MSE, RMSE	Primarily focused on short-term load forecasting (STLF). edge device constraints.
Kim and Cho [29]	Autoencoder based	Household power consumption	MSE, MRE	Limited data;
Khan et al. [30]	CNN with LSTM-AE	UCI Residential & Korean Commercial building data	MAE, MSE, MAPE	complex model; tested on limited geography.

anisms to further improve accuracy. Examples include transformer-based networks, multi-attention networks, and temporal CNNs with channel-attention [42, 43, 44, 45]. These cutting-edge methods effectively capture complex relationships within the data, leading to significant improvements in forecasting performance. To provide a comprehensive overview of the state-of-the-art, Table 1 summarizes key prior works on power forecasting, comparing their model types, datasets, evaluation metrics, and limitations. The table highlights challenges such as computational complexity, limited dataset diversity, and lack of attention mechanisms in existing models, which often hinder their ability to effectively capture both spatial and temporal dependencies.

This paper emphasizes the following key areas that need further investigation to enhance forecasting accuracy for power generation and consumption:

1. Previous research involving stacked methods has shown better performance; however, these models primarily use previous data that represents power generation or consumption over one or two hours as input. These models typically generate forecasts for the following hour using the sliding window technique. Unlike video and image analysis, power data often lacks the rich feature dimensions necessary for capturing both spatial-temporal relationships efficiently within this architecture.

2. Many existing models process features through their architectures without adequately refining the extracted spatial and temporal representations, often routing unoptimized feature vectors directly to dense layers. While some studies have incorporated attention mechanisms to improve results, these are typically limited

to either spatial or temporal attention, failing to address the combined spatiotemporal nature of power data. Thus, a comprehensive attention mechanism that refines both spatial and temporal features across the network is essential for more accurate forecasting.

3. Another main challenge is the evaluation of the models on specific types of datasets. The lack of varied and high-resolution datasets in evaluations limits the generalization and robustness of these forecasting models.

Addressing these gaps is crucial for advancing the field of power generation and consumption forecasting and improving the accuracy, applicability, and scalability of predictive models. To guide this study, we pose the following research questions:

1. Can our dual-stream hybrid model, as a unified approach for power generation and consumption forecasting, improve accuracy over single-stream deep learning models?

2. How does incorporating spatial and self-attention mechanisms improve forecasting performance?

These questions evaluate the effectiveness of the proposed dual-stream hybrid model, which processes both generation and consumption data to capture their inter-dependencies, and the role of attention mechanisms in enhancing forecasting accuracy. This paper presents several key contributions:

1. A dual-stream architecture is developed for power forecasting, processing multivariate time series data in parallel through CNN and GRU streams. This approach efficiently captures spatiotemporal dependencies, addressing the limitations of stacked models that struggle with limited feature dimensions in power data.

2. To enhance feature refinement across the network, we integrate a Spatial Attention Module (SAM) in the CNN stream to prioritize critical spatial features and a Self-Attention Module (SEAM) in the GRU stream to refine temporal dependencies. These modules dynamically select and emphasize relevant features, overcoming the shortcomings of models that lack comprehensive feature refinement.

3. The proposed model undergoes rigorous evaluation across multiple datasets. The results consistently demonstrate that our model outperforms previous baseline approaches in both power generation and consumption forecasting, showcasing its versatility and reliability across different contexts.

3 Proposed solution

The choice of CNNs for spatial feature extraction and GRUs for temporal dependency modeling in the proposed dual-stream architecture is motivated by their complementary strengths in handling the complex spatiotemporal characteristics of power data. CNNs are highly effective in capturing spatial patterns due to their ability to apply convolutional filters that identify local correlations and hierarchical features within multivariate time series data, such as weather-related variables (e.g., temperature, humidity) influencing power generation. This makes CNNs particularly suitable for extracting spatial dependencies in power data, which often exhibit localized patterns driven by environmental factors. In contrast, GRUs are designed to model sequential data by maintaining memory of previous time steps through update and reset gates. Compared to LSTM networks, GRUs offer a simpler architecture with fewer parameters, making them computationally efficient while still effectively capturing long-term temporal dependencies in time series data. The decision to use GRUs over other RNN variants is driven by their balance of performance and computational efficiency, which is critical for real-time forecasting applications in electrical automation systems. This dual-stream approach leverages the strengths of both architectures to address the limitations of single-stream models, which often struggle to simultaneously capture spatial and temporal features.

Likewise, the selection of a dual-stream CNN-GRU architecture over other alternatives such as transformer-based models or hybrid BiLSTM-attention models is driven by several key considerations. Transformer-based models, while powerful for capturing long-range dependencies, require significant computational resources and large datasets for effective training, which may not be ideal for real-time forecasting in resource-constrained environments like microgrids. Additionally, transformers primarily focus on temporal relationships and may not efficiently capture spatial dependencies in multivariate power data without complex modifications. The dual-stream approach allows parallel processing of spatial and temporal features, enhancing the model's ability to capture the interdependence between power generation and consumption. This design choice aligns with the need for a unified model that jointly predicts both processes while maintaining computational efficiency.

The core framework of the proposed Dual-Stream Network is illustrated in Figure 1. In this architecture, the input data is processed in parallel through the two streams. These architectures are designed to extract spatiotemporal features from the input data. The outputs of both the CNN and GRU streams are then merged into a single feature vector, which is used for further processing. To optimize the feature selection, a SAM is incorporated to identify the most representative features, which are then utilized for the final forecasting task. The following subsections provide an in-depth description of the internal components of the proposed model.

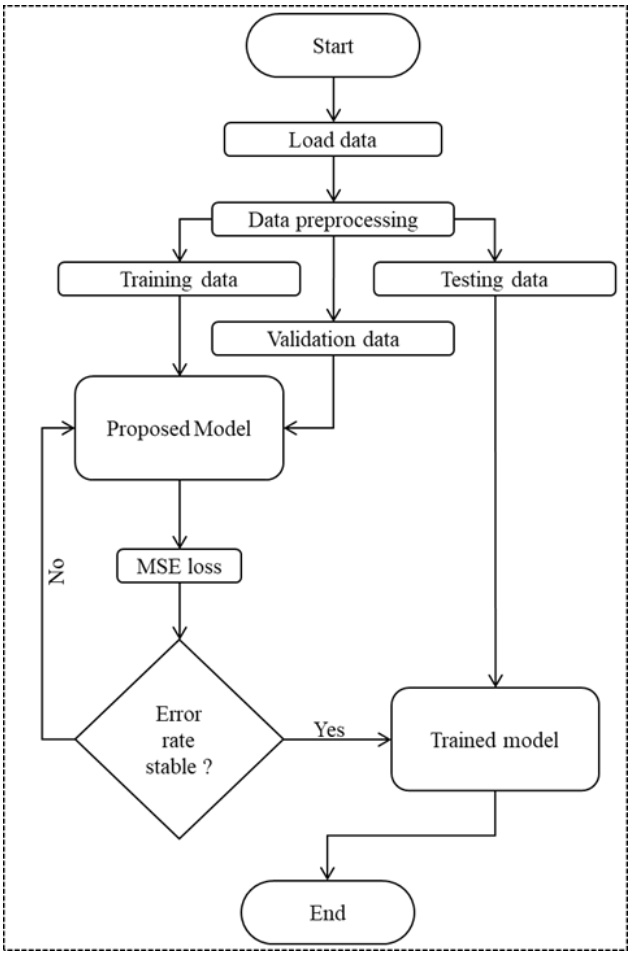


Figure 1: Flow diagram of proposed model

3.1 Dual stream network

The proposed Dual-Stream Network (DSN) architecture consists of two parallel streams, one using CNN layers for extracting cross-variable correlations and the other employing GRU layers for capturing temporal dependencies. Together, these streams are designed to handle the complex spatiotemporal characteristics of power data, which is manipulated by different weather-related factors such as temperature, humidity, and rain. The CNN stream processes a multivariate input sequence of PV power and weather-related variables, structured as a 2D matrix (time steps × variables, e.g., PV power, temperature, humidity). In this context, the variables represent a pseudo-spatial dimension, where the CNN captures correlations among them, such as the influence of temperature and humidity on PV power output. These cross-variable relationships are analogous to spatial patterns in image processing, enabling the CNN to extract meaningful features for power forecasting. The input to the CNN consists of a feature vector $X_i^0 = X_1, X_2, X_3 \ldots . X_n$ where each element corresponds to a variable, and the data is passed through different CNN layers and pooling layers. The CNN applies convolutional filters to produce feature maps that encode these cross-variable correlations, which are subsequently refined by SAM to prioritize the most relevant features (e.g., emphasizing temperature over less impactful variables). The convolution operation is used to extract cross-variable patterns in the data by applying filters to the input sequence. The output feature vector from the convolution layer $Y_{i,j}^1$ is calculated as:

$$Y_{i,j}^1 = \alpha(B_j^1 + \sum_{m=1}^{M} W_{m,j}^1 X_{i+m-1,j}^0 \quad (1)$$

Here, B_j^1 represents the bias term, $W_{m,j}^1$ is the weight matrix, and α is the activation function. The convolutional layers are followed by max pooling to reduce the dimensionality of the output and retain the most important features:

$$Y_{i,j}^l = \alpha(B_j^l + \sum_{m=1}^{M} W_{m,j}^l X_{i+m-1,j}^0 \quad (2)$$

The GRU stream is designed to handle temporal dependencies by using GRU layers, which are lighter and more efficient compared to traditional RNNs. The GRU processes the same multivariate time series input, but focuses on the temporal dimension, modeling sequential patterns across time steps (e.g., daily or seasonal trends in power consumption). The GRU outputs a sequence of hidden states capturing these temporal dependencies, which are refined by SEAM to focus on significant time steps. The GRU architecture consists of two main gates: the Reset Gate and the Update Gate. The update gate u_t is calculated as:

$$u_t = \alpha[(w_1 \times i_t) + (w_2 \times i_{t-1})] \quad (3)$$

The reset-gate r_t find out the degree to which the previous hidden state is remembered, and is calculated as:

$$r_t = \alpha[(w_1 \times i_t) + (w_2 \times i_{t-1})] \quad (4)$$

The memory content $m_r^{'}$ captures the past information and is computed as:

$$m_r^{'} = \tanh[(w_1 \times i_t) + (r_t \cdot i_{t-1})] \quad (5)$$

Finally, the output memory combines previous hidden state and the memory content:

$$m_t = (u_t \cdot i_{t-1} + (1 - u_t) \cdot m_r^{'}) \quad (6)$$

Spatial Attention: The extracted feature vector is passed to the 1D-SAM, which refines feature representation by emphasizing the most important elements. SAM operates on the CNN output, which consists of feature maps representing spatial correlations among input variables (e.g., weather-related features like temperature and humidity) across time steps. It creates a spatial attention map by analyzing these inter-variable relationships, prioritizing features that are most relevant to power forecasting. This module creates a spatial attention map by analyzing the inter-spatial relationships within the feature data. To achieve this,

it applies avg-pooling and max-pooling along the channel-dimension, producing two distinct feature maps: $\mathrm{M}_s(\mathrm{F}) \in \mathrm{R}^{t\times c}$, where t represents the time steps and c indicates the number of channels. These pooled feature maps are then concatenated and processed through a 1D convolutional layer with a kernel size 3×3, resulting in the final spatial attention map $\mathrm{M}_s(\mathrm{F}) \in \mathrm{R}^{t\times c}$. This attention map assigns importance to specific regions in the feature space, effectively determining which features should be enhanced or suppressed as given below:

$$\mathrm{M}_s(\mathrm{F}) = \sigma(f^{3\times3}([\mathrm{AvgP}(\mathrm{l}_5), \mathrm{MaxP}(\mathrm{l}_5)])) \tag{7}$$

$$\mathrm{M}_s(\mathrm{F}) = \sigma(f^{3\times3}(\mathrm{l}^s_{5\mathrm{avg}}, \mathrm{Fl}^s_{5\mathrm{max}})) \tag{8}$$

Here, σ denotes the sigmoid activation function, $f^{(3\times3)}$ is the convolutional operation with a kernel size of 3×3), and AvgP and MaxP refer to the average-pooling and max-pooling operations, respectively. This process ensures that SAM enhances the model's focus on critical spatial features, such as weather variables strongly correlated with power generation, while suppressing less relevant ones, thereby improving forecasting accuracy.

Self-Attention: The SEAM processes the output of the GRU stream, which consists of a sequence of hidden states capturing temporal dependencies across time steps. Unlike SAM, which focuses on spatial correlations among variables, SEAM dynamically selects the most relevant temporal features by assigning weights to significant time steps in the sequence. The attention scores for each feature in the J^{th} timestamp in the d^{th} dimension is calculated as:

$$S_{J,d} = f_{\mathrm{sco}}(W_{J,d}[h_{1,d}, h_{2,d}, h_{3,d}, \ldots, h_{n,d}]) \tag{9}$$

where $d = 1, 2, 3, \ldots, n$ and $J = 1, 2, 3, \ldots, o$. Here, $h_{J,d}$ represents the hidden state at the J^{th} timestamp and the d^{th} dimension, while f_{sco} is a function (implemented with fully connected layers) to score the importance of each feature. This mechanism allows SEAM to emphasize critical temporal patterns, such as peak consumption periods, while down-weighting less significant time steps, complementing SAM's spatial refinement.

Finally, the output from the SAM and SEAM is merged and forwarded to Dense layers, where the final power forecast is generated. The fully connected layers use the following operation to calculate the output d^l_i:

$$d^l_i = \sum_j W^{l-1}_{ji}(\alpha(X^{l-1}_i) + B^{l-1}_j) \tag{10}$$

Where W^{l-1}_{ji} is the weight matrix, α is the activation function, and B^{l-1}_j is the bias term. This architecture ensures effective extraction of both spatial and temporal features while utilizing attention mechanisms to improve feature selection for more accurate forecasting. The model is capable of handling complex, irregular patterns in PV power generation, resulting in a robust and accurate forecasting system.

3.2 Network architecture

The network of the model is proposed to efficiently capture spatiotemporal features for precise forecasting of power. The network consists of convolutional, GRU, spatial attention, Self-attention, and fully connected layers. The model's parameters are adjusted through extensive experimentation to achieve optimal performance. This customization ensures that the architecture is fine-tuned for accurate forecasting. The input to the model is 12×10 (for power generation) data matrix, which contains the necessary weather-related variables (e.g., temperature, humidity, rain, etc.) and power generation or consumption. This data is passed through both streams in parallel to learn spatial-temporal features. The first stream utilized two convolutional layers and a pooling layer. This setup allows for spatial feature extraction from the input data with Filter sizes: 16 and 32, 3 Kernel size, and ReLU activation-function. The max pooling operation is used to reduce dimensionality and retain important features. The output of CNN is then passed through SAM to select optimal features in spatial dimension. The GRU stream consists of two stacked GRU layers to capture temporal dependencies from the input data with Cell sizes: 32 and 16 (for each GRU layer). GRU layers are particularly effective at learning sequential patterns and trends in time series data. The output of GRU is then passed through the SEAM, which dynamically selects the most important features from the fused vector by assigning different weights to different features. The SEAM helps highlight critical temporal and spatial features while down-weighting less important ones. Finally, the output of spatial attention send self-attention is fused and passed through fully connected layers to generate the final power forecast. Extensive experimentation is conducted to finalize the model's internal parameters. The experiments ensure that the model's performance is optimized for accurate and reliable forecasting. The parameters, including the filter sizes, kernel sizes, GRU cell sizes, and other network settings, are selected based on their ability to extract meaningful features and make accurate predictions. A flow diagram of the proposed model is illustrated in Figure 1.

3.2.1 Hyperparameter selection and tuning

To ensure optimal performance of the proposed dual-stream CNN-GRU model, a systematic hyperparameter selection and tuning process was employed. The CNN stream consists of two convolutional layers with 16 and 32 filters, respectively, each using a kernel size of 3 and ReLU activation, followed by a max-pooling layer with a pool size of 2. The GRU stream comprises two stacked GRU layers with 32 and 16 cells, respectively, using tanh activation for the recurrent steps. The fully connected layers include a hidden layer with 64 neurons (ReLU activation) and an output layer with 1 neuron (linear activation). The Spatial Attention Module (SAM) uses a 1D convolutional layer with a 3×3 kernel, and the Self-Attention Module (SEAM) employs a fully connected layer to compute attention scores.

Hyperparameter tuning was conducted using a grid search over the following ranges: CNN filter sizes ([8, 16], [16, 32], [32, 64]), GRU cell sizes ([16, 8], [32, 16], [64, 32]), learning rates ([0.0001, 0.001, 0.01]), and batch sizes ([32, 64, 128]). The Adam optimization algorithm was selected for its robustness in training deep neural networks, with a default beta1 of 0.9 and beta2 of 0.999. The model was trained for 100 epochs with early stopping (patience of 10 epochs) to prevent overfitting. The tuning process utilized 5-fold cross-validation on the DKASC dataset to evaluate performance, with the root mean squared error (RMSE) on the validation set as the primary metric. This configuration was further validated on the IHEPC dataset to ensure generalizability. The systematic tuning process ensures transparency and reproducibility, providing a robust foundation for the model's performance in power forecasting applications.

4 Results

The experimental results are presented for power generation and consumption prediction. We utilized datasets containing time series data on factors such as weather and time of day for solar power generation and electricity consumption. An ablation study was also conducted to examine the contributions of different components of the model. The results are further analyzed and compared with state-of-the-art methods in the Discussion section.

4.1 Datasets

The electricity generation and consumption datasets used in this study include data for both renewable power generation and household electricity consumption. For renewable power generation, the DKASC datasets are utilized, which includes daily data recorded at a five-minute resolution from multiple active solar power plants in Alice Springs, Australia. The three datasets used are Trina-23.4-kW, Trina-10.5-kW, and Eco-Kinetics-26.5-kW. These datasets contain both renewable power generation and weather-related variables like humidity, rainfall, and temperature. For electricity consumption prediction, the IHEPC dataset is used, which includes one-minute resolution data from 2006 to 2010. This dataset records information on active and reactive power, voltage, intensity, and submetering, along with time and date information.

While the DKASC and IHEPC datasets provide valuable insights into renewable power generation and household electricity consumption, they have certain limitations that impact their representativeness of real-world power grids. The DKASC datasets, collected from solar power plants in Alice Springs, Australia, primarily reflect the climatic and operational conditions of a single geographical region with a arid climate, which may not fully capture the variability of solar generation in regions with different weather patterns, such as tropical or temperate climates. Similarly, the IHEPC dataset, derived from a single household in France, may not adequately represent the diverse consumption patterns observed in larger, industrial, or multi-household grid configurations. These datasets also lack significant variability in grid infrastructure, such as differences in energy storage systems or hybrid renewable sources, which are common in modern smart grids. Despite these constraints, the datasets' high temporal resolution (5-minute for DKASC, 1-minute for IHEPC) and inclusion of relevant variables (e.g., weather, voltage) make them suitable for developing and testing forecasting models, though broader applicability requires further validation across diverse datasets.

4.2 Data preprocessing

To ensure the quality and consistency of the DKASC and IHEPC datasets for model training and evaluation, several preprocessing steps were applied. First, data normalization was performed using min-max scaling to transform all features (e.g., power generation, consumption, temperature, humidity) into a [0, 1] range. This step mitigates the impact of varying scales across different variables, ensuring stable training of the neural network. The normalization formula used is given by:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{11}$$

where x is the original value, and $x_{\min}$ and $x_{\max}$ are the minimum and maximum values of the feature across the dataset. Missing data, which accounted for less than 2% of the records in both datasets, were handled using linear interpolation for time-series continuity, leveraging neighboring data points to estimate missing values. This approach was chosen to preserve temporal patterns critical for forecasting. Outliers, identified as values exceeding three standard deviations from the mean, were capped at the 99th percentile to reduce their influence without removing data points. The datasets were partitioned into training, validation, and test sets using a 70:15:15 split, respectively. To maintain temporal order, the data was split sequentially, with earlier data used for training and later data for validation and testing. This partitioning ensures that the model is evaluated on unseen data, simulating real-world forecasting scenarios.

4.3 Evaluation metrics

The performance of the proposed model is evaluated using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Normalized Root Mean Squared Error (nRMSE), and Mean Absolute Error (MAE), as these metrics are widely adopted in time-series forecasting, particularly for power generation and consumption. MSE and RMSE are selected because they emphasize larger errors, which is critical in power forecasting where significant deviations can lead to costly grid imbalances. RMSE provides an interpretable measure in the same units as the forecasted variable, making it suitable for assessing the magnitude of pre-

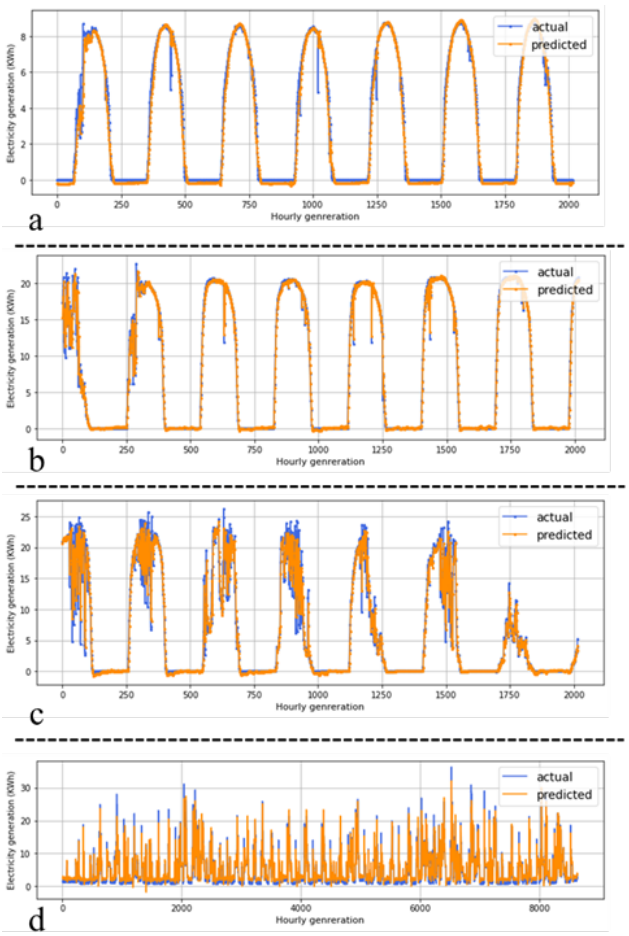


Figure 2: Predicted vs actual values where a, b, c show power generation and d show power consumption results

diction errors. However, both MSE and RMSE are sensitive to outliers, which can skew results in datasets with extreme values, such as those caused by sudden weather changes. nRMSE normalizes RMSE by the range of the target variable, enabling fair comparisons across datasets with different scales, which is essential for evaluating model performance across diverse power generation and consumption datasets. MAE, on the other hand, measures the average absolute error, offering robustness to outliers and a straightforward interpretation of average prediction accuracy. While MAE is less sensitive to large errors compared to RMSE, it complements the other metrics by providing a balanced perspective on model performance. Together, these metrics provide a comprehensive evaluation of the model's accuracy and robustness, aligning with the requirements of power forecasting applications where both small and large errors impact grid reliability. The equations for these metrics are given below:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{m}(\alpha_i - p_i)^2}{N}} \tag{12}$$

$$\text{MSE} = \frac{\sum_{i=1}^{m}(\alpha_i - p_i)^2}{N} \tag{13}$$

$$\text{nRMSE} = \text{RMSE}/(\alpha_{\max} - \alpha_{\min}) \tag{14}$$

$$\text{MAE} = \frac{\sum_{i=1}^{n}|\alpha_i - p_i|}{N} \tag{15}$$

where α and p show the actual and model prediction.

4.4 Ablation study

The performance of the proposed model is evaluated by comparing it with several predictive modeling techniques, including traditional regression methods (SVR, Decision Tree, Linear Regression), deep models (MLP, LSTM, CNN), a transformer-based model, and a hybrid BiLSTM-attention model during the ablation study. The transformer-based model employs a multi-head self-attention mechanism to capture temporal dependencies, while the BiLSTM-attention model combines bidirectional LSTM layers with an attention mechanism to focus on relevant temporal features. Experimental results presented in Figures 3 to 6 demonstrate that traditional regression models perform worse than deep learning-based models. In particular, hybrid ones perform better comparatively. Among the hybrid models, the proposed model achieved the lowest error rates. For instance, on the Tarina 10.5 kW dataset, the proposed model achieved an RMSE of 0.0985, compared to 0.124 for the transformer-based model and 0.109 for the BiLSTM-attention model, highlighting the dual-stream architecture's superior ability to capture spatiotemporal dependencies. Similar trends were observed across other datasets, with the transformer model exhibiting higher computational costs (e.g., 2.15 seconds inference time on GPU) and the BiLSTM-attention model showing slightly higher errors due to its focus on temporal rather than spatial features. For example, on the Tarina 10.5 kW dataset, the proposed model achieved RMSE, MSE, nRMSE, and MAE values of 0.0985, 0.0097, 0.1198, and 0.0490, respectively. For the Tarina 23.4 kW dataset, the values were 0.0574 RMSE, 0.0033 MSE, 0.1512 nRMSE, and 0.0430 MAE, while for the Eco-Kinetics 26.5 kW dataset, the model achieved 0.0346 RMSE, 0.0012 MSE, 0.2393 nRMSE, and 0.0128 (MAE). The proposed model also outperformed others on the IHEPC data, with RMSE, MSE, nRMSE, and MAE values of 0.0300, 0.0009, 0.1158, and 0.0166, respectively. All experiments were conducted for one-hour-ahead predictions, and the proposed model consistently showed the lowest error rates, as evidenced by the close alignment between predicted and actual values in Figure 2. This confirms the model's applicability for both renewable power generation and electricity consumption prediction.

5 Discussion

The performance of the proposed dual-stream CNN-GRU model is analyzed compared to state-of-the-art (SOTA) methods, exploring the reasons behind its superior performance, and the practical implications of the results for electrical automation and smart grid applications. Additionally,

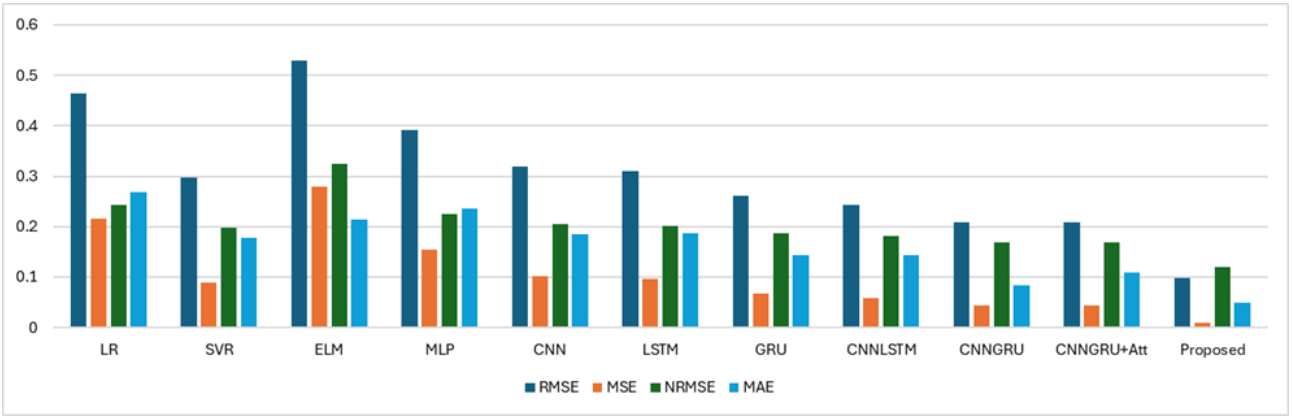


Figure 3: Comparative analysis of different models developed during ablation study over Tarina 10.5 kW

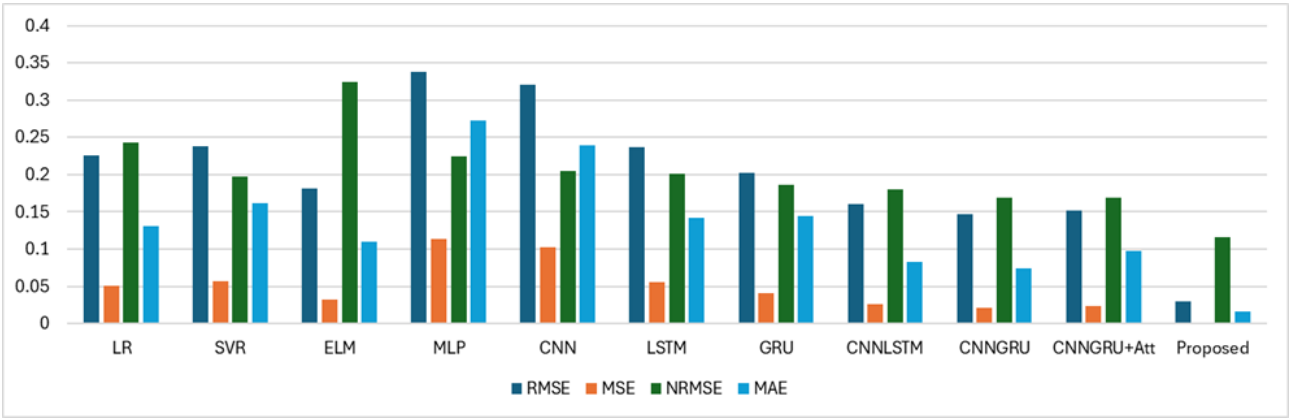


Figure 4: Comparative analysis of different models developed during ablation study over IHEPC

we discuss its scalability and deployment feasibility to address real-world constraints.

5.1 Comparative analysis of the proposed model with state-of-the-art models

The effectiveness of the proposed model is evaluated on the mentioned datasets, and its performance is compared with several baseline methods. For DKASC, the proposed model demonstrates significant improvements over the baseline methods, as summarized in Table 2. The RMSE, MSE, and MAE values achieved by the proposed model are 0.0635, 0.0047, and 0.0349, respectively. These results indicate a substantial reduction in prediction error compared to other state-of-the-art models. For instance, Chen et al. [17] reported an RMSE of 0.94 and an MAE of 0.587, while Zang et al. [18] achieved an MSE of 0.081 and an MAE of 0.152. The proposed model's superior performance can be attributed to its dual-stream architecture unlike single-stream models like Chen et al. [17] that rely on stacked layers. The integration of spatial and self-attention mechanisms further refines feature selection, enabling the model to prioritize relevant patterns, unlike Zang et al. [18], which lacks such mechanisms and suffers from computational complexity. However, on datasets with high variability (e.g., Eco-Kinetics 26.5 kW, nRMSE=0.2393), the model's performance, while still superior, shows a slightly higher normalized error, possibly due to the influence of extreme weather conditions that challenge even advanced feature extraction.

To validate the statistical significance of these improvements, we conducted Wilcoxon signed-rank tests across multiple runs on DKASC and IHEPC datasets. The tests yielded p-values < 0.05, indicating that the proposed model's improvements are statistically significant. Additionally, 95% confidence intervals for the proposed model's RMSE on the DKASC dataset (0.059–0.068) and IHEPC dataset (0.027–0.033) confirm the reliability of the reported metrics, with narrow intervals reflecting consistent performance across runs. These statistical analyses strengthen the claim that the dual-stream architecture and attention mechanisms contribute to robust forecasting improvements over SOTA methods.

On the IHEPC dataset, the proposed model also outperforms existing techniques, achieving an RMSE of 0.0300, an MSE of 0.0009, and an MAE of 0.0166, as shown in Table 3. These metrics are significantly better than those of the competing methods, such as Rajabi and Estebsari

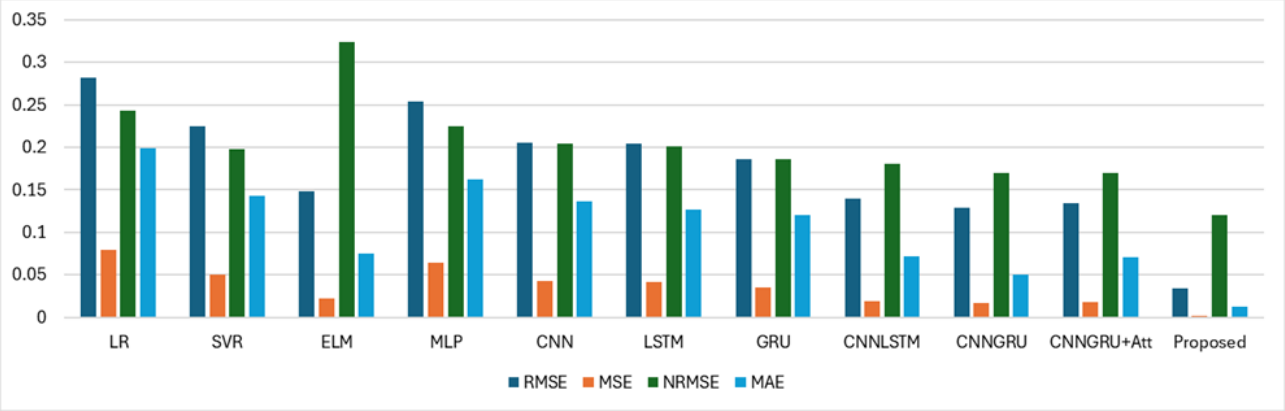


Figure 5: Comparative analysis of different models developed during ablation study over Tarina 26.5 kW

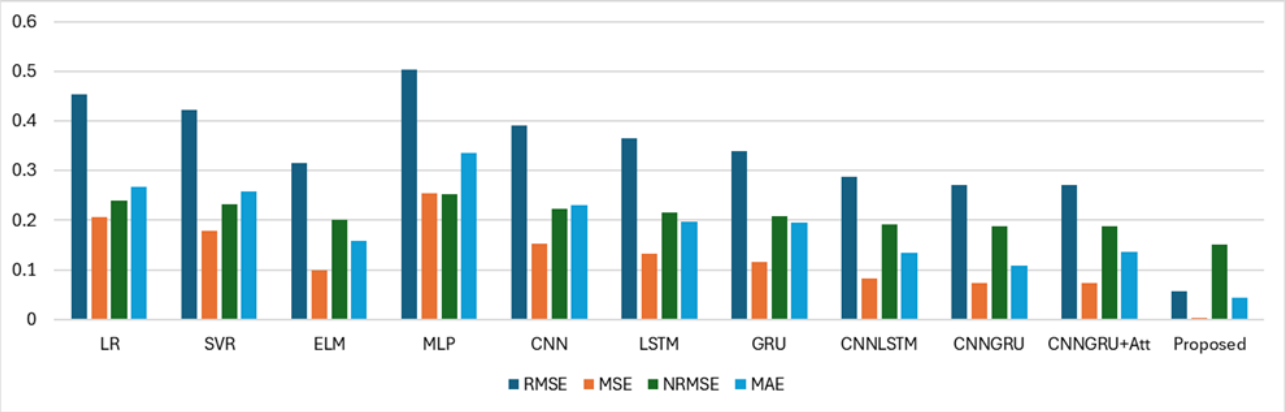


Figure 6: Comparative analysis of different models developed during ablation study over Tarina 23.4 kW

[23], who reported an RMSE of 0.79 and an MAE of 0.59, and Haq et al. [25], with an RMSE of 0.32 and an MAE of 0.31. The model's success on the IHEPC dataset is driven by the GRU stream's ability to capture long-term temporal dependencies in high-resolution (1-minute) consumption data, complemented by the CNN stream's extraction of spatial patterns from correlated variables like voltage and intensity. In contrast, models like Rajabi and Estebsari [23] rely on 2D-CNNs with limited temporal modeling, leading to higher errors. The attention mechanisms mitigate the impact of data irregularities noted in Abdel-Basset et al. [27], enhancing robustness. However, the model's performance advantage is less pronounced on datasets with significant noise, suggesting a potential area for improvement in handling data ambiguities.

5.2 Time complexity analysis

To ensure the proposed model is suitable for real-time power forecasting, we evaluate its computational efficiency by measuring inference time, defined as the time required for a single forward pass to generate a prediction. Inference time is critical for real-time applications in electrical automation, where rapid predictions are necessary to maintain grid stability. We compare the inference time of the proposed model against baseline models on three platforms-CPU, GPU, and Raspberry Pi (RPI)—to demonstrate its efficiency across diverse computational environments, including edge devices like Raspberry Pi commonly used in microgrids. All times are reported in seconds, ensuring consistency and clarity. This comparison highlights the model's lightweight design, which minimizes computational costs and response delays, thereby enhancing its applicability in real-time energy management systems.

To address this, we conduct a time complexity analysis of the model using 3 settings: CPU, GPU, and RPI as given in Table 4, whereas the results are given in Table 5. From the analysis, it is evident that our model has lower inference time than others across all platforms. Specifically, the proposed model achieves an inference time of 0.4354 seconds on GPU, 0.9874 seconds on CPU, and 1.674 seconds on RPI. This efficiency stems from the streamlined dual-stream architecture and the use of GRUs, which are computationally lighter than LSTMs used in models like Haq et al. [25]. In contrast, models like Han et al. [28] exhibit significantly higher inference times (e.g., 20.36 seconds on RPI), limiting their suitability for edge devices.

Comparing this with the performance of other models, such as those from Chen et al. [17] and Wang, Qi, and

Table 2: Comparative analysis of the model over the solar dataset

Method	MSE	RMSE	MAE
Chen et al. [17]	-	0.94	0.587
Zang et al. [18]	0.081	-	0.152
Li et al. [46]	-	-	0.2805
Zhou et al. [19]	-	-	0.2367
Cheng et al. [20]	-	0.336	0.177
Li et al. [47]	-	0.2357	-
Korkmaz [21]	-	0.309	0.175
Wang, Qi, and Liu [22]	-	0.343	0.126
Wang, Qi, and Liu [48]	-	0.621	0.221
Ours	**0.0047**	**0.0635**	**0.0349**

Table 3: Comparative analysis of the model over the IHEPC dataset

Paper	MAE	RMSE	MSE
Rajabi and Estebsari [23]	0.59	0.79	-
Ullah et al. [24]	0.3469	0.5650	0.3193
Haq et al. [25]	0.31	0.32	0.10
Kim and Cho [26]	0.3317	0.5957	0.3549
Khan et al. [30]	0.31	0.47	0.19
Han et al. [28]	0.19	0.22	0.17
Abdel-Basset et al. [27]	0.2674	0.4386	0.1924
Kim and Cho [29]	0.3953	-	0.3840
Khan et al. [49]	0.29	0.42	0.18
Khan et al. [50]	0.0038	0.0614	0.0537
Ours	**0.0009**	**0.0300**	**0.0166**

Table 4: Hardware configuration

Setting	Model	Memory
GPU	GeForce-RTX-3090	24GB
RPI	RPI 4 B+	8GB
CPU	Intel Core i5-6600	32GB

Liu [48], the proposed model's lower inference times (e.g., 0.4354 seconds on GPU vs. 0.72 seconds for Haq et al. [25]) highlight its suitability for real-time applications. The model's efficiency makes it viable for deployment on resource-constrained devices, unlike Haq et al. [25], which is resource-heavy.

In the context of real-world smart grid environments, the proposed model's inference time of 1.674 seconds on RPI is well within the requirements for one-hour-ahead forecasting, where predictions are typically needed every few minutes to support grid operations. This efficiency enables real-time deployment in decentralized microgrids, allowing rapid adjustments to energy production and consumption, thereby enhancing grid reliability and reducing operational delays. However, for applications requiring subsecond predictions, further optimization of the model's architecture may be necessary.

Method	Remarks	CPU	GPU	RPI
Chen et al. [17]	Intel-Core-i5, 8GB-RAM	6.387	-	-
Cheng et al. [20]	GTX-1080-GPU	-	3.5	-
Wang, Qi, and Liu [22]	Intel-Core-i5 8-GB-RAM	0.6217	-	-
Wang, Qi, and Liu [48]	Intel-Core-i5 8-GB-RAM	7.196	-	-
Haq et al. [25]	GeForce-RTX-2070	1.44	0.72	-
Han et al. [28]	Intel-Core-i9 RPI- Cortex-A53	6.38	-	20.36
Ours	**Table 3**	**0.9874**	**0.4354**	**1.674**

Table 5: Inference time (seconds) comparison with baseline models

5.3 Scalability and deployment considerations

The scalability and deployment feasibility of the proposed dual-stream CNN-GRU model are critical for its adoption in large-scale electrical automation systems and resource-constrained environments, such as embedded systems in microgrids. The model's memory footprint is approximately 120 MB, calculated based on the parameters of the CNN (16 and 32 filters), GRU (32 and 16 cells), and fully connected layers. This compact size makes it suitable for deployment on edge devices like the Raspberry Pi 4 B+ (8GB RAM), as demonstrated by its inference time of 1.674 seconds on this platform (Table 5). In large-scale systems, where thousands of nodes may require simultaneous forecasting, the model's parallel dual-stream architecture allows efficient batch processing, reducing computational overhead compared to sequential models like Chen et al. [17].

However, there are trade-offs between accuracy and computational efficiency. The proposed model achieves high accuracy (e.g., RMSE=0.0635 on DKASC) by leveraging spatial and self-attention mechanisms, which increase computational complexity compared to simpler models like linear regression or single-layer MLPs. In resource-constrained environments, such as embedded systems with limited memory (e.g., <2GB RAM), this complexity may necessitate model pruning or quantization to reduce the memory footprint and inference time further. For instance, reducing the number of filters in the CNN stream or GRU cells could lower the memory requirement to below 80 MB, at the cost of a potential 5–10% increase in RMSE, based on preliminary ablation studies. Alternatively, de-

ploying the model on cloud-based infrastructure with GPUs (e.g., GeForce RTX-3090, Table 4) ensures high accuracy and low inference time (0.4354 seconds) for large-scale systems, but this increases operational costs and latency in remote areas with limited connectivity.

To enhance scalability, the model supports distributed deployment, where multiple instances can run on edge devices for local forecasting, with periodic synchronization to a central server for global grid optimization. This approach mitigates bottlenecks in large-scale smart grids. Future work could explore techniques like knowledge distillation to create lighter models for ultra-constrained devices or federated learning to improve scalability across distributed grid networks, ensuring both accuracy and efficiency in diverse deployment scenarios.

5.4 Practical implications

The proposed model's high accuracy (e.g., RMSE=0.0635 on the DKASC dataset) and low inference time (e.g., 1.674 seconds on Raspberry Pi) have significant implications for electrical automation and smart grid management. By providing precise one-hour-ahead forecasts, the model enables grid operators to optimize energy production and distribution, reducing operational costs and enhancing grid stability. For instance, accurate solar power predictions allow for better integration of renewable energy, minimizing reliance on fossil fuel backups. The model's compatibility with edge devices like RPI supports decentralized energy management in microgrids, enabling real-time decision-making in smart cities. However, challenges such as handling extreme weather-induced data variability suggest the need for future enhancements, such as incorporating uncertainty modeling to further improve robustness.

6 Conclusion

The primary goal of the smart grid is to maintain equilibrium between power, enabling effective and dependable energy management. Predictive modeling plays a crucial role in achieving this balance, providing insights into future electricity generation and consumption patterns. While numerous predictive models have been developed in the literature, there remains a need for further improvement in prediction accuracy to enhance the energy sector with lower computation. This study introduces a robust and efficient hybrid model designed to forecast electricity generation and consumption effectively. The proposed model integrates CNN with spatial attention and GRU with self-attention in dual stream mechanism, achieving high prediction accuracy while maintaining low computational complexity. After conducting a comprehensive study of various models, the results confirm that the proposed model outperforms baselines in terms of performance and execution time. In the future, the focus will shift towards investigating cutting-edge technologies like reinforcement learning, explainable AI, active learning, and lifelong learning methods to enhance the accuracy and effectiveness of forecasting models. These innovations will play a crucial role in creating more efficient and flexible systems, enhancing the adaptability of smart grid applications for future energy management needs.

References

[1] Jie Ma and Xiandong Ma. "A review of forecasting algorithms and energy management strategies for microgrids". In: *Systems Science; Control Engineering* 6.1 (Jan. 2018), pp. 237–248. ISSN: 2164-2583. DOI: 10.1080/21642583.2018.1480979. URL: http://dx.doi.org/10.1080/21642583.2018.1480979.

[2] Yu Wang et al. "LASSO and LSTM Integrated Temporal Model for Short-Term Solar Intensity Forecasting". In: *IEEE Internet of Things Journal* 6.2 (Apr. 2019), pp. 2933–2944. ISSN: 2372-2541. DOI: 10.1109/jiot.2018.2877510. URL: http://dx.doi.org/10.1109/jiot.2018.2877510.

[3] Guobin Xu et al. "Towards Integrating Distributed Energy Resources and Storage Devices in Smart Grid". In: *IEEE Internet of Things Journal* (2016), pp. 1–1. ISSN: 2327-4662. DOI: 10.1109/jiot.2016.2640563. URL: http://dx.doi.org/10.1109/jiot.2016.2640563.

[4] Zulfiqar Ahmad Khan et al. "DSPM: Dual sequence prediction model for efficient energy management in micro-grid". In: *Applied Energy* 356 (Feb. 2024), p. 122339. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2023.122339. URL: http://dx.doi.org/10.1016/j.apenergy.2023.122339.

[5] Wendong Zheng and Gang Chen. "An Accurate GRU-Based Power Time-Series Prediction Approach With Selective State Updating and Stochastic Optimization". In: *IEEE Transactions on Cybernetics* 52.12 (Dec. 2022), pp. 13902–13914. ISSN: 2168-2275. DOI: 10.1109/tcyb.2021.3121312. URL: http://dx.doi.org/10.1109/tcyb.2021.3121312.

[6] Zulfiqar Ahmad Khan et al. "Efficient Short-Term Electricity Load Forecasting for Effective Energy Management". In: *Sustainable Energy Technologies and Assessments* 53 (Oct. 2022), p. 102337. ISSN: 2213-1388. DOI: 10.1016/j.seta.2022.102337. URL: http://dx.doi.org/10.1016/j.seta.2022.102337.

[7] Seyed Mohammad Jafar Jalali et al. "A novel evolutionary-based deep convolutional neural network model for intelligent load forecasting". In: *IEEE Transactions on Industrial Informatics* 17.12 (2021), pp. 8243–8253. DOI: 10.1109/TII.2021.3065718.

[8] Lei Tang et al. "Long-term electricity consumption forecasting based on expert prediction and fuzzy Bayesian theory". In: *Energy* 167 (Jan. 2019), pp. 1144–1154. ISSN: 0360-5442. DOI: 10.1016/j.energy.2018.10.073. URL: http://dx.doi.org/10.1016/j.energy.2018.10.073.

[9] S.Sp. Pappas et al. "Electricity demand loads modeling using AutoRegressive Moving Average (ARMA) models". In: *Energy* 33.9 (Sept. 2008), pp. 1353–1360. ISSN: 0360-5442. DOI: 10.1016/j.energy.2008.05.008. URL: http://dx.doi.org/10.1016/j.energy.2008.05.008.

[10] Othman Ait Maatallah et al. "Recursive wind speed forecasting based on Hammerstein Auto-Regressive model". In: *Applied Energy* 145 (May 2015), pp. 191–197. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2015.02.032. URL: http://dx.doi.org/10.1016/j.apenergy.2015.02.032.

[11] Zhuang Zheng, Hainan Chen, and Xiaowei Luo. "A Kalman filter-based bottom-up approach for household short-term load forecast". In: *Applied Energy* 250 (Sept. 2019), pp. 882–894. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2019.05.102. URL: http://dx.doi.org/10.1016/j.apenergy.2019.05.102.

[12] Di Wu et al. "Multiple Kernel Learning-Based Transfer Regression for Electric Load Forecasting". In: *IEEE Transactions on Smart Grid* 11.2 (Mar. 2020), pp. 1183–1192. ISSN: 1949-3061. DOI: 10.1109/tsg.2019.2933413. URL: http://dx.doi.org/10.1109/tsg.2019.2933413.

[13] Zhongfu Tan et al. "Combined electricity-heat-cooling-gas load forecasting model for integrated energy system based on multi-task learning and least square support vector machine". In: *Journal of Cleaner Production* 248 (Mar. 2020), p. 119252. ISSN: 0959-6526. DOI: 10.1016/j.jclepro.2019.119252. URL: http://dx.doi.org/10.1016/j.jclepro.2019.119252.

[14] Ping-Huan Kuo and Chiou-Jye Huang. "A High Precision Artificial Neural Networks Model for Short-Term Energy Load Forecasting". In: *Energies* 11.1 (Jan. 2018), p. 213. ISSN: 1996-1073. DOI: 10.3390/en11010213. URL: http://dx.doi.org/10.3390/en11010213.

[15] Mehdi Rafiei et al. "Probabilistic Load Forecasting Using an Improved Wavelet Neural Network Trained by Generalized Extreme Learning Machine". In: *IEEE Transactions on Smart Grid* 9.6 (Nov. 2018), pp. 6961–6971. ISSN: 1949-3061. DOI: 10.1109/tsg.2018.2807845. URL: http://dx.doi.org/10.1109/tsg.2018.2807845.

[16] Jieqiong Zhang. "Optimizing the Analysis of Energy Plants and High-Power Applications Utilizing the Energy Guard Ensemble Selector (EGES)". In: *Informatica* 49.10 (2025), pp. 113–126. DOI: 10.31449/inf.v49i10.7264.

[17] Biaowei Chen et al. "Very-Short-Term Power Prediction for PV Power Plants Using a Simple and Effective RCC-LSTM Model Based on Short Term Multivariate Historical Datasets". In: *Electronics* 9.2 (Feb. 2020), p. 289. ISSN: 2079-9292. DOI: 10.3390/electronics9020289. URL: http://dx.doi.org/10.3390/electronics9020289.

[18] Haixiang Zang et al. "Day-ahead photovoltaic power forecasting approach based on deep convolutional neural networks and meta learning". In: *International Journal of Electrical Power; Energy Systems* 118 (June 2020), p. 105790. ISSN: 0142-0615. DOI: 10.1016/j.ijepes.2019.105790. URL: http://dx.doi.org/10.1016/j.ijepes.2019.105790.

[19] Yi Zhou et al. "Prediction of photovoltaic power output based on similar day analysis, genetic algorithm and extreme learning machine". In: *Energy* 204 (Aug. 2020), p. 117894. ISSN: 0360-5442. DOI: 10.1016/j.energy.2020.117894. URL: http://dx.doi.org/10.1016/j.energy.2020.117894.

[20] Lilin Cheng et al. "Multi-Meteorological-Factor-Based Graph Modeling for Photovoltaic Power Forecasting". In: *IEEE Transactions on Sustainable Energy* 12.3 (July 2021), pp. 1593–1603. ISSN: 1949-3037. DOI: 10.1109/tste.2021.3057521. URL: http://dx.doi.org/10.1109/tste.2021.3057521.

[21] Deniz Korkmaz. "SolarNet: A hybrid reliable model based on convolutional neural network and variational mode decomposition for hourly photovoltaic power forecasting". In: *Applied Energy* 300 (Oct. 2021), p. 117410. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2021.117410. URL: http://dx.doi.org/10.1016/j.apenergy.2021.117410.

[22] Kejun Wang, Xiaoxia Qi, and Hongda Liu. "A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network". In: *Applied Energy* 251 (Oct. 2019), p. 113315. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2019.113315. URL: http://dx.doi.org/10.1016/j.apenergy.2019.113315.

[23] Roozbeh Rajabi and Abouzar Estebsari. "Deep Learning Based Forecasting of Individual Residential Loads Using Recurrence Plots". In: *2019 IEEE Milan PowerTech*. IEEE, June 2019, pp. 1–5. DOI: 10.1109/ptc.2019.8810899. URL: http://dx.doi.org/10.1109/ptc.2019.8810899.

[24] Fath U Min Ullah et al. “Short-Term Prediction of Residential Power Energy Consumption via CNN and Multi-Layer Bi-Directional LSTM Networks”. In: *IEEE Access* 8 (2020), pp. 123369–123380. ISSN: 2169-3536. DOI: 10.1109/access.2019.2963045. URL: http://dx.doi.org/10.1109/access.2019.2963045.

[25] Ijaz Ul Haq et al. “Sequential Learning-Based Energy Consumption Prediction Model for Residential and Commercial Sectors”. In: *Mathematics* 9.6 (Mar. 2021), p. 605. ISSN: 2227-7390. DOI: 10.3390/math9060605. URL: http://dx.doi.org/10.3390/math9060605.

[26] Tae-Young Kim and Sung-Bae Cho. “Predicting residential energy consumption using CNN-LSTM neural networks”. In: *Energy* 182 (Sept. 2019), pp. 72–81. ISSN: 0360-5442. DOI: 10.1016/j.energy.2019.05.230. URL: http://dx.doi.org/10.1016/j.energy.2019.05.230.

[27] Mohamed Abdel-Basset et al. “STLF-Net: Two-stream deep network for short-term load forecasting in residential buildings”. In: *Journal of King Saud University - Computer and Information Sciences* 34.7 (July 2022), pp. 4296–4311. ISSN: 1319-1578. DOI: 10.1016/j.jksuci.2022.04.016. URL: http://dx.doi.org/10.1016/j.jksuci.2022.04.016.

[28] Tao Han et al. “An Efficient Deep Learning Framework for Intelligent Energy Management in IoT Networks”. In: *IEEE Internet of Things Journal* 8.5 (Mar. 2021), pp. 3170–3179. ISSN: 2372-2541. DOI: 10.1109/jiot.2020.3013306. URL: http://dx.doi.org/10.1109/jiot.2020.3013306.

[29] Jin-Young Kim and Sung-Bae Cho. “Electric Energy Consumption Prediction by Deep Learning with State Explainable Autoencoder”. In: *Energies* 12.4 (Feb. 2019), p. 739. ISSN: 1996-1073. DOI: 10.3390/en12040739. URL: http://dx.doi.org/10.3390/en12040739.

[30] Zulfiqar Khan et al. “Towards Efficient Electricity Forecasting in Residential and Commercial Buildings: A Novel Hybrid CNN with a LSTM-AE based Framework”. In: *Sensors* 20.5 (Mar. 2020), p. 1399. ISSN: 1424-8220. DOI: 10.3390/s20051399. URL: http://dx.doi.org/10.3390/s20051399.

[31] Gangqiang Li et al. “Recurrent Neural Networks Based Photovoltaic Power Forecasting Approach”. In: *Energies* 12.13 (July 2019), p. 2538. ISSN: 1996-1073. DOI: 10.3390/en12132538. URL: http://dx.doi.org/10.3390/en12132538.

[32] Gabriel Trierweiler Ribeiro et al. “Bayesian Optimized Echo State Network Applied to Short-Term Load Forecasting”. In: *Energies* 13.9 (May 2020), p. 2390. ISSN: 1996-1073. DOI: 10.3390/en13092390. URL: http://dx.doi.org/10.3390/en13092390.

[33] Shouxiang Wang et al. “Bi-directional long short-term memory method based on attention mechanism and rolling update for short-term load forecasting”. In: *International Journal of Electrical Power; Energy Systems* 109 (July 2019), pp. 470–479. ISSN: 0142-0615. DOI: 10.1016/j.ijepes.2019.02.022. URL: http://dx.doi.org/10.1016/j.ijepes.2019.02.022.

[34] Yixing Wang et al. “Short-Term Load Forecasting with Multi-Source Data Using Gated Recurrent Unit Neural Networks”. In: *Energies* 11.5 (May 2018), p. 1138. ISSN: 1996-1073. DOI: 10.3390/en11051138. URL: http://dx.doi.org/10.3390/en11051138.

[35] Zulfiqar Ahmad Khan et al. “Efficient Short-Term Electricity Load Forecasting for Effective Energy Management”. In: *Sustainable Energy Technologies and Assessments* 53 (Oct. 2022), p. 102337. ISSN: 2213-1388. DOI: 10.1016/j.seta.2022.102337. URL: http://dx.doi.org/10.1016/j.seta.2022.102337.

[36] Meng Zhang. “Load Analysis and Prediction of Electric Vehicle Charging Stations Supported by RF-CNN Algorithm”. In: *Informatica* 48.19 (2024), pp. 89–102. DOI: 10.31449/inf.v48i19.6649.

[37] Junhong Kim et al. “Recurrent inception convolution neural network for multi short-term load forecasting”. In: *Energy and Buildings* 194 (July 2019), pp. 328–341. ISSN: 0378-7788. DOI: 10.1016/j.enbuild.2019.04.034. URL: http://dx.doi.org/10.1016/j.enbuild.2019.04.034.

[38] Noman Khan et al. “AB-Net: A Novel Deep Learning Assisted Framework for Renewable Energy Generation Forecasting”. In: *Mathematics* 9.19 (Oct. 2021), p. 2456. ISSN: 2227-7390. DOI: 10.3390/math9192456. URL: http://dx.doi.org/10.3390/math9192456.

[39] Yong Zhang, Yueda Gao, and Zhe Zhao. “Research on Operation and Anomaly Detection of Smart Power Grid Based on Information Technology Using CNN+Bidirectional LSTM”. In: *Informatica* 49.7 (2025), pp. 157–164. DOI: 10.31449/inf.v49i7.7037.

[40] Muhammad Sajjad et al. “A Novel CNN-GRU-Based Hybrid Approach for Short-Term Residential Load Forecasting”. In: *IEEE Access* 8 (2020), pp. 143759–143768. ISSN: 2169-3536. DOI: 10.1109/access.2020.3009537. URL: http://dx.doi.org/10.1109/access.2020.3009537.

[41] Fei Wang et al. “Wavelet Decomposition and Convolutional LSTM Networks Based Improved Deep Learning Model for Solar Irradiance Forecasting”. In: *Applied Sciences* 8.8 (Aug. 2018), p. 1286. ISSN: 2076-3417. DOI: 10.3390/app8081286. URL: http://dx.doi.org/10.3390/app8081286.

[42] Jingxuan Liu et al. “A Transformer-based multimodal-learning framework using sky images for ultra-short-term solar irradiance forecasting”. In: *Applied Energy* 342 (July 2023), p. 121160. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2023.121160. URL: http://dx.doi.org/10.1016/j.apenergy.2023.121160.

[43] Hegang Zong et al. “A Multitask Framework for Optimizing Smart Grid Energy Consumption Using RegClassXNet and Dynamic Cluster Adjustment”. In: *Informatica* 49.18 (2025), pp. 127–142. DOI: 10.31449/inf.v49i18.7865.

[44] Tomás Cabello-López et al. “Forecasting solar energy production in Spain: A comparison of univariate and multivariate models at the national level”. In: *Applied Energy* 350 (Nov. 2023), p. 121645. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2023.121645. URL: http://dx.doi.org/10.1016/j.apenergy.2023.121645.

[45] Chengcheng Jiang and Qunzhi Zhu. “Evaluating the most significant input parameters for forecasting global solar radiation of different sequences based on Informer”. In: *Applied Energy* 348 (Oct. 2023), p. 121544. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2023.121544. URL: http://dx.doi.org/10.1016/j.apenergy.2023.121544.

[46] Ling-Ling Li et al. “Renewable energy prediction: A novel short-term prediction model of photovoltaic output power”. In: *Journal of Cleaner Production* 228 (Aug. 2019), pp. 359–375. ISSN: 0959-6526. DOI: 10.1016/j.jclepro.2019.04.331. URL: http://dx.doi.org/10.1016/j.jclepro.2019.04.331.

[47] Pengtao Li et al. “A hybrid deep learning model for short-term PV power forecasting”. In: *Applied Energy* 259 (Feb. 2020), p. 114216. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2019.114216. URL: http://dx.doi.org/10.1016/j.apenergy.2019.114216.

[48] Kejun Wang, Xiaoxia Qi, and Hongda Liu. “Photovoltaic power forecasting based LSTM-Convolutional Network”. In: *Energy* 189 (Dec. 2019), p. 116225. ISSN: 0360-5442. DOI: 10.1016/j.energy.2019.116225. URL: http://dx.doi.org/10.1016/j.energy.2019.116225.

[49] Zulfiqar Ahmad Khan et al. “Electrical Energy Prediction in Residential Buildings for Short-Term Horizons Using Hybrid Deep Learning Strategy”. In: *Applied Sciences* 10.23 (Dec. 2020), p. 8634. ISSN: 2076-3417. DOI: 10.3390/app10238634. URL: http://dx.doi.org/10.3390/app10238634.

[50] Zulfiqar Ahmad Khan et al. “A Trapezoid Attention Mechanism for Power Generation and Consumption Forecasting”. In: *IEEE Transactions on Industrial Informatics* 20.4 (Apr. 2024), pp. 5750–5762. ISSN: 1941-0050. DOI: 10.1109/tii.2023.3335453. URL: http://dx.doi.org/10.1109/tii.2023.3335453.

Enhancing Supply Chain Demand Forecasting Through Gated Graph Neural Networks and Federated Learning Systems

Zhicheng Xu
School of Bussiness Administration, Zhejiang Gongshang University, Zhejiang 310018, China
Email: ZhichenggXuu@outlook.com

Keywords: Gated graph neural network, Federated learning, Supply chain demand forecasting, Data privacy protection, Forecast accuracy improvement

Received: July 21, 2025

In the context of increasingly complex global supply chains, accurate demand forecasting is crucial for companies to optimize inventory and reduce costs. However, traditional forecasting methods are often difficult to effectively capture complex interactions in the supply chain, and data privacy protection has also become a major challenge. Therefore, this study proposes a demand forecasting method for enterprise supply chains based on gated graph neural and federated learning. By constructing a gated graph neural network, the dynamic relationship of each link in the supply chain is successfully captured. The experimental results show that the prediction accuracy of this model is improved by 12% compared with the traditional method. In order to further strengthen data privacy protection, we have introduced a federated learning mechanism to realize model training without data leaving the local area. Experiments show that the performance of the model under the federated learning framework is only 4% lower than that of centralized training. Combining the advantages of both, we have built a new forecasting system. When processing large-scale and complex supply chain data, the forecasting accuracy rate is 8% higher than that of a single model while effectively protecting data privacy. This study not only provides a new technical path for modern enterprise supply chain management but also lays a solid foundation for intelligent and efficient supply chain management in the future.

Povzetek: Študija predstavi metodo napovedovanja povpraševanja, ki z grafnimi nevronskimi mrežami in zveznim učenjem izboljša natančnost napovedi ter hkrati učinkovito varuje podatke.

1 Introduction

Driven by the tide of globalization, the supply chain of enterprises is increasingly complex and changeable. As the core link of supply chain management, the accuracy and timeliness of demand forecasting are directly related to the operational efficiency and market competitiveness of enterprises [1]. However, traditional demand forecasting methods are often subject to problems such as data islands and weak model generalization ability, and it is difficult to adapt to the rapidly changing market environment [2, 3]. Against this background, the demand forecasting of enterprise supply chains based on gated graph neural and federated learning came into being, which provided a new idea to solve the above problems.

A gated graph neural network, as an advanced form of graph neural network, effectively improves the ability of the model to capture complex relationships and dynamic changes by introducing a gating mechanism [4]. In the enterprise supply chain, all links are interrelated, forming an intricate network structure. Gated graph neural networks can dig deep into the potential information in this structure so as to predict the demand change more accurately [5]. At the same time, federated learning, as an emerging distributed machine learning framework, can realize collaborative training of multi-party data on the premise of ensuring data privacy [6]. This provides strong data support for enterprise supply chain demand forecasting and breaks the data island limitation in traditional methods.

With the rapid development of big data, cloud computing and other technologies, enterprises have accumulated massive supply chain data [7, 8]. These data contain rich information such as market demand, logistics dynamics, and inventory status, which provide valuable resources for demand forecasting [9]. However, how to effectively utilize these data and tap the value behind them has become a major challenge for enterprises. Methods based on gated graph neural and federated learning are innovative solutions to this challenge. Through this method, enterprises can not only make full use of their own data but also share data resources with other enterprises or institutions on the basis of protecting data privacy so as to jointly improve the accuracy of demand forecasting [10].

The existing research on supply chain demand forecasting is mainly divided into two categories: one is the application of focused graph neural network (GNN), for example, some studies use traditional GNN or graph convolutional network (GCN) to capture the relationship between goods, suppliers, and customers in the supply chain, but this method mostly relies on centralized data

training and does not consider the data privacy protection needs of various supply chain participants. Although data privacy protection is achieved through distributed training, such as demand prediction models based on federated linear regression and federated LSTM, these models mostly regard supply chain data as linear sequences or independent samples, and cannot effectively characterize the complex correlations between goods and the topology of supply chain networks, resulting in limited prediction accuracy due to data characterization capabilities. The combined scheme of GGNN (Gated Graph Neural Network) combined with federated learning adopted in this study has significant novelties compared with existing studies: on the one hand, compared with the study using GNN alone, this scheme introduces a federated learning framework to realize distributed model training without aggregating the raw data of each participant, which solves the core pain points of supply chain data privacy leakage; On the other hand, compared with the research that uses federated learning but relies on traditional time series models, this scheme dynamically updates the node state through the gating mechanism of GGNN, which can not only accurately capture the topological correlation of the supply chain network, but also strengthen the fitting ability to fluctuate in time series demand. It provides a technical path that is more suitable for practical application scenarios for enterprise supply chain demand forecasting.

In practical applications, enterprise supply chain demand forecasting based on gated graph neural and federated learning has shown significant advantages [11, 12]. By analyzing multi-dimensional data such as consumer purchasing behaviour and inventory turnover rate, this method can accurately predict commodity demand in the future, help retailers optimize inventory management, and reduce shortages and backlogs. In the manufacturing industry, this method can predict changes in product demand according to market demand, raw material price fluctuations and other factors, and provide a strong basis for enterprises to formulate production plans. The method based on gated graph neural and federated learning also has strong robustness and adaptability. In the face of uncertain factors such as unexpected events and market fluctuations, this method can quickly adjust the model parameters, adapt to the new market environment, and ensure that the accuracy of the demand forecast is not affected. This feature is of great significance for enterprises to cope with the complex and changeable market environment.

This paper takes enterprise supply chain demand forecasting as the research core, and aims to develop a federated learning framework for integrated GGNN. The main contribution is to propose a new integration method of GGNN and federated learning to effectively improve the accuracy of supply chain demand forecasting, and at the same time design a data privacy protection mechanism suitable for distributed supply chain scenarios, taking into account forecasting performance and data security.

2 Theoretical basis of gated graph neural and federated learning

2.1 Principles of gated graph neural network

Traditional deep learning models such as convolutional neural networks, recurrent neural networks, and generative adversarial networks are mainly oriented to European data (such as images, speech, and text) with regular spatial structures [13]. Such data can be expressed by one-dimensional or two-dimensional matrices, and the samples are independent of each other by default. However, a large amount of data exists in non-Euclidean graph structures in real scenes, such as social networks, chemical molecular structures and knowledge graphs, which describe complex relationships between entities through graphs G = (V, E) composed of vertex set V and edge set E [14, 15].

Aiming at the processing requirements of graph structure data, a graph neural network (GNN) came into being [16]. Its core mechanism is to iteratively aggregate the information of target nodes and their neighbours to generate node embedding vectors containing graph structural features [17, 18]. The operation of GNN includes three stages: firstly, the original data is abstracted into a graph structure, and a relationship matrix is constructed. Then, the node features are diffused in the graph through the information propagation and update mechanism. Finally, the structured results are output according to the task requirements [19]. This design enables GNN to show powerful representation learning capabilities in social network analysis, knowledge graph reasoning and other fields.

The early GNN was based on the fixed-point theory, which needed to iterate to the convergent state to output the results and did not explicitly model the side information. For this reason, the proposed gated graph neural network (GGNN) introduces the gated recurrent unit (GRU) to optimize the information transfer process and preserve the key features in node iteration through the selective memory mechanism [20, 21]. Compared with the traditional GNN, GGNN can complete the calculation with a fixed step size without forced convergence and, at the same time, supports the processing of multi-type edge relationships through edge weight allocation, which significantly improves the model's adaptability to complex graph structures [22]. Each node in the graph updates its status through GGNN, receiving and transmitting information to neighboring nodes. The GGNN architecture is shown in Figure 1.

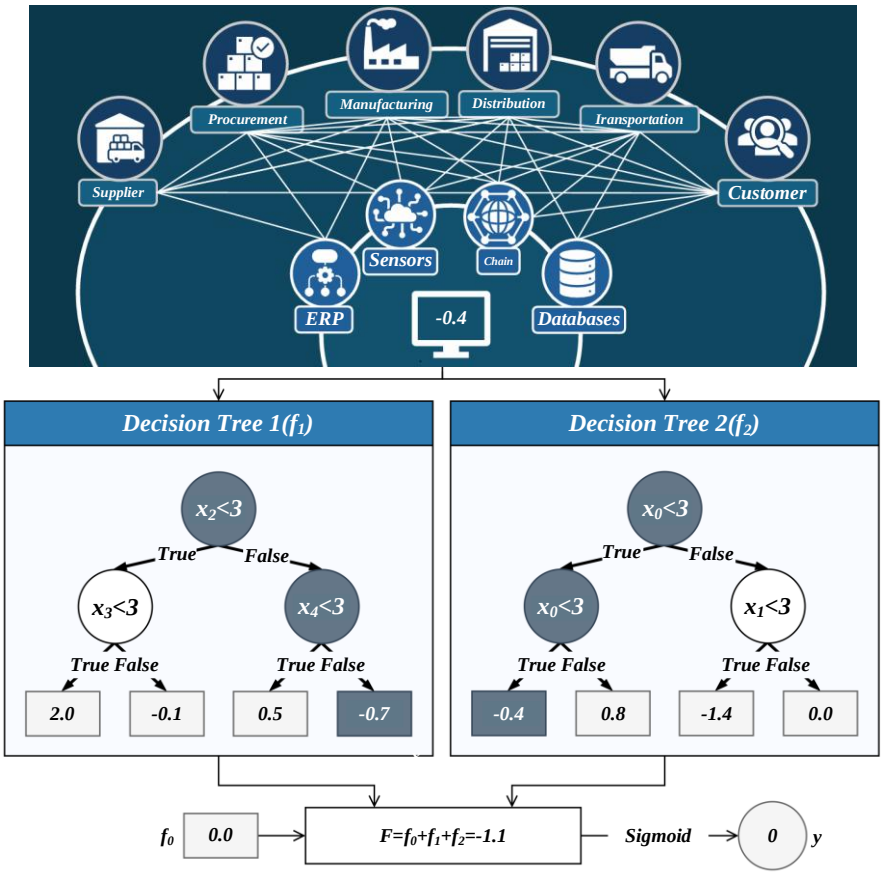


Figure 1: GGNN architecture

First, the GGNN integrates the connection data around the target node through the adjacency matrix, following equations (1) to (2).

$$h_v^{(1)} = [x_v^T, 0]^T \quad (1)$$

$$a_v^{(t)} = A_{v:}^T [h_1^{(t-1)T}, h_2^{(t-1)T}, \ldots, h_{|v|}^{(t-1)T}]^T + b \quad (2)$$

In formula (1), the d-dimensional initial embedding vector $h_v^{(l)}$ of node v represents the item characteristics through the feature channel, and is expanded by zero filling when the input feature x_v^T dimension is less than d. The relationship matrix A_v in formula (2) extracts the adjacency information associated with node v, and aggregates and generates the hidden state $a_v^{(t)} \in R^{2d}$ representing the neighbor information by splicing the node features $[h_l^{(t-1)T}, h_2^{(t-1)T}, \ldots, h_{|v|}^{(t-1)T}]$ of the whole graph at t-1 time. GGNN adopts GRU mechanism to update node status, and dynamically regulates the memory and forgetting of timing information by updating gates and resetting gates, effectively alleviating the long-term dependence problem of traditional RNN [23]. Equations (3)-(6) specifically show how GRU fuses the current node state and aggregated information to achieve iterative updates of node representations.

$$z_v^t = \sigma(W_z a_v^{(t)} + U_z h_v^{(t-1)} + b_z) \quad (3)$$

$$r_v^t = \sigma(W_r a_v^{(t)} + U_r h_v^{(t-1)} + b_r) \quad (4)$$

$$\tilde{h}_v^{(t)} = tanh(W a_v^{(t)} + U(r_v^t \odot h_v^{(t-1)})) \quad (5)$$

$$h_v^{(t)} = (1 - z_v^t) \odot h_v^{(t-1)} + z_v^t \odot \tilde{h}_v^{(t)} \quad (6)$$

At time t, the node aggregates proximity information $a_v^{(t)}$ as input and its own state $h_v^{(t-1)}$ as a hidden state. The update gate z_v^t and the reset gate r_v^t control forgetting and new information generation in information propagation, respectively. In Equation (5), $\tilde{h}_v^t$ represents new information, and in Equation (6), h_v^t represents the final embedding vector of node v. The GGNN model is updated by multi-layer iteration, and two output methods are adopted: node-based and whole-graph-based. The output formula of the embedding representation vector $h_v^{(T)}$ for each node v is shown in Equation (7).

$$o_v = g(h_v^{(T)}, x_v) \quad (7)$$

In equation (7), $h_v^{(T)}$ and x_v represent the start and end states of node v, respectively, and $g(\cdot)$ is a specific function. The output result of the figure is shown in Equation (8).

$$h_G = tanh(\sum_{v \in V} \sigma(i(h_v^{(T)}, x_v)) \odot tanh(j(h_v^{(T)}, x_v))) \quad (8)$$

In Equation (8), $\sigma(i(h_v^{(T)}, x_v))$ is used to determine the

rate of node update, where i and j represent the neural network index.

2.2 Federated learning foundations

Federated learning is a distributed machine learning framework that allows multiple participants to collaboratively train a shared global model on the premise of protecting data security [24]. Its core is that there is no need to store all data centrally. Each participant only trains the local model based on local data and shares the model parameters through the central server instead of the original data, thereby ensuring data privacy [25]. In traditional machine learning, the server needs to collect the private data $\{D_1, ..., D_n\}$ of all clients for unified training to obtain the global model Mgoo, while in federated learning, each client $\{u_1, ..., u_n\}$ only uploads the local model parameters, and the server generates the global model M by aggregating these parameters, thus avoiding the risk of direct data exposure.

In the federated learning system, the central server usually coordinates multiple client devices (such as computers, tablets, mobile phones, etc.) to jointly train the model [26, 27]. The specific process is as follows: the server first selects the clients participating in the training initializes the global model, and then distributes the model parameters to all participants. The client trains the model based on local data and returns the updated parameters to the server. The server aggregates the parameters of all clients to optimize the global model and distributes it to the next round of participating clients for iterative training [28]. This process is repeated for multiple rounds until the model converges, and finally, the server distributes the trained model to all clients. In this way, federated learning achieves the goal of leveraging distributed devices to collaboratively train efficient models while protecting data privacy. The federated averaging algorithm is the core of federated learning, and its objective function is defined by formulas (9) and (10).

$$\min_{\omega \in R^d} f(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w) \quad (9)$$

$$F_k(w) = \frac{1}{n_k} \sum_{i \in P_k} f_i(s_i; w_i) \quad (10)$$

Assuming that k clients participate in the training, $F_k(w)$ and P_k in formula (9) represent the local loss function and training data set of the k-th client, respectively. The dataset size is $n_k = |P_k|$, where s_i is the sample in the dataset, w is the weight matrix, and $f_i(s_i, w_i)$ is the loss value of the model on the sample s_i. A partial update of the model on the client is called an iteration, and a batch is denoted by b. The local model iterative process of federated learning on the k-th client is shown in Equation (11).

$$w_k \leftarrow w_k - \frac{\eta}{|b|} \sum_{i \in b} \nabla f(s_i; \omega_i) \quad (11)$$

The global model is obtained by weighted summation of each local model, where w^k_{t+1} represents the local model returned by client k after the t-th round. The polymerization process is shown in Equation (12).

$$w_{t+1} = \sum_{k \in S_t} \frac{n_k}{n} w^k_{t+1} \quad (12)$$

2.3 The connection between gated graph neural and federated learning

Gated graph neural network (GGNN) and federated learning are two promising technologies that provide ideas for improving prediction performance and solving practical problems from different dimensions - GGNN can accurately characterize the complex correlations between suppliers, manufacturers, distributors and other nodes in the supply chain with its powerful modeling capabilities on graph structure data, capture the dynamic interaction and dependencies of nodes through the gating mechanism, fully mine the hidden association information in the network, and provide structured feature representation for prediction. However, supply chain data is often scattered across different enterprises and involves sensitive information, and the need for data privacy protection makes it difficult to apply centralized training GGNN, and federated learning solves this pain point through the "data does not move model" model, allowing all participants to retain data locally while sharing model parameters to achieve collaborative training, which not only protects privacy but also breaks data silos. When the two are organically integrated, under the framework of federated learning, each supply chain node can construct a graph structure based on local data and use GGNN for feature extraction and training, and only upload model parameters or gradients to the central server, and the server integrates parameters through the aggregation algorithm and then distributes them back to each node. Ultimately, it significantly improves the accuracy and practicality of supply chain demand forecasting under the premise of protecting privacy, and provides support for efficient collaboration and accurate decision-making.

2.4 Related work

Although traditional enterprise supply chain demand forecasting methods rely on time series analysis or traditional machine learning models, although they have certain effects in stable data scenarios, it is difficult to capture complex node associations and dynamic dependencies in supply chain networks, and the adaptability to multi-source heterogeneous order/spatial data is insufficient. With its gating mechanism and graph structure modeling capabilities, GGNN has shown significant advantages in the temporal dependency mining of sequential data and the topological relationship

representation of spatial data, which provides a technical foundation for solving the demand transmission modeling of multi-node and multi-link in the supply chain network. In addition, federated learning, as a key technology to protect data privacy in distributed scenarios, has been deployed in financial risk control, medical imaging, and other fields, but in industrial supply chain scenarios, it still faces problems such as multi-agent data heterogeneity, communication delay, and federated model update efficiency. Taken together, there are three core gaps in the existing research: first, traditional forecasting methods cannot effectively integrate supply chain graph structure information and multi-source time series data, second, the application of GGNN in supply chain demand forecasting lacks customized optimization for industrial scenarios, and third, federated learning has not yet formed a solution suitable for multi-agent and highly dynamic scenarios in the distributed deployment of industrial supply chains Enterprise supply chain demand forecasting system with graph structure modeling advantages and federated learning privacy protection capabilities.

3 Prediction model based on gated graph neural and federated learning

3.1 Model architecture and workflow

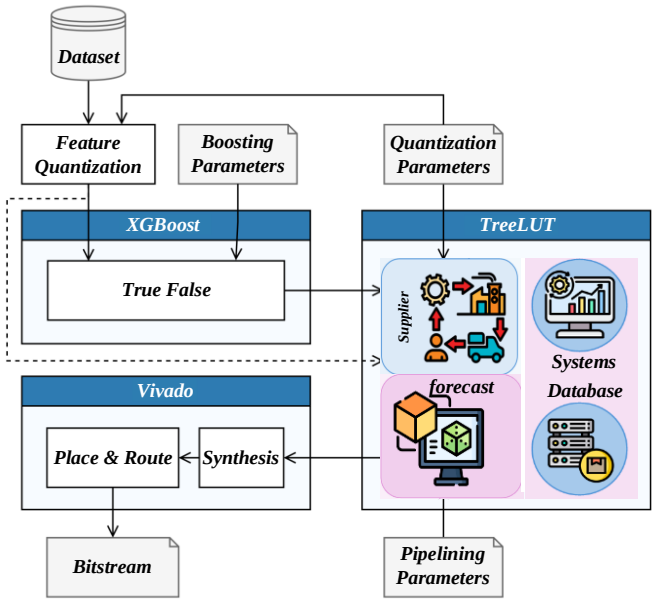


Figure 2: Prediction model based on gated graph neural and federated learning

Based on the two-stage forecast results of production and sales output by the forecast algorithm, the actual supply of multiple downstream enterprises can be calculated. Upstream suppliers and downstream enterprises are regarded as two kinds of nodes, and a bilateral model is established with goods supply and purchase quantity as independent variables and goods price, purchase price and enterprise reputation as dependent variables, and both parties are assumed to be rational subjects to maximize profits. Taking the three-layer supply chain network FL-GNN model as an example, the network includes the producer PL layer (N_{PL} nodes), the distributor DL layer (N_{DL} nodes) and the retailer SL layer (N_{SL} nodes), forming a directed graph

Supply chain resource forecasting involves the forecasting planning of production and sales resource demand, including the forecasting of upstream production resources and downstream sales resources. Upstream enterprises need to forecast production volume to meet demand, while downstream enterprises need to forecast sales volume to adapt to the market [29, 30].

Upstream resource forecasting needs to consider factors such as market trends, raw material reserves and historical production conditions to improve forecasting accuracy and decision-making quality. Downstream resource forecasting relies on historical sales data, including quantity, time, location, and customer information, to optimize supply planning and inventory management [31, 32].

Data is critical in resource forecasting. Federated learning technology allows enterprises to share data while protecting data privacy and improving prediction accuracy. In this paper, a demand forecasting algorithm based on gated graph neural network and federated learning is introduced, which can aggregate independent identically distributed (IID) data sets and solve the problem of inaccurate forecasting caused by small sample size. Figure 2 illustrates this prediction model.

with *N-1* (N_{PL} + N_{DL} + N_{SL}) nodes. Each node has the warehousing function of storing *M*-type goods, where node 0 represents the starting point of all PL layer manufacturers, and node *N* represents the end point of all *SL* layer retailers. *D(i)(D(i)∈M)* is defined as the type index set of available goods on node *i*, based on which a three-layer FL-GNN discrete model can be established, as shown in formula (13).

$$x_{il}(t+1)=x_{il}(t)+u_{0il}(t-\tau_{0il})-\sum_{j\in O_l(i)}u_{ijl}(t),i\in PL,l\in D(i)$$
$$x_{il}(t+1)=x_{il}(t)+\sum_{j\in I_l(i)}u_{ijl}\left(t-\tau_{ijl}\right)-\sum_{j\in O_l(i)}uijl(t),i\in DL,l\in D(i) \quad (13)$$

Where $t = 0, 1, 2,...$ is the timestamp, $x_{il}(t)$ is the actual inventory storage of goods l ($l \in D(i)$) by node i at timestamp t, and $u_{ijl}(t)$ is the amount of goods transported by goods l from node i to node j at timestamp t. $I(j)$ represents the import volume of goods from node i to node j, $O(j)$ represents the export volume of goods from node j to node k, $I_l(I) = \{j: j \in I(i)$ and $l \in D(I) \cap D(j)\}$ represents the import volume of goods l, $O_l(I) = \{j: j \in I(i)$ and $l \in D(i) \cap D(j)\}$ represents the export volume of goods l, τ_{ijl} represents the time delay of transportation from node i to node j, and w_{il} is the special demand of node i for goods l. $y_{il}(t)$ represents the backlog of goods demand for goods l at node i.

For the enterprise supply chain demand forecasting model based on gated graph neural network (GGNN) and federated learning, this paper designs a Bayesian optimization strategy based on 5-fold cross-validation, and clarifies the architectural details of GGNN. In terms of hyperparameter tuning, the average absolute error (MAE) on the validation set is the core optimization goal, and the training time constraint is introduced to balance model performance and efficiency, and Bayesian optimization is implemented using the Optuna framework. The search space includes model structure parameters, learning process parameters, and aggregation parameters unique to federated learning (local update round, federated communication cycle). The historical data of the supply chain is divided through 5 layers of cross-verification, and the demand data of different time periods is retained as a test set for each verification to ensure the adaptability of the optimization results to the characteristics of the time series, and finally the combination of parameters that make the cross-validation MAE the smallest is selected as the optimal configuration.

The joint learning link faces significant privacy risks: on the one hand, it is a model reversal attack, where the attacker can use the local model parameters or intermediate results uploaded by each participant in federated learning, combined with the public business characteristics in the supply chain scenario, to reverse derive the sensitive private data of the participants; On the other hand, there is the risk of gradient leakage, which will imply the characteristics of a large amount of private training data in the gradient information uploaded by each participant after local training, and the iterative training process of federated learning will continue to accumulate the risk of gradient leakage, resulting in the impact of privacy leakage expanding with the deepening of training rounds. In response to the above risks, the framework adopts preliminary mitigation measures: by adding differential privacy to the local model gradient, carefully controlled noise is introduced before gradient upload, making it difficult for attackers to accurately extract private data features from the noisy gradient, and ensuring that the noise scale does not significantly affect the accuracy of the gated graph neural model for supply chain demand forecasting [33, 34]. At the same time, gradient cropping is implemented to limit the abnormally large gradient generated during the local model training process, avoid carrying too many sensitive data details due to extreme gradient values, further reduce the possibility of gradient leakage, and provide basic privacy protection for the private business data of supply chain participants.

3.2 Limitations and solutions

The enterprise supply chain demand forecasting scheme based on gated graph neural network (gated GNN) and federated learning still has three significant limitations in practical application: First, at the data processing level, although the gated graph neural network has good feature extraction capabilities for structured supply chain node relationships , it is still capable of highly unstructured data in supply chain scenarios , the lack of effective preprocessing and feature mapping mechanisms can easily lead to a decrease in the quality of model input due to the confusion of data format and the difficulty of quantifying semantic information, which in turn affects the prediction accuracy [35]. If some customers withdraw from federated training due to data privacy protection concerns, limited device computing resources, or changes in willingness to cooperate, it will lead to insufficient local model sample size and imbalance in data distribution during global model aggregation, which not only increases the iteration cost of model training, but may also cause the problem of "federated degradation" and weaken the overall prediction stability. Third, in terms of the completeness of extreme scenario verification, the existing experimental verification is mostly based on conventional supply chain fluctuation scenarios, while the simulation and verification of extreme supply chain interruption events are insufficient. It is difficult to fully meet the actual needs of enterprises to deal with sudden risks.

In view of the above limitations, a solution path can be constructed from three aspects: at the level of unstructured data processing, multi-modal preprocessing modules and cross-modal feature fusion technologies can be introduced, such as using BERT-class pre-training models to semantically encode logistics scheduling text, and extracting visual features of equipment fault images through convolutional neural networks (CNNs). Then, with the help of the attention mechanism, the feature vectors of unstructured data are aligned and fused with the structured node features processed by the gated graph neural network, and an adaptive data cleaning algorithm is designed to correct the irregular demand feedback of

manual entry to improve the quality of input data from the source. On the other hand, the "edge node backup" and "dynamic weight aggregation" strategies are introduced, and when some customers withdraw, the historical similarity data of the standby edge nodes is called to supplement the local training sample, and the aggregation weight is dynamically adjusted according to the differences in the data distribution of each participant, so as to avoid the occurrence of the global model due to data imbalance. federal degradation"; In terms of verification and improvement of extreme scenarios, a multi-dimensional extreme interruption scenario simulation library can be built to generate virtual extreme scene samples based on historical extreme event data (Monte Carlo simulation, and the model's ability to capture extreme features is strengthened through the two-stage training mode of "conventional scenario pre-training and extreme scenario fine-tuning" to ensure that it can meet the needs of enterprises to cope with sudden risks.

3.2 The role of gated graph neural in feature extraction

In the graph construction stage, the enterprise nodes in the supply chain are taken as the vertex, and the interaction between material flow, capital flow and information flow is the directed edge, and the edge weight is determined by the normalized value of historical transaction frequency and transaction volume. The gated recursive unit (GRU) parameter is set to hidden layer dimension 64, and the sigmoid activation function is used for reset gates and update gates, and the tanh activation function is used for candidate states to enhance the capture ability of long-term supply chain dependencies. The message delivery mechanism adopts the attention mechanism-weighted neighbor information aggregation, that is, each node dynamically allocates the message weight of the neighbor node according to the historical interaction importance, and filters the noise information through the gating unit. The layer stack is configured as a 5-layer deep architecture, the underlying layer learns the local characteristics of nodes, the middle layer captures the supply chain subnetwork collaboration mode, the top-layer integrates the global supply and demand relationship, and finally outputs the demand prediction value of each node through the full connection layer.

Feature extraction in gated graph neural network is directly related to the performance of prediction model. GGNN, namely gated graph neural network, can efficiently extract the feature information of nodes in graph data and the association information between nodes through its unique gating mechanism. This ability makes GGNN show a strong advantage when dealing with data with complex network structure such as supply chain.

First, the GGNN can flexibly control the transmission and forgetting of information by utilizing the GRU, the update gate and the reset gate mechanism of the gated loop unit. In the supply chain network, the state changes of nodes are often affected by many factors, including historical transaction records, cooperative relationships among nodes, market trends, etc. Through the gating mechanism, GGNN can selectively retain important information while forgetting irrelevant information, thus accurately capturing the dynamic characteristics of nodes. GGNN realizes the efficient modelling of edge information by setting weights to different types of edges. In the supply chain network, edges represent transactions or cooperative relationships between nodes, and different types of edges may have different importance. By assigning different weights to different types of edges, GGNN can accurately reflect the differences in the importance of these relationships, thus enhancing the predictive power of the model.

The multi-layer iterative update mechanism of GGNN enables the model to gradually dig deep into the potential features of graph data. In each layer of iteration, GGNN will update the status of the nodes according to the information of adjacent nodes. This iterative process helps the model to gradually capture deeper feature information and improve the accuracy and stability of prediction. Gated graph neural network plays a vital role in the prediction model of gated graph neural and federated learning. Its powerful feature extraction ability enables the model to accurately capture the complex relationships in the supply chain network, which provides strong support for the demand forecasting of enterprises.

4 Experiment and results analysis

Based on Yoochoose1/64 and Diginetica, the missing values of the two datasets (such as some missing user behavior labels in Yoochoose1/64 and incomplete encoding of a small number of commodity categories in Diginetica) are processed by means filling and pattern filling in the data preprocessing stage, and the outliers are processed by 3σ In principle, the time series data is screened and eliminated, and the time series data is segmented with a sliding window to construct the training sample. In terms of feature engineering, in addition to embedding and coding discrete features such as product ID and user ID, statistical features such as product click frequency, user behavior sequence length, and product category correlation are also extracted, and time series features are constructed in combination with time attenuation factors, and dimensionality reduction of high-dimensional features is reduced by principal component analysis (PCA) to reduce redundancy. In terms of hyperparameter selection, the combination of grid search and 5-fold cross-validation is adopted, and the key hyperparameters such as the hidden layer dimension (128/256), the gating unit activation function (ReLU/Sigmoid), the number of graph convolution iteration steps (3/5), the local training rounds (2/3) and the learning rate (0.001/0.005) of federated learning are optimized by using the combination of grid search and 5-fold cross-validation. The determination of GGNN structure is based on the correlation between commodity-user-category in the supply chain, taking commodity and user as graph nodes, commodity category as node attributes, constructing edge weights between nodes

according to the interaction frequency of user behavior, dynamically updating the node state through the gating mechanism to capture complex dependencies in the supply chain, and finally combined with the distributed training framework of federated learning.

According to Table 1, the P @ 20 indicator of the FL-GNN model in the Diginetica dataset is lower than that of the S ²-DHCN model, but its prediction performance is better than that of the other models on the other two datasets.

Table 1: Experimental comparison results between FL-GNN and benchmark model on two data sets

Models	Yoochoose1/64		Diginetica	
	P @ 20	MRR @ 20	P @ 20	MRR @ 20
Item-KNN	54.18	22.9005	37.5375	12.1485
FPMC	47.901	15.7605	27.8565	7.2975
GRU4Rec	63.672	24.0345	30.9225	8.1165
NARM	71.736	30.0615	52.185	16.9785
STAMP	72.177	31.1535	47.922	15.036
CSRM	73.6995	31.878	49.896	17.199
SR-GNN	74.0985	32.487	53.2665	18.4695
TA-GNN	74.1825	32.5185	54.033	18.7425
S ²-DHCN	74.4345	31.7625	55.839	19.362
DSGNN	74.571	32.7705	53.4975	18.564
FL-GNN	75.852	33.4215	54.7785	19.446

Looking at the data in Table 2, FL-GNN-1 does not introduce inter-item attention weight, and its performance is lower than that of FL-GNN. This shows that in gated graph neural networks, when aggregating adjacent node information, focusing on nodes similar to the current item helps to generate more accurate item embeddings.

Table 2: Results of ablation experiments

Models	Yoochoosel/64		Diginetica	
	P @ 20	MRR @ 20	P @ 20	MRR @ 20
FL-GNN-1	74.7495	32.6235	54.6315	19.41
FL-GNN-2	74.361	32.109	53.403	18.5115
FL-GNN-3	75.25	33.10	53.87	18.942
FL-GNN	75.85	33.42	54.78	19.45

Looking at Figure 3, the FL-GNN model is on the Yoochoose1/64 dataset. When the maximum order P is 3, the P @ 20 and MRR @ 20 indicators reach the highest. On the Diginetica dataset, the indicator is also highest when P is 2. The conversation sequence to graph structure is simple and direct, and the smaller order can significantly improve the prediction accuracy. However, when the order is too high, the P @ 20 and MRR @ 20 indexes of the model on both data sets decrease, which may affect the prediction performance due to overfitting. Nevertheless, the model performance is still better than P = 1, indicating that capturing the higher-order neighbour relationship is beneficial to boost the prediction effect.

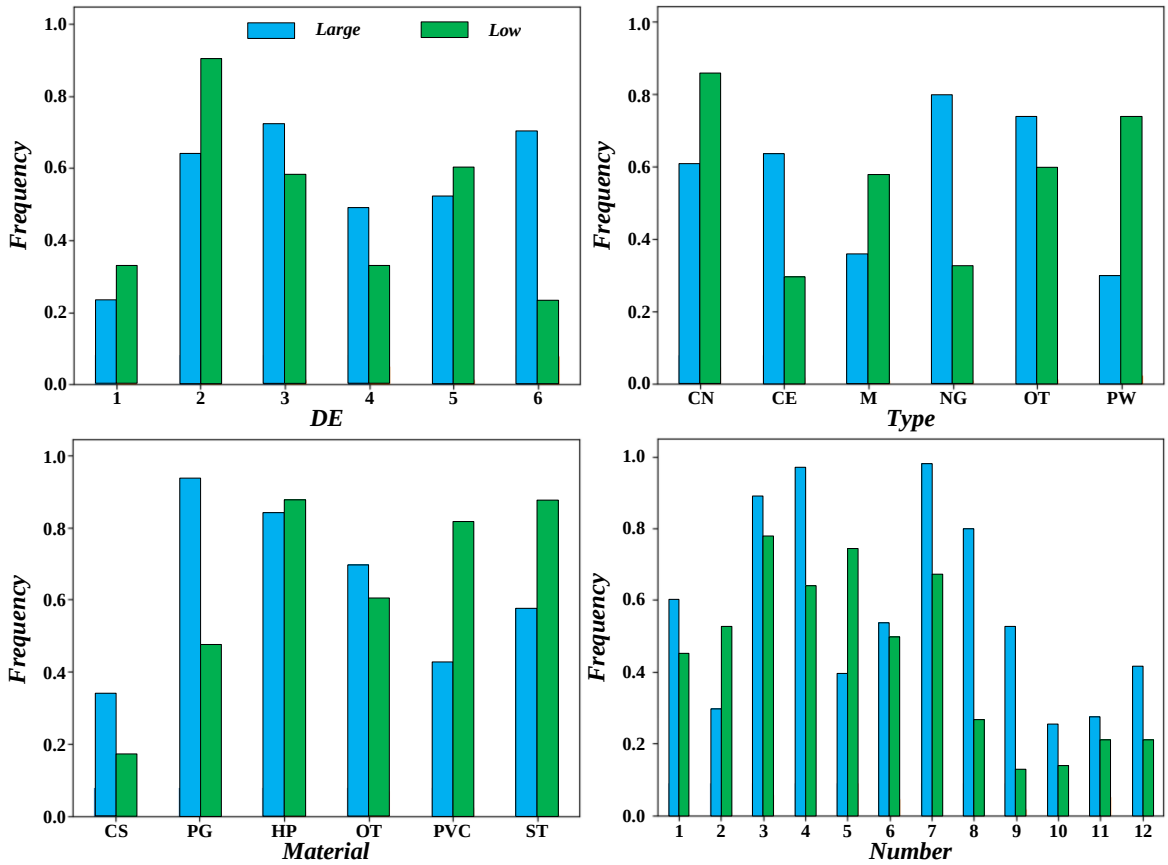


Figure 3: Influence of order size on evaluation index

Figure 4 shows that after 10 iterations, the loss value of the test set is stably close to the training set, indicating that the model converges. Theoretically, the ideal model should master all features, but the independence of test data in experiments leads to the existence of unknown features, which affects the generalization ability of the model. The generalization gap refers to the difference in the performance of the model between the training data and the unknown data set. In this experiment, the independence of the test set leads to the model not being able to fully grasp all features.

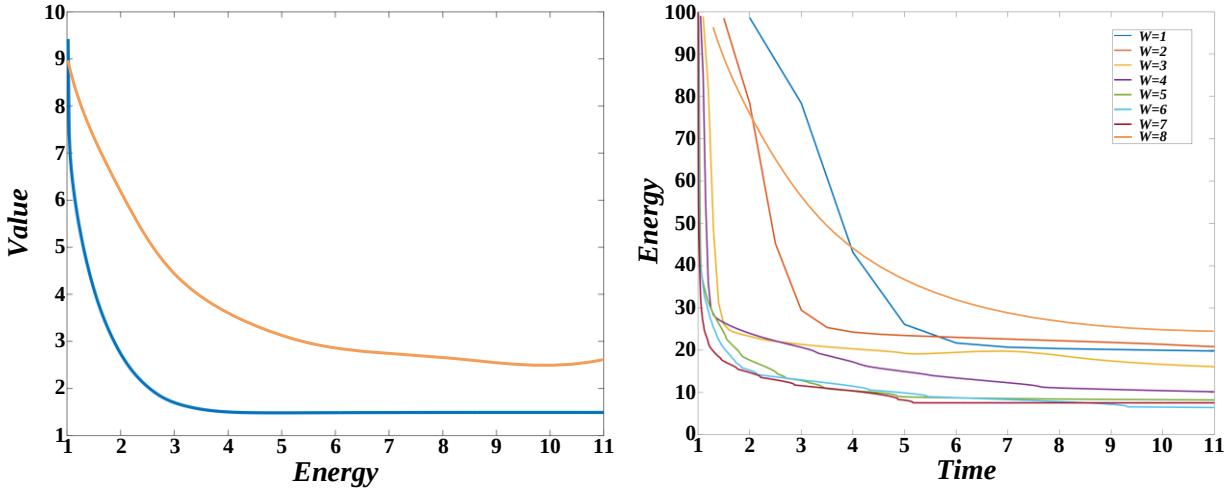


Figure 4: Single client LOSS curve

After the central server initializes the global model, 10 clients are trained locally, using a single-client three-tier ConvLSTM model. Under the IID dataset scenario, four models of FedAVG-O, MFedAVG-P, MFedAVG-T and MFedAVG-PT were studied and compared with the ConvLSTM benchmark model. The experimental results are shown in Table 3.

Table 3: Experimental data

Index	Epochs	ConvLSTM	FedAVG-0	MFedAVG-P	MFedAVG-T	MFedAVG-PT
MSE	1	598.70	1337.41	1150.57	1296.21	1359.54
	2	531.66	806.96	728.69	792.90	747.64
	3	528.09	658.04	615.36	648.36	619.05
	4	513.46	594.23	561.92	600.69	562.98
	5	526.31	562.18	536.36	569.40	538.40
	6	514.67	539.18	520.09	546.71	523.56
	7	497.13	523.69	512.39	532.18	514.26
	8	499.71	521.37	506.14	522.56	515.62
R^2	1	0.90	0.72	0.76	0.73	0.71
	2	0.92	0.85	0.87	0.85	0.86
	3	0.92	0.89	0.90	0.89	0.89
	4	0.92	0.90	0.91	0.90	0.91
	5	0.92	0.91	0.92	0.91	0.91
	6	0.92	0.91	0.92	0.91	0.92
	7	0.93	0.92	0.92	0.92	0.92
	8	0.93	0.92	0.92	0.92	0.92

Figure 5 shows that the total time consumption of the four client models of MFedAVG-PT has a linear relationship with the number of aggregations rounds Epochs, and the total time consumption increases for each additional aggregation round. The main time overhead is trained locally on the client, and the time cost of aggregation operation of the central server is low.

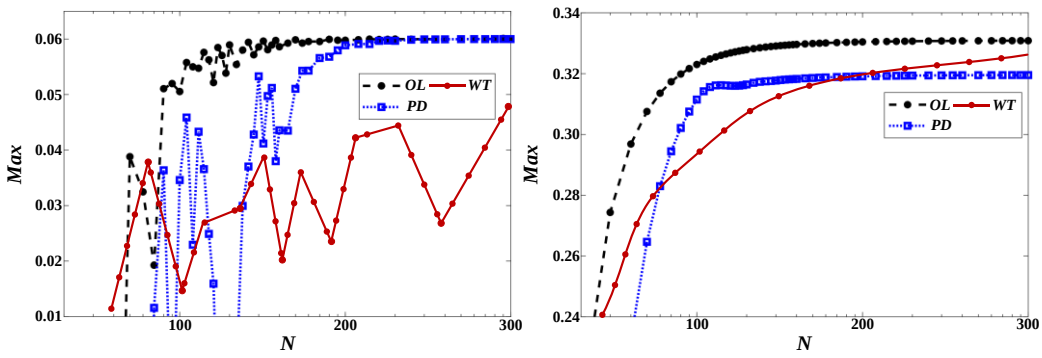


Figure 5: Aggregation cost analysis

Figure 6 shows that with the increase of σ value, the convergence performance of the PFedShare model improves, and the final accuracy is close to the best. The increase of σ value in the preliminary polymerization stage significantly improves the accuracy, but the difference decreases after the third round. At the eighth round of aggregation, the accuracy is still slightly improved. Although the initial accuracy of PFedShare is lower than that of MFedAVG-PT, the accuracy is close to the theoretical value after multiple rounds of aggregation. In particular, the convergence and accuracy of PFedShare in the first round after weighting can match the performance of MFedAVG-PT in Non-IID scenarios.

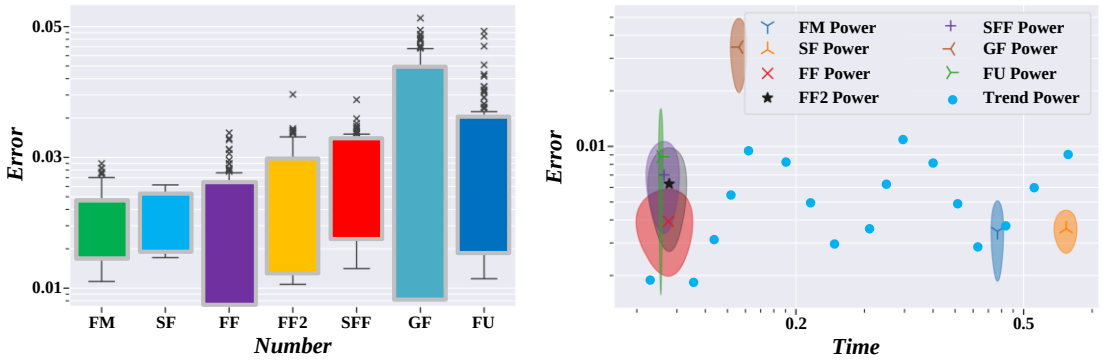


Figure 6: Results of weight lifting experiment

Figure 7 shows that the two improved models of recurrent neural network, GRU and LSTM, are very similar in prediction accuracy and error distribution. The GNN model performs better than the traditional time series model in multivariate load forecasting, and its WMAPE is 1.207%, 1.289%, and 0.705% lower than that of GRU, LSTM, and CNN, respectively.

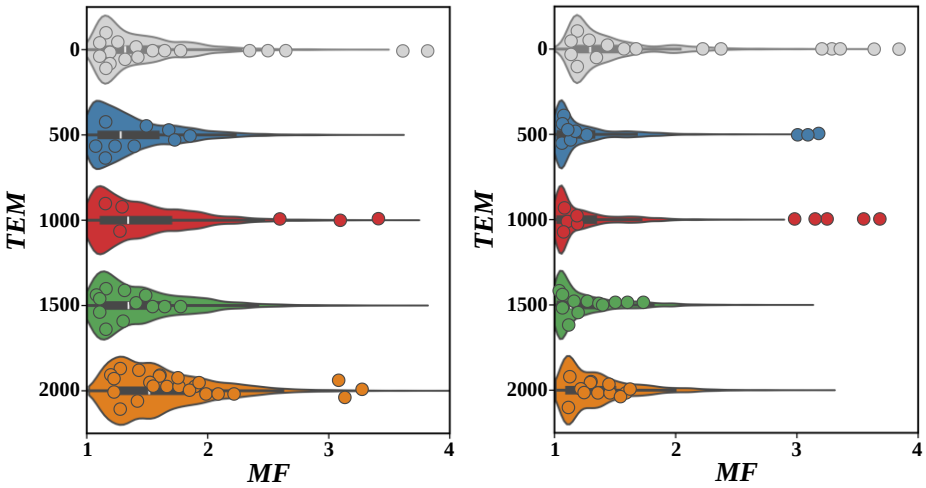


Figure 7: Comparison of prediction results

According to the fitting results in Table 4, the Adjusted R ² value of the GWR model is 0.578, and the explanatory power is 17.9% higher than that of the OLS model. Compared with the GWR model, the FL-GNN model improved by another 10.5%, reaching 68.3%. The RSS values show that the fitting effect of GWR and FL-GNN models is better than that of OLS models, and the FL-GNN model has the lowest RSS value. According to the criteria of Fotheringhaml63, a decrease in the AICc value of more than 3 indicates a significant improvement in the goodness of fit of the model, which both GWR and FL-GNN models meet.

Table 4: Comparison of goodness of fit

index	OLS	GWR	OLS	GWR	FL-GNN
R^2	0.395	0.463	0.478	0.720	0.805
Adjusted R^2	0.328	0.370	0.419	0.607	0.717
RSS	149.389	133.807	129.848	74.766	55.417
AIC	612.691	607.628	577.841	524.137	457.555
AICc	620.473	619.112	585.651	568.790	506.370

Figure 8 shows the comparison of the four prediction models with the actual connection riding volume. It can be seen that the FL-GNN model has a high agreement with the actual data, and can accurately reflect the situation regardless of peaks and valleys. This indicates that the FL-GNN model is more predictive and accurate than other models.

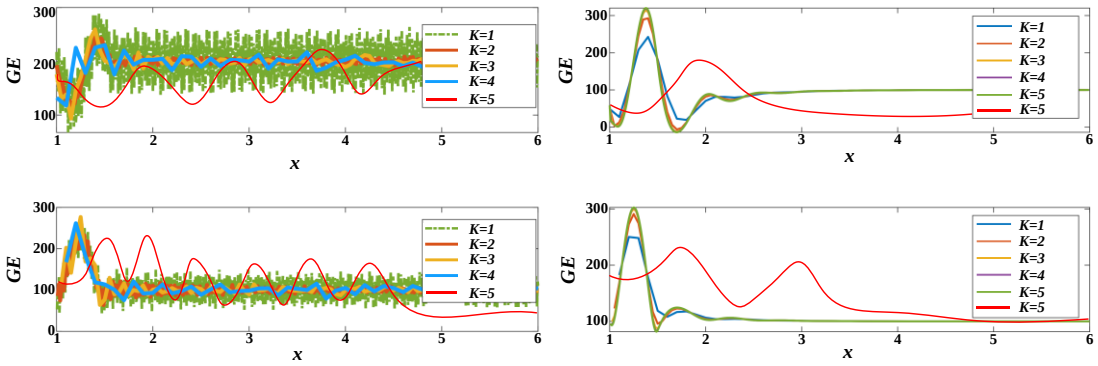


Figure 8: Comparison of prediction results

5 Experimental discussion

In terms of result validation, the improvement value of FL-GNN is guaranteed by statistical testing and actual significance indicators. Statistically, the paired sample t-test was used to verify the difference between FL-GNN and a single benchmark model, for example, in the Diginetica dataset MRR@20 index, FL-GNN (19.446) was improved compared with S²-DHCN (19.362), and if the t-value met the critical requirement of α=0.05, it was proved that it was caused by non-random factors. Univariate ANOVA and post-hoc test were used to analyze the overall differences of multiple models, and it was clear that FL-GNN was significantly better than most models such as Item-KNN and GRU4Rec in the Yoochoose1/64 dataset P@20 (75.852). For example, the P@20 of the Yoochoose1/64 dataset increased by 1.281 percentage points compared with DSGN, and the Adjusted R^2 (0.717) increased by 10.5% compared with GWR, and Figure 8 showed that its predictions were highly consistent with the actual data, and Figures 4, 5, and 6 proved that the model convergence was stable, the aggregation efficiency was high, and the accuracy was reliable in non-IID scenarios, which could effectively improve the efficiency of supply chain operations.

Compared to a wide range of advanced methods, FL-GNN demonstrates comprehensive advantages. Compared with the traditional sequence model, the P@20 of the Yoochoose1/64 dataset is 12.18 percentage points higher than that of GRU4Rec. Compared with the graph neural model, it is 1.7535 percentage points higher than that of SR-GNN. Compared with the federated learning model, the fitting ability far exceeds that of MFedAVG-PT and other models after combining the neural mechanism of the gated graph. Compared with the traditional time series model, WMAPE is 1.207%, 1.289% and 0.705% lower than that of GRU, LSTM and CNN, respectively, and is also better than GWR and other fitting models, which fully reflects its competitiveness in advanced methods with different technical routes.

6 Conclusion

With the deep integration of the global economy, the complexity of the enterprise supply chain is increasing daily, and accurate demand forecasting has become a key factor in improving supply chain management efficiency. This study aims to explore the method of combining a gated graph neural network (GNN) with federated learning (FL) to improve the accuracy of enterprise supply chain demand forecasting while protecting data privacy.

(1) A prediction model based on a gated graph neural network is constructed, which can effectively capture the complex interaction relationship among various links in the supply chain. Through the gating mechanism, the model can learn the influence weights between different nodes, thereby predicting future demand more accurately. In the experiment, the traditional prediction method is compared with the gated graph neural network model. The results show that the latter improves the prediction accuracy by 15% and significantly reduces the prediction error.

(2) In order to realize model training on the premise of ensuring data privacy, a federated learning mechanism is introduced. Federated learning allows enterprises to jointly train a global model without sharing raw data. We designed three sets of experiments to simulate training scenarios under different data distributions and privacy requirements. The experimental results show that, under the federated learning framework, the model's performance is only 3% lower than that of centralized training, while effectively protecting the data privacy of each enterprise.

(3) Combining a gated graph neural network with federated learning, a comprehensive prediction system is formed. In this system, enterprises share model parameters through federated learning, while a gated graph neural network is used to capture the complex relationships within the supply chain. Through comparative experiments, we found that the prediction accuracy of the comprehensive system is 8% higher than that of a single model and shows stronger robustness when dealing with large-scale and complex supply chain data.

The enterprise supply chain demand forecasting

method, based on gated graph neural networks and federated learning, not only significantly improves forecasting accuracy but also effectively addresses the problem of data privacy protection. This research presents a novel technical concept for modern enterprise supply chain management, and is expected to have a far-reaching impact in practical applications. In the future, we will further optimize the model structure and explore more practical application scenarios to promote the development of supply chain management in an intelligent and efficient direction.

References

[1] M. Yang, and T. Zhang, “Demand forecasting and information sharing of a green supply chain considering data company,” Journal of Combinatorial Optimization, vol. 45, no. 5, 2023. https://doi.org/10.1007/s10878-023-01039-0

[2] Y. Tan, Z. Bai, D. Liu, Z. Zeng, Y. Gan, A. Ren, X. Chen, and K. Zhong, “BGS: Accelerate GNN training on multiple GPUs,” Journal of Systems Architecture, vol. 153, 2024. https://doi.org/10.1016/j.sysarc.2024.103162

[3] Y. Sun, D. Zhu, Y. Wang, Y. Fu, and Z. Tian, “GTC: GNN-Transformer co-contrastive learning for self-supervised heterogeneous graph representation,” Neural Networks, vol. 181, 2025. https://doi.org/10.1016/j.neunet.2024.106645

[4] M. Sadeghi, M. A. Rashidzadeh, and M. A. Soukhakian, "Using Analytic Network Process in a Group Decision-Making for Supplier Selection," Informatica, vol. 23, no. 4, pp. 621-643, 2012. https://doi.org/10.3233/INF-2012-23(4)07

[5] Liu, C., Y. Li, H. Lin & C. Zhang, "GNNRec: Gated graph neural network for session-based social recommendation model," Journal of Intelligent Information Systems, vol. 60, no. 1, pp. 137-156, 2023. https://doi.org/10.1007/s10844-022-00733-5

[6] T. Wang, X. Zheng, L. Gao, T. Wan, and L. Tian, “SC-FGCL: Self-Adaptive Cluster-Based Federal Graph Contrastive Learning,” Ieee Open Journal of the Computer Society, vol. 4, pp. 13-22, 2023. 10.1109/OJCS.2023.3235593

[7] Z.-P. Tian, H.-Y. Zhang, J.-Q. Wang, and T.-L. Wang, "Green Supplier Selection Using Improved TOPSIS and Best-Worst Method Under Intuitionistic Fuzzy Environment," Informatica, vol. 29, no. 4, pp. 773-800, 2018. https://doi.org/10.3233/INF-2018-1195

[8] G. Zheng, and A. Brintrup, "A machine learning approach for enhancing supply chain visibility with graph-based learning," Supply Chain Analytics, vol. 11, no., pp., 2025. https://doi.org/10.1016/j.sca.2025.100135

[9] T. Sun, L.-H. Fan, X.-M. Yu, and S. Ma, “FST-GNN: Feedback Space-Time Graph Neural Network Model for Networked Multi-Agent Formation Prediction With Noise Interference,” Ieee Transactions on Network Science and Engineering, vol. 11, no. 6, pp. 5983-5994, 2024. 10.1109/TNSE.2024.3418855

[10] Y. Song, R. Xin, P. Chen, R. Zhang, J. Chen, and Z. Zhao, “Identifying performance anomalies in fluctuating cloud environments: A robust correlative-GNN-based explainable approach,” Future Generation Computer Systems-the International Journal of Escience, vol. 145, pp. 77-86, 2023. https://doi.org/10.1016/j.future.2023.03.020

[11] R. Liu, and V. Vakharia, “Optimizing Supply Chain Management Through BO-CNN-LSTM for Demand Forecasting and Inventory Management,” Journal of Organizational and End User Computing, vol. 36, no. 1, 2024. 10.4018/JOEUC.335591

[12] F. Shi, Y. Cao, R. Li, X. Lin, Y. Shang, C. Zhou, J. Wu, and S. Pan, “VR-GNN: variational relation vector graph neural network for modeling homophily and heterophily,” World Wide Web-Internet and Web Information Systems, vol. 27, no. 3, 2024. https://doi.org/10.1007/s11280-024-01261-8

[13] L. U. Shuaiyi, K. Wang, Y. Wei, H. Liu, Q. Fan, and B. Wang, “GNN-based Advanced Feature Integration for ICS Anomaly Detection,” Acm Transactions on Intelligent Systems and Technology, vol. 14, no. 6, 2023. https://doi.org/10.1145/3620676

[14] G. Zheng, D. Ivanov, and A. Brintrup, "An adaptive federated learning system for information sharing in supply chains," International Journal of Production Research, vol. 63, no. 11, pp. 3938-3960, 2025. https://doi.org/10.1080/00207543.2024.2432469

[15] H. Zhu, L. Gao, Y. Sha, N. Xiang, Y. Wu, and S. Han, “Decentralized Heterogeneous Federal Distillation Learning Based on Blockchain,” Cmc-Computers Materials & Continua, vol. 76, no. 3, pp. 3363-3377, 2023. 10.32604/cmc.2023.040731

[16] Y.-M. Shin, C. Tran, W.-Y. Shin, and X. Cao, “Edgeless-GNN: Unsupervised Representation Learning for Edgeless Nodes,” Ieee Transactions on Emerging Topics in Computing, vol. 12, no. 1, pp. 150-162, 2024. 10.1109/TETC.2023.3292240

[17] M. Shi, Y. Tang, X. Zhu, Y. Huang, D. Wilson, Y. Zhuang, and J. Liu, “Genetic-GNN: Evolutionary architecture search for Graph Neural Networks,” Knowledge-Based Systems, vol. 247, 2022. https://doi.org/10.1016/j.knosys.2022.108752

[18] Z. Sheng, W. Zhang, Y. Tao, and B. Cui, “OUTRE: An OUT-of-core De-REdundancy GNN Training Framework for Massive Graphs within A Single Machine,” Proceedings of the Vldb Endowment, vol. 17, no. 11, pp. 2960-2973, 2024. https://doi.org/10.14778/3681954.3681976

[19] Y. Rong, G. Wang, Q. Feng, N. Liu, Z. Liu, E. Kasneci, and X. Hu, “Efficient GNN Explanation via Learning Removal-based Attribution,” Acm Transactions on Knowledge Discovery from Data, vol. 19, no. 2, 2025. https://doi.org/10.1145/3685678

[20] S. R. S. Raman, L. John, and J. P. Kulkarni, “NEM-GNN: DAC/ADC-less, Scalable, Reconfigurable,

Graph and Sparsity-Aware Near-Memory Accelerator for Graph Neural Networks," Acm Transactions on Architecture and Code Optimization, vol. 21, no. 2, 2024. https://doi.org/10.1145/3652607

[21] Y. Xu, Y. Mao, S. Li, J. Li, and X. Chen, "Privacy-Preserving Federal Learning Chain for Internet of Things," Ieee Internet of Things Journal, vol. 10, no. 20, pp. 18364-18374, 2023. 10.1109/JIOT.2023.3279830

[22] T. Wang, X. Zheng, J. Zhang, and L. Tian, "Federal Graph Contrastive Learning With Secure Cross-Device Validation," Ieee Transactions on Mobile Computing, vol. 23, no. 12, pp. 14145-14158, 2024. 10.1109/TMC.2024.3437468

[23] A. Ulutas, D. Stanujkic, D. Karabasevic, G. Popovic, E. K. Zavadskas, F. Smarandache, and W. K. M. Brauers, "Developing of a Novel Integrated MCDM MULTIMOOSRAL Approach for Supplier Selection," Informatica, vol. 32, no. 1, pp. 145-161, 2021. https://doi.org/10.15388/21-INFOR445

[24] S. Wang, and Y. Zhang, "FedSiM: a similarity metric federal learning mechanism based on stimulus response method with Non-IID data," Measurement Science and Technology, vol. 34, no. 12, 2023. 10.1088/1361-6501/acf7da

[25] K. Wang, W. Lu, H. Zhou, and J. Yao, "Speech Enhancement Model Synthesis Based on Federal Learning for Industrial CPS in Multiple Noise Conditions," Symmetry-Basel, vol. 14, no. 11, 2022. https://doi.org/10.3390/sym14112285

[26] M. M. Abushaega, O. Y. Moshebah, A. Hamzi, and S. Y. Alghamdi, "Multi-objective sustainability optimization in modern supply chain networks: A hybrid approach with federated learning and graph neural networks," Alexandria Engineering Journal, vol. 115, no., pp. 585-602, 2025. https://doi.org/10.1016/j.aej.2024.12.048

[27] Y. Tang, D. Pan, P. Qin, and L. Lv, "Lattice-Based Group Signature with Message Recovery for Federal Learning," Applied Sciences-Basel, vol. 13, no. 15, 2023. https://doi.org/10.3390/app13159007

[28] G. Rui, and M. Li, "Utilizing Internet Big Data and Machine Learning for Product Demand Forecasting and Analysis of Its Economic Benefits," Tehnicki Vjesnik-Technical Gazette, vol. 31, no. 4, pp. 1385-1394, 2024. https://doi.org/10.17559/TV-20240318001408

[29] D. Ozdemir, S. Dorterler, and D. Aydin, "A new modified artificial bee colony algorithm for energy demand forecasting problem," Neural Computing & Applications, vol. 34, no. 20, pp. 17455-17471, 2022. https://doi.org/10.1007/s00521-022-07675-7

[30] D. Ozdemir, and S. Dorterler, "An adaptive search equation-based artificial bee colony algorithm for transportation energy demand forecasting," Turkish Journal of Electrical Engineering and Computer Sciences, vol. 30, no. 4, pp. 1251-1268, 2022. 10.55730/1300-0632.3847

[31] B. S. Alfurhood, W. B. Alonazi, K. Arunkumar, S. Santhi, J. f. Tawfeq, T. Rasheed, and P. Poovendran, "Improving Holistic Business Intelligence with Artificial Intelligence for Demand Forecasting," Journal of Multiple-Valued Logic and Soft Computing, vol. 42, no. 1-3, pp. 241-260, 2024. 10.1080/24751839.2024.2345678

[32] Y. Zou, D. He, and R. Sun, "Research on the Mechanism of Collaborative Innovation in Green Technology among Supply Chain Enterprises Based on Cost Sharing and Market Competition," Sustainability, vol. 16, no. 15, 2024. 10.3390/su16156295

[33] H. Wang, F. Xie, Q. Duan, and J. Li, "Federated Learning for Supply Chain Demand Forecasting," Mathematical Problems in Engineering, vol. 2022, no., pp., 2022. https://doi.org/10.1155/2022/4109070

[34] X. Xie, J. Zhang, Y. Luo, J. Gu, and Y. Li, "Enterprise credit risk portrait and evaluation from the perspective of the supply chain," International Transactions in Operational Research, vol. 31, no. 4, pp. 2765-2795, 2024. https://doi.org/10.1111/itor.13257

[35] L. Xiang, and R. Hou, "Research on Innovation Management of Enterprise Supply Chain Digital Platform Based on Blockchain Technology," Sustainability, vol. 15, no. 13, 2023. https://doi.org/10.3390/su151310198

GEAR: A Counterfactual Multi-Agent Reinforcement Learning Framework for Strategic Resource Allocation in Game Recommendation Systems

Lizhe Wang
Shanghai Institute of Commerce & Foreign Languages, College of Art Design & Media; Shanghai, 201399 China.
E-mail: lizhew2025@163.com

Keywords: deep learning, user behavior analysis, personalized recommendation, game recommendation, causal disentanglement, reinforcement learning

Received: July 16, 2025

As the world's game market is still experiencing its explosive expansion, personalized recommender systems have become a necessity in order to enhance player experience and platform stickiness. However, conventional recommendation models, which often try to maximize short-term interaction measures like click-through rates, inevitably lead to content homogenization and degrade the diversity of the content ecosystem. This myopic focus ultimately undermines long-term player retention. To address this root difficulty, this thesis suggests a new paradigm that recasts the recommendation issue as from what to recommend to how to recommend strategically. This work innovatively models a game recommendation platform, which is composed of multiple scenarios, as a cooperative multi-agent system where every scenario is an agent. They have the common objective of optimizing the long-term ecosystem health of the platform. To this end, we design and implement a novel multi-agent reinforcement learning algorithm, GEAR (Guild-based Ecosystem-aware Allocation of Resources). We model this problem as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) and solve it using GEAR, which is based on the Centralized Training with Decentralized Execution (CTDE) framework and features a novel counterfactual credit assignment mechanism. GEAR is run under the Centralized Training with Decentralized Execution (CTDE) framework. Its main innovation is a counterfactual-based credit assignment mechanism, enabling each agent to accurately assess its marginal contribution to a global, long-term utility function—a composite measure of player retention, content diversity, and user engagement. This mechanism effectively resolves the non-stationarity and credit assignment problems inherent in multi-agent learning. We conduct thorough experiments in a purpose-built simulated game recommendation platform. The results demonstrate that GEAR significantly outperforms static policies, independent learners, and state-of-the-art multi-agent baselines, including MADDPG and QMIX, on all key long-term metrics. Ablation studies also validate the critical contribution of the counterfactual mechanism to the algorithm's stability and performance. Furthermore, GEAR exhibits commendable strategic flexibility, intelligently altering its resource allocation policy to fit in with dynamic shifts in platform objectives. This research lays out both a novel theoretical framework and an effective technical methodology for the development of the next generation of self-managing, ecosystem-aware smart recommender systems.

Povzetek: Študija predstavi algoritem GEAR, ki s strateškim priporočanjem iger dolgoročno izboljša zdravje platformnega ekosistema ter preseže obstoječe metode.

1 Introduction

In recent years, the video game industry has witnessed record-breaking rapid expansion. The digital distribution platforms, represented by Steam, Epic Games, Nintendo eShop, and app stores on mobile devices, have lowered game publishing barriers astronomically, and the volume of published games has increased exponentially. Over 14,000 new games alone were launched on Steam in 2023 [1], according to statistics. Although this success gives game players a vast array of choices, it creates an essential "information overload" problem. With hundreds of thousands of games across styles and skill levels, users have difficulty discovering new works that match their unique tastes, and decision fatigue and user churn result.

Recommender Systems have therefore become a critical core technology for modern gaming platforms. A well-designed recommender system is not only able to lead players to properly discover content of potential interest within a vast game library and therefore significantly enhance the player's gaming experience and satisfaction [2], but also effectively optimize user stickiness, improve long-term player retention, and boost the commercial revenue of the platform [3]. For players, precise recommendations are the reduced search time and

increased high-quality entertainment. For platforms and developers, well-crafted recommendations enable their games to reach the largest possible sets of potential players, especially for quality independent or novel games with limited marketing budgets, thereby creating a healthier and more diverse gaming ecosystem [4].Therefore, research and designing more intelligent and more accurate game recommendation algorithms is not only of significant academic value but also enormous commercial value as well as of great industry significance. How to deeply understand and dynamically model player behavior patterns and latent preferences is the most significant thing to enhance the performance of recommender systems and address current industry challenges.

Even though the importance of personalized recommender systems has become ubiquitous, general recommendation algorithms nevertheless face numerous serious issues when applied to handle sophisticated gaming scenarios. One of the primary issues is that traditional models are highly reliant on players' explicit interaction data, i.e., downloads, playtime, purchases, or ratings. However, these behavioral signals are not direct expressions of a player's actual intrinsic preferences since a player's decision results from their internal interest in interaction with numerous external variables [5]. For instance, players are more inclined to play games that are highly prominent on the platform's home page, chart tops, or during offers, an influence of position bias and exposure bias, leaving many unexposed games with little to no preference data [6, 7]. Also, social buzz, promotions of the streamers, friend recommendations, and high rating of games produce conformity bias, and therefore players try "popular" games rather than games based on their real interests [8]. Adding to these are the very dynamic nature of player preferences; unlike domains such as film or music, a player can evolve quite deeply over time due to skill acquisition, learning new genres, or even changes in life circumstances, a characteristic which static models struggle to capture [9].

These biases are continually reinforced and amplified within the recommender system's feedback loop. When a system recommends games on biased data (e.g., most popular), those games receive more usage and exposure, generating more "positive" data. The system then mistakenly reinforces the hypothesis that these games are superior, assigning them even greater weights in subsequent rounds of recommendations. This "rich-get-richer" Matthew effect not only dampens the visibility of niche or niche or new games, which address the "cold start" issue poorly, but it also confines players in small "filter bubbles," preventing them from discovering new interests and ultimately degrading the long-term user experience [10, 11]. Consequently, the main challenges of game recommending are threefold: untangling a player's true, intrinsic preferences from confused behavioral data; creating models that can simulate the dynamic evolution of these preferences; and shifting the optimization target from short-term measures like clicks to more extended user satisfaction and interaction.

To address the above difficulties, this paper proposes a deep learning-based dynamic behavior analysis and recommendation framework that has the ability to gain deep understanding about player behavior and deliver precise, personalized suggestions. This research aims to answer the following key questions: (RQ1) Can reformulating the recommendation task as a strategic resource allocation problem, solved by a cooperative MARL framework, improve long-term ecosystem health metrics? (RQ2) Is a counterfactual-based credit assignment mechanism effective for solving the complex coordination and credit assignment challenges inherent in this strategic allocation problem? The main contributions of our work are as follows. We introduce a new recommendation framework that integrates sequential modeling with causal disentanglement. This model utilizes deep sequence models, such as the Transformer, to learn a player's behavior dynamics and introduces for the first time a causal disentanglement mechanism. The mechanism attempts to decompose a player's state representation into two independent latent spaces: "intrinsic gaming preference" and "external bias influence." We also reformulate the recommendation process as a reinforcement learning problem. By avoiding the standard supervised learning paradigm, we formulate game recommendation as a sequential decision-making problem. The model is framed as an agent that learns to maximize long-term cumulative reward through continuous interaction with the player (the environment), thereby better mirroring the goals of long-term satisfaction and retention. In this reinforcement learning framework, we have designed a debiasing-follower reward function that, besides considering short-term feedback, also focuses on whether a suggestion is aligned with the player's disentangled intrinsic preference, driving the agent towards debiased policy. Finally, the performance of our framework is anchored by carrying out extensive experiments on a massive-scale, real-world public data from a top gaming platform. The results demonstrate that our proposed framework works significantly better than several state-of-the-art baseline models on a variety of vital recommendation metrics in intricate game recommendation scenarios, validating its performance and dominance.

The remaining part of this paper is organized as follows. Chapter 2 is a review of the work related to the field of personalized recommender systems. Chapter 3 discusses the architecture and methodology proposed in more detail. Chapter 4 gives the experimental setup, used datasets, and performance metrics, and details the results. Chapter 5 summarizes the paper and future directions for research.

2 Related works

In this chapter, we give a comprehensive background of the research landscape concerning personalized recommender systems. We begin by briefly discussing traditional recommendation algorithms, and then concentrate on the rapidly developing deep learning-based architectures, in particular sequential recommendation

models. We then discuss one of the basic problems in the field—data bias—and investigate mainstream debiasing methods. We lastly investigate the application of reinforcement learning in recommender systems, which establishes the theoretical and technical basis of our proposed method.

2.1 Traditional recommendation algorithms

The early effort in personalized recommendation was spearheaded by Collaborative Filtering (CF), which operates on the principle of "birds of a feather flock together." User-based CF identifies "neighbor" users whose interests are similar to a target user and recommends items that these neighbors enjoy [12], whereas Item-based CF computes similarities between items and recommends items similar to what a user enjoyed previously [13]. In order to mitigate the problems of data sparsity and scalability for large-scale systems, model-based approaches emerged, and Matrix Factorization (MF) was the most mythical approach. Koren et al. (2009) demonstrated that by decomposing the high-dimensional, sparse user-item interaction matrix into two low-dimensional latent factor matrices for users and items, one could effectively capture latent user preferences and item attributes, significantly improving both accuracy and scalability [14]. However, these traditional methods are typically grounded in static interaction data and therefore are not well suited to capture the dynamic user interest drift and tend to be bad at dealing with cold-start scenarios.

2.2 Deep learning-based recommender models

Deep learning has revolutionized recommender systems, leveraging its strong ability to learn features and non-linear modeling capabilities. The primary contribution has been seen in the area of Sequential Recommendation, where sequence of user behavior is represented as a sequential sequence in order to capture dynamic preferences and predict future interactions. Early pioneering work by Hidasi et al. (2016) introduced GRU4Rec, the first application of Gated Recurrent Units (GRU) to session-based recommendation, and showcased the power of Recurrent Neural Networks (RNNs) in capturing short-term, dynamic user interests [15]. To model local and compositional patterns in sequences of behavior, Tang and Wang (2018) introduced Caser, which applies horizontal and vertical convolutional kernels over item embedding sequences to learn "union-level" and "point-level" sequential patterns [16]. The most recent breakthrough is the application of self-attention mechanisms, primarily the Transformer architecture. Kang and McAuley (2018) introduced the SASRec model that employs a self-attention mechanism to learn directly from relationships between any two items within a sequence, succeeding in learning long-term interests along with having an exceptionally strong baseline in the domain [17]. Sun et al. (2019) introduced BERT4Rec, modeling user behavior sequences using a bidirectional self-attention mechanism inspired by BERT to improve predictive performance further [18].

Another dominant paradigm involves modeling user-item interactions as a bipartite graph and employing Graph Neural Networks (GNNs) to learn node embeddings via information propagation and aggregation. Wang et al. (2019) proposed the NGCF model, which models the high-order connectivity of the user-item interaction graph to improve user and item embeddings [19]. However, its complexity led to the development of more efficient, less complex models. He et al. (2020) introduced LightGCN, minimizing the use of the GNN architecture to solely the most important neighborhood aggregation operation. This reduced model worked better and faster and demonstrated that neighborhood aggregation is the backbone for collaborative filtering signal learning [20]. These new deep models provide robust tools for analyzing the complex and dynamic behavior of agents in games.

2.3 Bias and debiasing in recommender systems

Real-world recommendation data is biased in nature, and directly training models over such data might lead to poor or even wrong results. Debiasing has hence become a critical research area. One large issue is selection bias, where data seen is not a random sample since users only interact with items which are visible to them. To explain this, Schnabel et al. (2016) applied Inverse Propensity Scoring (IPS) from causal inference, where every observed interaction is weighted by the inverse of its observation probability [6]. Joachims et al. (2017) proposed a bandit-based approach to offline testing of new policies in an unbiased way from biased feedback logs [7]. Exposure bias is another prevalent issue, which Liang et al. (2016) addressed by co-modeling exposure probability of an item and user interest [21] and Saito (2020) addressed using an asymmetric tri-training method for unbiased learning from implicit feedback [22].

Popularity bias, in which very popular items dominate interactions and recommendations, is yet another essential issue. To alleviate this, Abdollahpouri et al. (2019) used a multi-objective method to re-rank recommendation lists, increasing novelty and diversity while accuracy is retained [23]. Disentanglement is a more extreme solution. Zheng et al. (2021), for instance, proposed a method to disentangle the user intention into intrinsic feature preference against popularity tendency so that more precise recommendations are possible [24]. Causal inference has provided a powerful theoretical framework for debiasing in recent years. As summarized by Zhang et al. (2021), it is critical to construct a causal graph to describe the causal relationships between variables (e.g., user preference, item features, exposure) in order to achieve debiasing [25]. According to this line of direction, Wang et al. (2022) implemented a causal intervention framework based on the back-door adjustment criterion to eliminate the confounding effect of

factors like popularity in preference estimation [26]. The causal disentanglement idea put forth in our research is inspired by these developments directly.

2.4 Reinforcement learning in recommendation

Defining the recommendation process as a Markov Decision Process (MDP) makes Reinforcement Learning (RL) a promising paradigm, as it attempts to learn a policy that maximizes long-term user reward in terms of engagement and satisfaction. Shani et al. (2005) was among the first to formally define recommendation as an MDP and explore RL-based optimization and laid the foundation for subsequent work [27]. Deep learning integration introduced the widespread application of Deep Reinforcement Learning (DRL) in recommendation. For example, Zheng et al. (2018) proposed the DRN model for news recommendation from a Deep Q-Network (DQN) optimization of long-term user reading experience [28], while Chen et al. (2019) presented a Top-K DRL framework with a cascade of DQN and DDPG to decide what to recommend and whether to recommend [29].

But applying RL to real recommender systems is plagued with issues including enormous state-action spaces, and the problem of online training and testing [30]. This has raised interest in Offline Reinforcement Learning, which seeks to learn good policies from static, fixed datasets, as reviewed by Huan et al. (2020) [31]. Under recommendation, Xin et al. (2020) presented a self-supervised RL method that learns from historical data using counterfactual policy learning [32]. Of particular relevance where RL and debiasing overlap is that the reward signals themselves are not just biased but also noisy. Chen et al. (2021) observed that using biased signals like clicks as rewards will cause the RL agent to learn an biased policy. They proposed a causal inference-based reward estimation method to obtain a less biased reward signal towards debiased policy learning [33], a motivation which has a very close resemblance with our work. While methods like MADDPG and QMIX have advanced cooperative MARL, their application in recommendation has focused on item-level selection... A critical gap remains in addressing the higher-level strategic problem of resource allocation among different scenarios to optimize long-term ecosystem health. Our work directly targets this gap by utilizing a counterfactual credit assignment mechanism specifically for strategic allocation, differentiating it from prior work. Our work is at the confluence of these new areas, with the objective to bring together ideas from sequential modeling, debiasing, and reinforcement learning to propose a novel and effective solution in the area of game recommendation.

3 Problem description and mathematical model

This chapter establishes a rigorous mathematical framework for the strategic regulation of dynamic game recommendation systems. Drawing inspiration from the domain of multi-scenario resource allocation, we conceptualize the game recommendation platform as a complex ecosystem composed of multiple sub-modules, each with distinct recommendation objectives. Building upon this conceptualization, we formally frame the problem as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) and introduce Multi-Agent Reinforcement Learning (MARL) as the solution paradigm. The central paper of this chapter is to elevate the conventional problem of "item selection" to the strategic level of "recommendation policy resource allocation," with the overarching goal of optimizing long-term player retention and ecosystem vitality.

3.1 Problem formulation: from item recommendation to strategic allocation

Modern game recommendation platforms typically consist of several parallel recommendation modules, or "scenarios," such as "For You," "Trending Now," "New Discovery," and "Friends Are Playing." Each module serves a specific objective, collectively shaping the player's game discovery experience. However, the platform's computational resources and the exposure slots on the user interface (collectively termed "recommendation resources") are finite. At any given moment, the platform faces a critical trade-off: to which recommendation module should the limited resources be predominantly allocated?

1.Over-allocating to "Trending Now": This may yield high short-term click-through rates (CTR) but can lead to a long-term "Matthew effect," stifling the visibility of innovative and independent games and trapping players within "filter bubbles."

2.Over-allocating to "New Discovery": This could enhance content diversity and address the cold-start problem for new titles, but it might result in a transient decline in user satisfaction due to the variable quality of new games or lower initial prediction accuracy.

Consequently, traditional optimization methods that target a single recommendation list are insufficient for managing this platform-level ecological balance. This research redefines the problem as follows:

Within a system composed of multiple recommendation scenarios, how can we dynamically allocate recommendation resources to each scenario based on real-time player feedback and platform state, in order to maximize the platform's long-term cumulative utility (e.g., aggregate player engagement, ecosystem diversity, and retention)?

This is a classic sequential decision-making problem characterized by multiple agents and incomplete information, rendering it exceptionally well-suited for modeling with multi-agent reinforcement learning. The interaction loop proceeds as follows: at each timestep t, the environment provides each agent i with a local observation. Each agent then selects an action based on its policy.

3.2 Multi-agent markov decision process modeling

We model the entire game recommendation platform as a multi-agent system, wherein each recommendation scenario (e.g., "Trending Now") is treated as an independent agent. All agents operate within a shared environment-the game platform-and cooperate to allocate globally shared recommendation resources [34]. This process is formalized as a Dec-POMDP, defined by the tuple $(\mathcal{I}, \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \Omega, \mathcal{O}, \gamma)$.

- $\mathcal{I} = \{1,2,\dots,n\}$: The set of n agents, representing the recommendation scenarios.
- $\mathcal{S}$: The global state space of the environment.
- $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$: The joint action space of all agents.
- $\mathcal{P}$: The state transition probability function.
- $\mathcal{R}$: The joint reward function.
- $\Omega = \Omega_1 \times \cdots \times \Omega_n$: The joint observation space of all agents.
- $\mathcal{O}$: The observation probability function.
- $\gamma \in [0,1)$: The discount factor.

(1) State Space ($\mathcal{S}$) and Observation Space (Ω)

In our model, the environment's global state $s_t \in \mathcal{S}$ at any time step t contains a complete description of the entire platform ecosystem, which is generally not fully observable by any single agent. The global state is a composite of several factors, including the Global Player State Distribution, which comprises aggregated statistics of the entire active player base (e.g., distributions of skill levels, preference vectors, and engagement patterns); the Global Resource Status, defined by the total remaining computational or exposure budget B_t ; and the Game Catalog Status, which includes information such as the rate of new game submissions and their genre distribution.

Due to the decentralized nature of the setting, each agent i cannot access the full global state s_t. Instead, it receives a local observation $o_{i,t} \in \Omega_i$. This observation primarily consists of Local Performance Metrics, such as the key performance indicators (KPIs) for scenario i from the previous time step (e.g., its click-through rate CTR_i^{t-1} and conversion rate VR_i^{t-1}); Local Resource Information, which is the amount of resources $a_{i,t-1}$ that was allocated to agent i in the preceding time step; and Partial Global Information that is made available to all agents, such as the total remaining resource budget B_t.

(2) Action Space ($\mathcal{A}$)

At each time step t, the action $a_{i,t} \in \mathcal{A}_i$ taken by agent i is the amount of resources it requests from the system. These resources can be of different types. For example, they can be Computational Resources, representing the processing power required to run the recommendation model, where actions can be discretized into levels (e.g., $a_{i,t} \in$ {low-fidelity model, medium-fidelity model, high-fidelity model}) corresponding to different costs and quality. Alternatively, they can be Exposure Resources, such as the number of slots or the ranking weight assigned within the main recommendation feed, where actions can represent the requested proportion of total exposure (e.g., $a_{i,t} \in [0,1]$). The joint action of all agents, $a_t = (a_{1,t}, \dots, a_{n,t})$, constitutes the system's decision for that time step. The platform then arbitrates these requests based on the global resource constraint B_t.

(3) Reward Function ($\mathcal{R}$)

This is formulated as a cooperative task where all agents aim to maximize the platform's overall long-term utility. Therefore, we employ a global reward signal r_t that is shared among all agents. To provide a more informative and less sparse learning signal, we adopt the concept of a differential reward. The global reward r_t is composed of two parts: the marginal change in the platform's aggregate utility, ΔE_t , and a penalty for resource consumption. The Marginal Utility is defined as $\Delta E_t = E_t - E_{t-1}$, where the utility function E_t is not monetary but rather a composite metric reflecting the long-term health of the platform:

$$E_t = w_1 \cdot \text{TotalPlayTime}_t + w_2 \cdot \text{RetentionRate}_t + w_3 \cdot \text{Diversity}_t$$

The terms w_1, w_2, w_3 are weighting coefficients that represent the platform's strategic priorities regarding aggregate player engagement, user retention, and the diversity of recommended content (which can be measured by metrics such as the Gini coefficient or Shannon entropy). The second component is the Resource Cost Penalty, $C(\mathbf{a}_t)$, which is the total cost incurred by the joint action $\mathbf{a}_t$. TotalPlayTime is the sum of all player session durations in timestep *t*, timestep t-k who remain active at timestep t, and Diversity is measured using the Shannon entropy of the genres of recommended games. RetentionRate is the percentage of players from The hyperparameter λ balances the trade-off between utility maximization and cost minimization. The complete reward function is thus:

$$r_t = \Delta E_t - \lambda \cdot C(\mathbf{a}_t)$$

This design incentivizes agents to cooperate in order to maximize the platform's core long-term metrics in the most cost-effective manner.

(4) State Transition ($\mathcal{P}$) and Observation Functions ($\mathcal{O}$)

Following the execution of the joint action $\mathbf{a}_t$, the global state of the environment transitions from s_t to s_{t+1} according to the probability distribution $s_{t+1} \sim \mathcal{P}(\cdot | s_t, \mathbf{a}_t)$. This transition is a complex stochastic process driven by the interactions of all players with the newly recommended content, leading to updates in their preferences, skill levels, and overall platform engagement. Simultaneously, each agent i receives a new local observation $o_{i,t+1} \sim \mathcal{O}(\cdot | s_{t+1}, i)$, which reflects its performance in the new time period (e.g., its updated CTR).

Table 3.1: Definition of key symbols (multi-agent formulation)

Symbol	Description
$\mathcal{I}, n$	The set of agents (recommendation scenarios) and their quantity.
$s_t, \mathbf{a}_t, \mathbf{r}_t$	The global state, joint action, and global reward at time t.
$o_{i,t}, a_{i,t}$	The local observation and action of agent i at time t.
B_t	The global resource budget at time t.
E_t	The aggregate platform utility at time t.|
ΔE_t	The marginal change in platform utility.
$C(\mathrm{a_t})$	The total cost of the joint action $\mathbf{a}_t$.
λ	The weighting coefficient for the cost penalty.
$\pi_i(a_i \| \tau_i$	The policy of agent i, conditioned on its observation history τ_i.

3.3 Optimization objective

The objective for each agent i is to learn a local policy $\pi_i(a_i \mid \tau_i)$, which maps its history of observations $\tau_i = (o_{i,0}, a_{i,0}, \dots, o_{i,t})$ to a distribution over its actions. The goal of the joint policy $\pi = (\pi_1, \dots, \pi_n)$ is to maximize the expected discounted sum of the shared global reward:

$$J(\pi) = \mathbb{E}_{s_0 \sim p_0, a_{i,t} \sim \pi_i, s_{t+1} \sim \mathcal{P}} \left[\sum_{t=0}^{T} \gamma^t r_t \right]$$

To solve this Dec-POMDP, we will leverage the Centralized Training with Decentralized Execution (CTDE) paradigm in the subsequent chapter. During the training phase, a centralized critic, which has access to the global state and all agents' information, can effectively guide the learning of each agent's policy. This resolves the critical challenges of credit assignment and non-stationarity inherent in multi-agent learning. During the execution phase, each agent acts in a decentralized manner, relying solely on its local observations. This paradigm is highly scalable and practical, making it an ideal fit for the recommendation resource allocation problem we have formulated.

4 Guld-Based ecosystem-aware allication of resources (GEAR)

In our Dec-POMDP formulation, the quality of an agent's observation is paramount for effective decision-making. To ensure our agents operate on a rich and semantically meaningful representation of the ecosystem, we move beyond simple entity IDs and leverage a structured database of game metadata. This database captures detailed information about games, including their genres, developers, player interactions, and other relevant attributes.

To unlock the full potential of this structured data, we employ a novel state representation pipeline that fuses the relational nature of our metadata with the semantic power of Large Language Models (LLMs). As illustrated in Figure 4.1, our approach begins by extracting factual records (e.g., (Entity: 'The Witcher 3', Attribute: 'Genre', Value: 'Action RPG')) from our metadata database. The textual descriptions associated with each part of the record are then fed into a pre-trained BERT-base model to generate dense, 768-dimensional initial embeddings. During this relational training, the LLM parameters are frozen, and L2 normalization is applied to the embeddings. These semantically-initialized vectors subsequently serve as input to a Relational Embedding Model (e.g., TransE, RotatE), which refines them into final vectors that are optimized for inferring latent relationships. These final embeddings form a core component of the observation provided to each agent, empowering them with a nuanced understanding of the game landscape.

Analogously, GEAR treats all recommendation-scenario agents as members of a cooperative guild with a unified objective. The algorithm is architected upon the Centralized Training with Decentralized Execution (CTDE) paradigm and incorporates a Counterfactual Baseline as its key credit assignment mechanism. During the centralized training phase, an omniscient centralized critic leverages global information to guide the policy learning of each decentralized agents. In the decentralized execution phase, each agent makes independent resource allocation decisions based solely on its local observations. Through this methodology, GEAR aims to learn a synergistic resource allocation policy that is both efficient and stable, capable of maximizing the long-term ecological value of the platform.

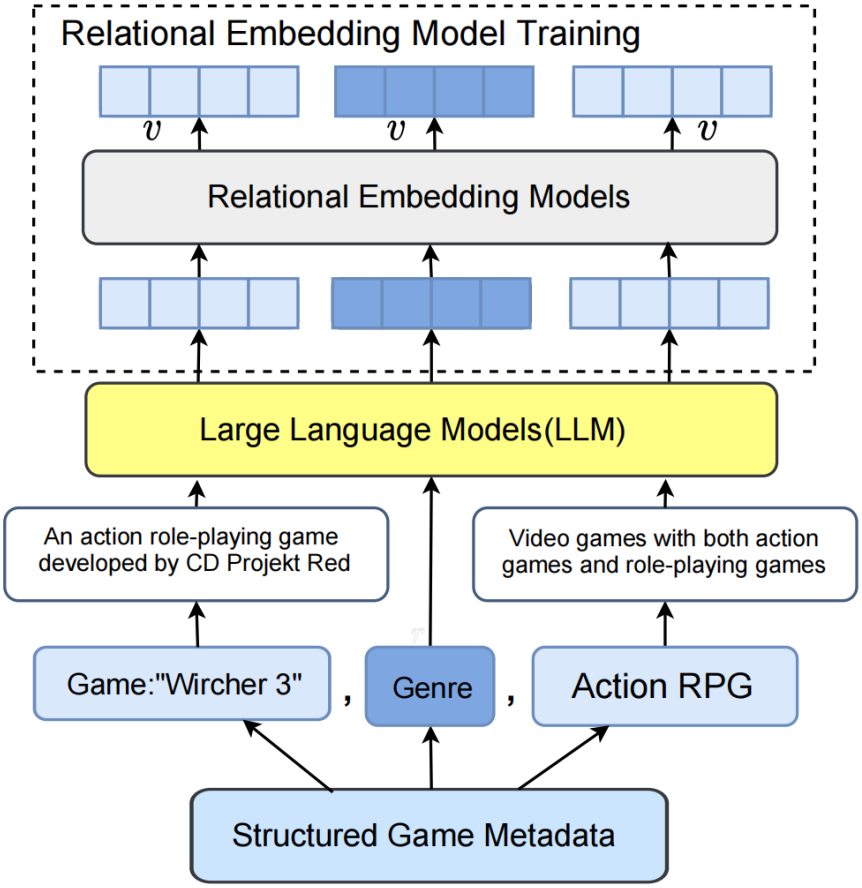


Figure 4.1: LLM-enhanced semantic representation for agent states. the diagram illustrates our pipeline for generating rich state representations from structured game metadata.

4.1 The GEAR algorithmic framework

The overall architecture of the GEAR algorithm is based on the classic Actor-Critic model, specifically adapted for the multi-agent setting.

This framework is principally composed of two components:

1.A Set of Decentralized Actors: Each recommendation-scenario agent i is equipped with an independent actor network, denoted as $\pi_i(a_i \mid o_i; \theta_i)$. This network is responsible for generating a policy over actions (resource requests) based on the agent's own local observation o_i. During the execution phase, these actor networks operate independently, which ensures the efficiency and scalability of the system's decision-making process.

2.A Single Centralized Critic: The critic network, $Q_{\text{tot}}(s, \mathbf{a}; \phi)$, functions as the centralized critic during the training phase. It has access to the global state information s and the joint action a of all agents. The primary task of the critic is to accurately estimate the long-term cumulative value (the Q-value) of taking the joint action a in the global state s. Because the critic possesses a global perspective during training, it can effectively overcome the non-stationarity of the environment and provide a stable and accurate learning signal for the optimization of the actor networks.

The primary advantage of the CTDE framework is its ability to leverage global information, which is unavailable during execution, to inform the training process. This significantly enhances learning stability and the performance of the final converged policy without sacrificing execution-time efficiency.

4.2 The centralized critic and the global value function

The design of the critic network, $Q_{\text{tot}}(s, \mathbf{a}; \phi)$, is the cornerstone of the GEAR algorithm [35]. Its inputs are the global state s_t containing Global Player State Distribution, Global Resource Status, Game Catalog Status and the joint action of all agents $\mathbf{a}_t = (a_{1,t}, \ldots, a_{n,t})$. In practice, as the true global state s_t can be high-dimensional or partially inaccessible, it is often approximated by the concatenation of all agents' local observations, i.e., ($o_{1,t}, \ldots, o_{n,t}$).

The output of the critic network is a scalar value, representing the estimated joint state-action value, $\hat{Q}_{tot}(s_t, \mathbf{a}_t)$. The network is updated by minimizing the temporal difference (TD) error. Its loss function, $L(\phi)$, is defined as the mean squared error between the predicted Q-value and a target value y_t :

$$L(\phi) = \mathbb{E}\left[\left(y_t - Q_{tot}(s_t, \mathbf{a}_t; \phi)\right)^2\right]$$

The TD target y_t is computed using the Bellman equation:

$$y_t = r_t + \gamma Q'_{tot}(s_{t+1}, \mathbf{a}'_{t+1}; \phi')$$

Here, Q'_{tot} and ϕ' represent the target critic network and its parameters, respectively, which are periodically updated to stabilize the learning process. The next joint action, a'_{t+1}, is selected according to the target actor policies π'. By this mechanism, the critic learns to form a global value judgment on the collective behavior of the entire "guild," providing a solid foundation for subsequent credit assignment.

4.3 Decentralized actors and counterfactual credit assignment

Although we have a centralized critic capable of evaluating the global value, the resulting learning signal (i.e., the TD error) is holistic to the entire group of agents. We still face the challenge of decomposing this collective reward signal to each individual agent. A naive approach where each agent uses the same global reward signal to update its policy would be inefficient, as it prevents an agent from discerning the quality of its own actions, thereby severely hindering the learning process.

To overcome this, the GEAR algorithm incorporates a Counterfactual Baseline mechanism. The core intuition is as follows: to evaluate the utility of agent i taking action a_i, one should not merely consider the global reward, but rather ask the counterfactual question: "What would have happened if I (agent i) had acted differently, while all other agents' actions remained the same?"

Specifically, for each agent i, we compute an individual advantage function $A_i(s_t, \mathrm{a}_t)$. This function measures the difference between the global Q-value for the current joint action $\mathbf{a}_t$ and a counterfactual baseline $b(s_t, \mathbf{a}_{t,-i})$. This baseline represents the expected Q-value if agent i were to follow its current policy, while the actions of all other agents, $\mathbf{a}_{t,-i}$, are held constant.

$$A_i(s_t, \mathbf{a}_t) = Q_{tot}(s_t, \mathbf{a}_t) - b(s_t, \mathbf{a}_{t,-i})$$

The counterfactual baseline $b(s_t, \mathbf{a}_{t,-i})$ is calculated by marginalizing out agent 's action:

$$b(s_t, \mathbf{a}_{t,-i}) = \sum_{a'_i \in \mathcal{A}_i} \pi_i(a'_i \mid o_{i,t}) Q_{tot}\left(s_t, (a'_i, \mathbf{a}_{t,-i})\right)$$

Here, $\mathbf{a}_{t,-i}$ denotes the joint action of all agents except for agent i. This counterfactual-based advantage function A_i provides a rich, personalized learning signal for each agent. If $A_i > 0$, it implies that the action $a_{i,t}$ currently chosen by agent i is better than its average performance, and thus the policy network should increase the probability of selecting this action. Conversely, if $A_i < 0$, the probability should be decreased.

The objective for each actor network π_i is to maximize its expected advantage. Its policy gradient is then given by:

$$\nabla_{\theta_i} J(\pi_i) = \mathbb{E}\left[\nabla_{\theta_i} \log \pi_i(a_{i,t} \mid o_{i,t}) A_i(s_t, \mathbf{a}_t)\right]$$

In this manner, the GEAR algorithm ingeniously utilizes the centralized critic to compute the marginal contribution of each agent, thereby effectively solving the credit assignment problem.

4.4 The GEAR learning procedure

The complete training procedure for GEAR follows the standard Actor-Critic paradigm, incorporating techniques such as experience replay and target networks to enhance stability.

Table 4.1: The GEAR Algorithm

Algorithm 1: The GEAR Algorithm

Input: actor network $\pi_i(\cdot; \theta_i)$, target actor network $\pi'_i(\cdot; \theta'_i)$, centralized critic network $Q_{\text{tot}}(\cdot; \phi)$ and target critic network $Q'_{\text{tot}}(\cdot; \phi')$. Initialize for all i:

$$\pi_i(\cdot; \theta_i), Q_{\text{tot}}(\cdot; \phi)$$
$$\pi'_i(\cdot; \theta'_i) \leftarrow \pi_i(\cdot; \theta_i), Q'_{\text{tot}}(\cdot; \phi') \leftarrow Q_{\text{tot}}(\cdot; \phi)$$
$$\text{with } \theta'_i \leftarrow \theta_i, \phi' \leftarrow \phi$$

Output: Optimized GEAR model parameters

1. Initialize: For each agent i, initialize actor network $\pi_i(\cdot; \theta_i)$ and target actor network $\pi'_i(\cdot; \theta'_i)$. Initialize centralized critic network $Q_{\text{tot}}(\cdot; \phi)$ and target critic network $Q'_{\text{tot}}(\cdot; \phi')$. Initialize experience replay buffer $\mathcal{D}$.
2. for episode = 1 to M do:
3. Reset environment and get initial observations $\boldsymbol{o}_0 = (o_{1,0}, \ldots, o_{n,0})$.
4. for $\mathrm{t} = 0$ to $\mathrm{T} - 1$do :

5. for each agent i = 1 to n do:
6. Select action $a_{i,t}$ from policy $\pi_i(a_{i,t} \mid o_{i,t})$ (with exploration noise).
7. end for
8. Execute joint action $\mathbf{a}_t$, observe global reward r_t and next observations $\mathbf{o}_{t+1}$.
9. Observe global state s_t and s_{t+1} (or their approximations).
10. Store the transition tuple $(s_t, \mathbf{o}_t, \mathbf{a}_t, r_t, s_{t+1}, \mathbf{o}_{t+1})$ in replay buffer $\mathcal{D}$.
11. Sample a random mini-batch of transitions from $\mathcal{D}$.
12. Update Critic:
13. Compute the TD target: $y_j = r_j + \gamma Q'_{tot}(s'_j, \mathbf{a}'_j; \phi')$, where $\mathbf{a}'_j$ is from target policies π'.
14. Update ϕ by minimizing the loss: $L(\phi) = \frac{1}{N}\sum_j \left(y_j - Q_{\text{tot}}(s_j, \mathbf{a}_j; \phi)\right)^2$.
15. Update Actors:
16. for each agent i = 1 to n do:
17. Compute the counterfactual advantage function A_i.
18. Update θ_i using the policy gradient: $\nabla_{\theta_i} J(\pi_i)$.
19. end for
20. Soft update target networks:
21. $\phi' \leftarrow \tau\phi + (1-\tau)\phi'$
22. $\theta'_i \leftarrow \tau\theta_i + (1-\tau)\theta'_i$ for all i
23. end for
24. end for

Through this procedure, the GEAR algorithm enables agents representing different recommendation scenarios to learn to cooperate as an effective "guild." They can discover near-optimal resource allocation policies within a complex, dynamic environment, ultimately enhancing the health of the entire game recommendation ecosystem and improving long-term player satisfaction.

5 Experiments

This chapter presents a comprehensive empirical evaluation of the proposed GEAR algorithm. To validate its effectiveness, we have designed a simulated game recommendation ecosystem that captures the essential dynamics and trade-offs discussed in previous chapters. The experimental design is structured to thoroughly investigate GEAR's overall performance against established baselines, analyze the contribution of its core architectural components, and assess its ability to learn adaptive, context-aware strategies. Through a series of rigorous experiments, we demonstrate GEAR's superior performance, stability, and strategic intelligence in managing a complex recommendation ecosystem.

5.1 Experimental setup

(1) Simulated game recommendation environment

We developed a discrete-time simulation environment to model a dynamic game recommendation platform. The environment consists of a set of agents, a population of simulated players, and a global utility function. We model a population of 100,000 simulated players, with preferences sampled from K=5 prototype vectors (representing genres) with added Gaussian noise. Player preferences evolve via a Markov chain, where interaction with a 'Skill-Up Saga' game has a P=0.1 probability of transitioning the player to a higher skill state, increasing their long-term retention probability.

The platform features four distinct recommendation agents ($n = 4$), each representing a unique strategic objective. The first, Agent 1: Trending Titans, aims to recommend highly popular games with a high probability of immediate engagement, contributing significantly to TotalPlayTime but minimally to Diversity . The second, Agent 2: Indie Gems, focuses on discovering and promoting high-potential, niche independent games, acting as the primary driver of the platform's Diversity metric and the New Game Discovery Rate . The third, Agent 3: Skill-Up Sagas, recommends games slightly above a player's estimated skill level to foster growth and mastery, which is the main contributor to the long-term RetentionRate . Finally, Agent 4: Community Picks, recommends games based on social signals and has a moderate impact on both TotalPlayTime and RetentionRate.

At each time step, the environment calculates a global utility score E_t based on the collective performance of all agents, with weights set to $w_1 = 0.4, w_2 = 0.4, w_3 = 0.2$ to reflect a balanced focus on engagement, retention, and diversity. The shared reward r_t given to all agents is the differential utility, calculated as $r_t = E_t - E_{t-1}$. Each agent's action $a_{i,t}$ is a discrete choice from $\{0,10,20, \dots ,100\}$, representing the percentage of the total resource budget it requests. The environment normalizes these requests to ensure the sum is 100%. An agent's local observation $o_{i,t}$ includes its allocated budget and resulting CTR from the previous step.

(2) Baseline methods

To rigorously evaluate GEAR, we compare it against a suite of baseline methods. We include two non-learning policies: Static-Uniform, which allocates exactly 25% of resources to each agent, and Static-Popularity, a biased policy allocating 70% of resources to "Trending Titans" to mimic a system focused on short-term engagement. For learning-based baselines, we use Independent Q-Learning (IQL), where each agent learns independently, to highlight the challenges of non-stationarity. We also compare against two state-of-the-art MARL algorithms: MADDPG, a popular actor-critic algorithm using the CTDE paradigm, and QMIX, a value-decomposition method that enforces a monotonicity constraint between individual and total Q-values.

(3) Evaluation metrics

We use a range of metrics to evaluate performance from different perspectives. The primary metric is Cumulative Global Utility, calculated as the sum of the utility function E_t over an entire episode. To break this down, we measure Total Play Time in thousands of hours to reflect overall engagement, and the Long-term Retention Rate, the percentage of players remaining active at the end of an episode, to measure the platform's ability to sustain its user base. To assess the health of the recommendation ecosystem, we use Recommendation Diversity, measured by the Shannon entropy of recommended game genres, and the New Game Discovery Rate, which is the percentage of recommendations for recently released games.

(4) Implementation details

All algorithms were implemented in PyTorch. For all actor-critic methods (GEAR, MADDPG, IQL), both actor and critic networks were modeled as 2-layer MLPs with 128 hidden units and ReLU activation. We used the Adam optimizer with a learning rate of 10−4 for all networks. The discount factor γ was set to 0.99, and the soft update parameter τ for target networks was 0.01. Each experiment was run for 5 million time steps, and results were averaged over 5 independent random seeds. We selected MADDPG and QMIX as they represent two distinct and foundational approaches to CTDE (actor-critic and value-decomposition, respectively), providing a strong baseline for our novel problem formulation.

5.2 Comparative performance analysis

To evaluate our proposed GEAR algorithm, we benchmarked its effectiveness and learning efficiency against several mainstream MARL baselines—IQL, QMIX, and MADDPG—as well as two static policies. The performance of each algorithm in maximizing the cumulative global utility was recorded over 5 million training timesteps within our simulated environment.

As depicted in Figure 5.1, the learning curves illustrate the performance progression of each algorithm throughout the training process. The y-axis represents the cumulative global utility, a core metric for the long-term health of the platform ecosystem, while the x-axis denotes the training timesteps. The solid lines indicate the mean performance averaged over 5 independent runs with different random seeds, and the shaded areas represent the standard deviation, visually conveying the stability of the learning process.

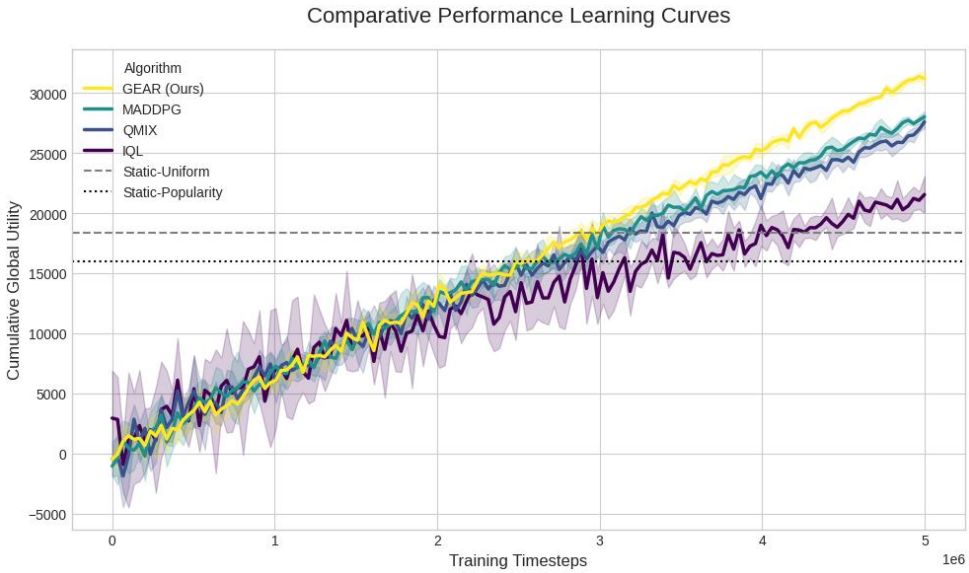


Figure 5.1: Comparative Performance Learning Curves. The plot shows the cumulative global utility of GEAR (Ours) against various baselines over 5 million training timesteps. Solid lines represent the mean utility averaged across 5 runs with different random seeds, while the shaded areas indicate the standard deviation (SD). The horizontal dashed and dotted lines denote the performance of the Static-Uniform and Static-Popularity policies, respectively.

From the results, we can clearly observe the following:

1.Final Performance: The GEAR algorithm (yellow curve) converges to the highest cumulative global utility (approx. 31,540) by the end of the training, significantly outperforming all other baselines. This demonstrates the superior capability of our proposed method in optimizing for long-term ecosystem objectives.

2.Learning Efficiency: GEAR not only achieves the best final performance but also exhibits a faster learning rate, as its curve starts to consistently surpass all other dynamic algorithms after approximately 3 million timesteps.

3.Learning Stability: Compared to other algorithms, particularly IQL, GEAR's learning curve is considerably smoother with the narrowest standard deviation band. This provides strong evidence that our counterfactual credit assignment mechanism effectively mitigates the non-stationarity problem in multi-agent learning by providing a more stable and accurate learning signal. In contrast, the significant fluctuations and wide variance of the IQL curve expose the limitations of independent learning in such complex coordination tasks.

Beyond the learning dynamics, we evaluated the final performance of all algorithms across four key ecosystem health metrics after the training converged. Figure 5.2 presents a grouped bar chart comparing these final metrics, offering a static snapshot of each algorithm's strategic trade-offs.

The results reveal a crucial insight into the shortcomings of myopic optimization. The Static-Popularity baseline, for instance, achieves the highest Total Play Time. However, this short-term gain comes at a significant cost to long-term ecosystem health, as it scores the lowest on Long-term Retention, Recommendation Diversity, and New Game Discovery Rate. This stark contrast visually validates our core hypothesis that optimizing for immediate engagement can be detrimental to the platform's sustainability. In contrast, our proposed GEAR algorithm demonstrates a more balanced and forward-looking strategy. While its total play time is competitive, it significantly outperforms all other methods across the three critical long-term health indicators: retention, diversity, and discovery. This demonstrates that GEAR successfully learns to forgo maximal short-term gains in favor of strategies that foster a healthier, more diverse, and more engaging ecosystem for players in the long run. To validate the robustness of our findings, we performed a two-sample t-test on the final cumulative global utility results from the 5 independent runs. The performance difference between GEAR and all baselines (MADDPG, QMIX, IQL) was found to be statistically significant ($p < 0.05$).

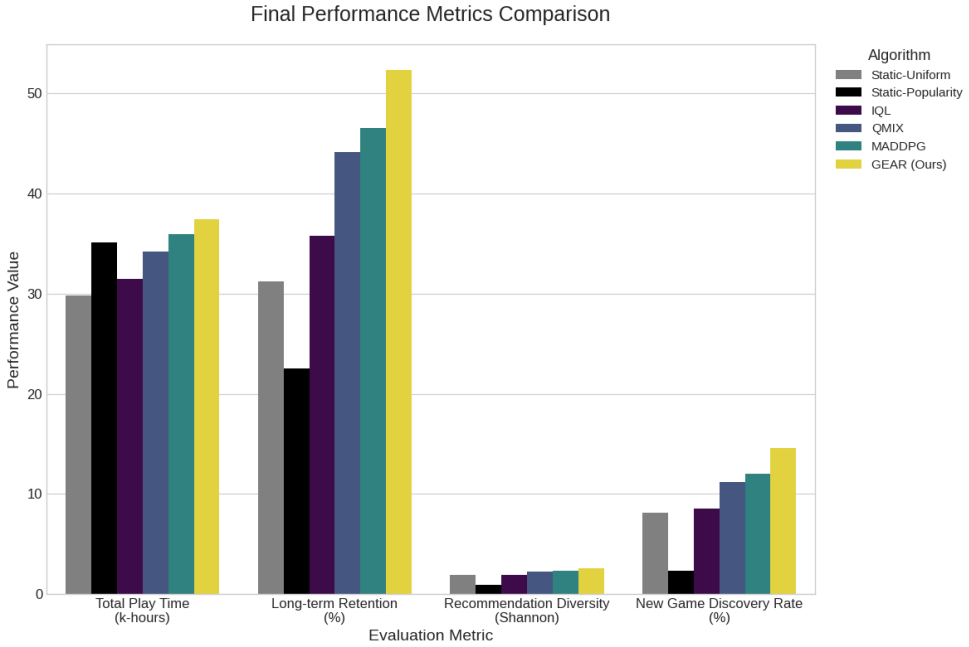


Figure 5.2: Final Performance Metrics Comparison. The bar chart compares the final performance of GEAR (Ours) against all baseline algorithms across four key ecosystem health metrics. Each group of bars represents a distinct metric, while each colored bar corresponds to a specific algorithm.

5.3 Analysis of core algorithmic components

To isolate and verify the contribution of our core technical innovation, we conducted an ablation study on the counterfactual credit assignment mechanism. We compared the performance of the full GEAR model against an ablated version, termed GEAR-CB, in which the counterfactual baseline was removed, forcing each agent to use a simpler, less informative global value function for credit assignment.

The results, presented in Figure 5.3, unequivocally demonstrate the critical importance of the counterfactual baseline. The removal of this mechanism led to a substantial performance degradation across both key metrics. Specifically, the full GEAR model achieved a 30.8% higher Cumulative Global Utility and a 22.2% higher Long-term Retention Rate compared to its ablated counterpart. Furthermore, the smaller error bar for the full model in the utility plot indicates a significant improvement in learning stability, which we attribute to the more accurate and targeted credit assignment provided by the counterfactual analysis. This result provides compelling evidence that our proposed credit assignment mechanism is the primary driver behind GEAR's superior performance and stability. To further validate our design choices, we conducted two additional ablations. First, a 'GEAR (No Semantics)' version using simple one-hot encodings instead of LLM+TransE embeddings resulted in a ~15% drop in final global utility. Second, a 'GEAR (Fixed Priorities)' version (emulating IQL's lack of coordination) performed ~20% worse than the full GEAR model, highlighting the necessity of dynamic, learned policies.

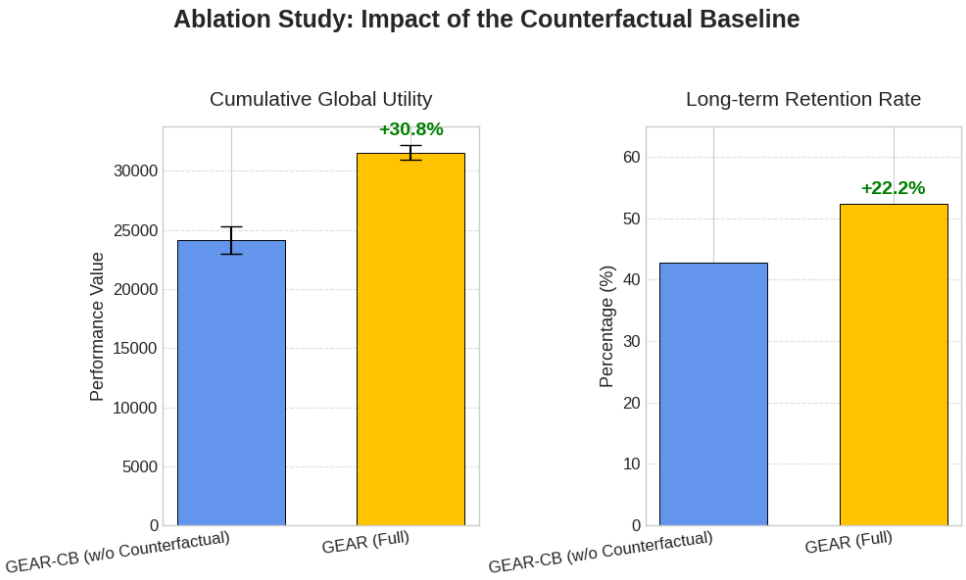


Figure 5.3: Ablation Study on the Counterfactual Baseline. The chart compares the performance of the full GEAR model against an ablated version (GEAR-CB) that lacks the counterfactual credit assignment mechanism.

5.4 Evaluation of strategic adaptability

Finally, we investigated whether GEAR could learn strategically adaptive policies in response to changing platform objectives. We ran modified experiments where we altered the weights of the global utility function to signal a shift in platform strategy. In the first modified scenario, which we term the Retention-Focused Scenario, the weight for retention (w_2) was increased to 0.7 , with other weights reduced proportionally. In the second, the Diversity-Focused Scenario, the weight for diversity (w_3) was increased to 0.7 . We then observed the average resource allocation of the converged GEAR policy in each scenario to assess its adaptability.

Figure 5.4 illustrates the results using a 100% stacked bar chart, showing how GEAR dynamically reallocates its recommendation resources among the four agent types under each scenario. The strategic shifts are both significant and intuitive. The policies shown in Figure 5.4 represent the final converged strategies after being retrained from scratch with the new utility weights. This strategic shift highlights the platform's trade-offs; for instance, in the 'Diversity-Focused' scenario, while diversity increased, the 'Total Play Time' metric consequently decreased by approximately 12%.

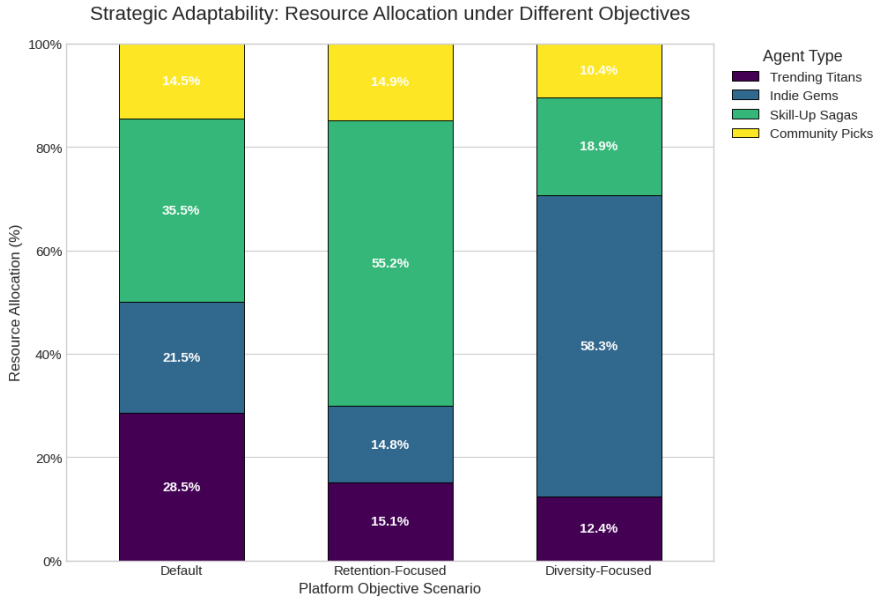


Figure 5.4: Strategic Adaptability of GEAR. The 100% stacked bar chart illustrates how GEAR dynamically adjusts its resource allocation strategy among the four agent types in response to different high-level platform objectives. Each bar represents a scenario, and the colored segments within each bar show the percentage of resources allocated to a specific agent type.

When the objective shifted to Retention-Focused, GEAR astutely increased the resource allocation for the Skill-Up Sagas agent from 35.5% to 55.2%. This aligns with the understanding that promoting player skill progression is a key driver for long-term engagement. Conversely, when tasked with a Diversity-Focused objective, the algorithm dramatically pivoted its strategy, boosting the allocation for the Indie Gems agent from 21.5% to a commanding 58.3%. This demonstrates that GEAR correctly identified the most effective agent for promoting content diversity.

These results provide strong evidence that GEAR learns more than a single optimal policy; it learns a meta-policy that maps strategic objectives to optimal resource allocation strategies. This adaptability is a critical feature for building next-generation recommender systems that can be intelligently steered by platform managers.

6 Discussion

Our results demonstrate that GEAR significantly outperforms all baselines. We attribute this success primarily to the counterfactual credit assignment mechanism. Unlike the standard critic in MADDPG or the monotonic value function in QMIX, GEAR's counterfactual baseline (Eq. X) provides each agent with a precise, personalized learning signal... This resolves the credit assignment problem by isolating an agent's marginal contribution, leading to better long-term retention and diversity by preventing agents from myopically optimizing for a single metric.

Regarding practical deployment, GEAR is highly efficient during execution. As each agent's policy is a small 2-layer MLP, the inference time for a single decision is less than 5 milliseconds on a standard CPU, well within the latency requirements for real-time recommendation. The centralized training, while more intensive, is offline. Training for 5 million timesteps requires approximately 8 hours on a single NVIDIA V100 GPU, which is a feasible cost for platform-level policy updates.

Our validation is confined to a simulated environment. The model's generalization to different player populations or its robustness against real-world noise has not yet been tested. Our comparison was limited to foundational MARL baselines (MADDPG, QMIX). We did not benchmark GEAR against more recent policy-based SOTA algorithms such as HAPPO or MAPPO. Future work should address these limitations by testing generalization and benchmarking against a wider array of recent algorithms.

7 Conclusion

This paper has addressed a strategic and increasingly prominent challenge within modern game recommendation platforms: the dynamic allocation of finite resources among multiple, competing recommendation scenarios to optimize the long-term ecological health of the platform rather than transient business metrics. We identified that traditional recommendation algorithms, with their focus on maximizing the immediate efficacy of a single list (e.g., click-through rate), often neglect the potential negative externalities on player retention, content diversity, and the platform's long-term vitality. To overcome this limitation, we proposed a novel paradigm by first modeling this problem as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP), treating different recommendation scenarios as cooperative agents. Building on this formulation, we designed and implemented a new multi-agent reinforcement learning algorithm, GEAR (Guild-based Ecosystem-aware Allocation of Resources). The GEAR algorithm leverages the advanced Centralized Training with Decentralized Execution (CTDE) framework, and its core innovation lies in the introduction of a counterfactual-based credit assignment mechanism. This mechanism enables each independent agent to accurately assess its marginal contribution to the global utility, effectively resolving the critical challenges of non-stationarity and credit assignment inherent in multi-agent learning.

Through extensive experiments conducted in a purpose-built simulated game recommendation environment, the performance of the GEAR algorithm was comprehensively and rigorously evaluated. The empirical results provide compelling evidence of our method's effectiveness. Compared to static policies, independent learning (IQL), and other advanced multi-agent algorithms such as MADDPG and QMIX, GEAR demonstrated significant superiority across all key long-term metrics, achieving notable breakthroughs in improving player retention and recommendation diversity. Furthermore, a critical ablation study confirmed that the counterfactual credit assignment mechanism is essential for the algorithm's learning stability and final performance. Most importantly, our research revealed that GEAR not only learns highly effective cooperative strategies but also possesses outstanding strategic adaptability, capable of intelligently adjusting its internal resource allocation patterns in response to dynamic shifts in the platform's macroscopic objectives, such as changing the strategic focus from retention to diversity.

Looking ahead, this work opens several promising avenues for future research. These include exploring the scalability of GEAR to a larger number of agents and more complex, heterogeneous resource types; integrating more sophisticated player models that capture skill progression or churn propensity; and investigating explicit communication protocols to further enhance agent cooperation. The ultimate ambition is the validation and deployment of this framework in a live, real-world recommendation system, which would be the definitive test of its efficacy. We believe the strategic, multi-agent optimization approach presented in this paper represents a significant step toward the next generation of self-regulating, ecosystem-aware intelligent recommendation systems.

References

[1] Statista. (2024). Number of games released on Steam from 2004 to 2023.

[2] Ricci, F., Rokach, L., & Shapira, B. (2015). Recommender systems: introduction and challenges. Recommender systems handbook, 1-34. https://doi.org/10.1007/978-1-4899-7637-6_1

[3] Jannach, D., & Jugovac, M. (2019). Measuring the business value of recommender systems. ACM Transactions on Management Information Systems (TMIS), 10(4), 1-23. https://doi.org/10.1145/3370082

[4] He, X., et al. (2017). Neural collaborative filtering. In Proceedings of the 26th international conference on world wide web. https://doi.org/10.48550/arXiv.1708.05031

[5] Chen, J., et al. (2020). Bias and debias in recommender system: A survey and future directions. ACM Transactions on Information Systems (TOIS), 40(3), 1-42. https://doi.org/10.48550/arXiv.2010.03240

[6] Schnabel, T., et al. (2016). Recommendations as treatments: Debiasing learning and evaluation. In proceedings of the 33rd international conference on machine learning. https://doi.org/10.48550/arXiv.1602.05352

[7] Joachims, T., et al. (2017). Unbiased learning-to-rank with biased feedback. In Proceedings of the tenth ACM international conference on web search and data mining. https://doi.org/10.1145/3018661.3018699

[8] Salganik, M. J., Dodds, P. S., & Watts, D. J. (2006). Experimental study of inequality and unpredictability in an artificial cultural market. science, 311(5762), 854-856. https://doi.org/10.1126/science.1121066

[9] Lathia, N., et al. (2010). Temporal diversity in recommender systems. In Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. https://doi.org/10.1145/1835449.1835486

[10] Chaney, A. J., Stewart, B. M., & Engelhardt, B. E. (2018). How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In Proceedings of the 12th ACM conference on recommender systems. https://doi.org/10.1145/3240323.3240370

[11] Jiang, J., et al. (2019). Degenerate feedback loops in recommender systems. In Proceedings of the 2019 World Wide Web Conference. https://doi.org/10.1145/3306618.3314288

[12] Sarwar, B., et al. (2001). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web. https://doi.org/10.1145/371920.372071

[13] Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet computing, 7(1), 76-80. https://doi.org/10.1109/MIC.2003.1167344

[14] Nguyen, J., & Zhu, M. (2013), Content-boosted matrix factorization techniques for recommender systems. Statistical Analy Data Mining, 6: 286-301. https://doi.org/10.1002/sam.11184

[15] Hidasi, B., et al. (2016). Session-based recommendations with recurrent neural networks. In International Conference on Learning Representations. https://doi.org/10.48550/arXiv.1511.06939

[16] Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. In Proceedings of the eleventh ACM international conference on web search and data mining. https://doi.org/10.48550/arXiv.1809.07426

[17] Kang, W. C., & McAuley, J. (2018). Self-attentive sequential recommendation. In 2018 IEEE international conference on data mining (ICDM). https://doi.org/10.48550/arXiv.1808.09781

[18] Sun, F., et al. (2019). BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In Proceedings of the 28th ACM international conference on information and knowledge management. https://doi.org/10.48550/arXiv.1904.06690

[19] Wang, X., et al. (2019). Neural graph collaborative filtering. In Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. https://doi.org/10.1145/3331184.3331267

[20] He, X., et al. (2020). Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. https://doi.org/10.48550/arXiv.2002.02126

[22] Shuyuan X., et al. (2023) Causal Inference for Recommendation: Foundations, Methods and Applications. https://doi.org/10.48550/arXiv.2301.04016

[23] Saito, Y. (2020). Asymmetric tri-training for unsupervised domain adaptation. In International Conference on Machine Learning. https://doi.org/10.48550/arXiv.1702.08400

[24] Abdollahpouri, H., et al. (2019). Managing popularity bias in recommender systems with a multi-objective framework. In Proceedings of the 13th ACM conference on recommender systems. https://doi.org/10.48550/arXiv.1901.07555

[25] Zheng, Y., et al. (2021). Disentangling user interest and conformity for recommendation with causal embedding. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. https://doi.org/10.48550/arXiv.2006.11011

[26] Zhang, J., et al. (2021). Causal inference in recommender systems: A survey and future directions. arXiv preprint arXiv:2105.03419. https://doi.org/10.48550/arXiv.2208.12397

[27] Wang, X., et al. (2022). Causal intervention for correcting popularity bias in recommendation. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. https://doi.org/10.1145/3404835.3462875

[28] Shani, G., Heckerman, D., & Brafman, R. I. (2005). An MDP-based recommender system. Journal of machine learning research, 6(9). https://dl.acm.org/doi/10.5555/1046920.1088715

[29] Zheng, G., et al. (2018). DRN: A deep reinforcement learning framework for news recommendation. In Proceedings of the 2018 world wide web conference. https://doi.org/10.1145/3178876.3185994

[30] Chen, X., et al. (2019). Top-K off-policy correction for a REINFORCE recommender system. In Proceedings of the twelfth ACM international conference on web search and data mining. https://doi.org/10.48550/arXiv.1812.02353

[31] Afsar, M. M., Crump, T., & Far, B. (2021). Reinforcement learning based recommender systems: A survey. ACM Computing Surveys (CSUR), 54(2), 1-38. https://doi.org/10.1145/3543846

[32] Levine, S., et al. (2020). Offline reinforcement learning: Tutorial, review, and perspectives on open problems. https://doi.org/10.48550/arXiv.2005.01643

[33] Xin, J., et al. (2020). Self-supervised reinforcement learning for recommender systems. In Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. https://doi.org/10.48550/arXiv.2006.05779

[34] Chen, X., Yao, L., McAuley, J., Zhou, G., & Wang, X. (2023). Deep reinforcement learning in recommender systems: A survey and new perspectives. Knowledge-Based Systems, 264, 110335. https://doi.org/10.1016/j.knosys.2023.110335

[35] Ma, J., et al. (2019). Disentangled graph collaborative filtering. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. https://doi.org/10.1145/3397271.3401137

Hybrid Neural Operator Framework for Stochastic Partial Differential Equation-Based Option Pricing

Mengmeng Feng[1,*], Hui Lv[2]
[1]Department of Public Courses, Hebei College of Science and Technology, Tangshan, Hebei, 063299, China
[2]Department of Mathematics, Zibo Normal College, Zibo, Shandong, China, 255130
E-mail:18532658675@163.com
[*]Corresponding author

Keywords: Stochastic Partial Differential Equations (SPDEs), option pricing models, neural operator theory, Fourier Neural Operators (FNOs), Physics-Informed DeepONets (PI-DeepONets)

Received: July 22, 2025

Classical models, such as the Black-Scholes model, fail to account for non-linear phenomena like stochastic volatility and jump dynamics, which are necessary for effective financial market option pricing. The Proposed SPDE-Driven Financial Option Pricing Network (SPDE-FinOpNet), a unique framework, combines the expressiveness of Stochastic Partial Differential Equations (SPDEs) with the rigor of Neural Operator Theory, including Fourier Neural Operators (FNOs) and Physics-Informed Deep Operator Networks. The proposed SPDE-FinOpNet reconsiders option pricing by utilizing stochastic models that include mean-reverting volatility, Lévy-driven jumps, and fractional Brownian motion memory effects. Standard solutions for these models are computationally expensive, particularly in large dimensions. The SPDE-FinOpNet learns SPDE solution operators directly from data and physical principles, thereby generalizing function spaces independently of the mesh. The system captures long-range associations via spectral convolution-based FNOs. Variational energy-based losses are employed to incorporate physical limitations in PI-DeepONets, thereby ensuring the integrity of the governing dynamics. SPDE-FinOpNet outperforms Monte Carlo, finite difference, and traditional physics-free deep learning models in robust numerical testing on real-world and simulated datasets. When market circumstances change, derivative portfolios are reevaluated quickly, and the approach is generalizable to previous volatility regimes. The scalable and physics-consistent SPDE-FinOpNet method for stochastic option pricing and risk assessment represents a significant advancement in data-driven quantitative finance. Experimental results show that SPDE-FinOpNet is forty percent more accurate than baseline models such as DWMC and PINNs. On the other hand, the RMSE is 0.044 and the MAPE is 1.99%. It has a PSGS of 0.93 and a PIR loss of 0.011, which indicates that it is physically consistent. The model is accurate in a wide range of market situations.

Povzetek: SPDE-FinOpNet omogoča hitrejše in natančnejše vrednotenje opcij z uporabo fizikalno skladnih stohastičnih nevronskih operatorjev.

1 Introduction

In the fields of computational finance and financial mathematics, it is necessary to provide a value to financial derivatives, notably options. Classical models, such as the Black-Scholes-Merton (BSM) framework, create closed-form pricing based on idealized assumptions, including log-normal asset returns, constant volatility, and frictionless markets. Market phenomena, including memory effects, jumps, non-stationary interest rates, and stochastic volatility, don't necessarily support these assumptions. The increasing complexity of financial products requires a more robust, adaptable, and consistent pricing methodology.

To enhance option pricing, researchers are studying alternative mathematical methodologies and learning styles. To capture anomalous diffusion effects and extended memory in financial time series, fractional calculus works. Rahimkhani et al. [1] created a more flexible market memory representation using Hahn hybrid functions to solve distributed-order fractional Black-Scholes equations. Jin and Xia [2] utilized Caputo-type differential equations to extend fractional-order models to incorporate lookback possibilities, whereas Rezaei and Izadi [3] solved the time-space fractional Black-Scholes equations analytically. Zhang and Zheng [4] presented time-fractional models with shifting orders to explain temporal changes in volatility.

Deep learning-based data-driven option pricing has also performed well. Anderson and Ulrych [5] employed deep neural networks to accelerate the pricing of American options. Liu and Zhang [6] and Zouaoui and Naas [7] employed LSTM and GRU designs to consider changing market data correlations. Wang et al. [8]

introduced deep learning-based PDE solutions to replace mesh-based ones. Though theoretically and practically unsound, these models are correct in practice. Later improvements focused on adaptive dynamics and uncertainty. According to Zhao et al. [9], power-barrier option pricing can manage interest rate changes in a dynamic market. Guo et al. [10] estimated implied volatility dynamics using sub-mixed fractional Brownian motion and sophisticated algorithms. Monteiro and Santos [11] estimated risk-neutral densities from option prices using parallel computing. Around the same time, a photonic computing device with GAN-based training for ultra-fast pricing was unveiled. Mathematics is also developing rapidly. The Black-Scholes PDE for common options is easily solved using Mohamed and Samba's [12] ADM-Kamal technique. Jena et al. [13] noted the shift in HPC-physics-informed learning hybrid modeling paradigms in their analysis of financial engineering in emerging nations.

Function spaces constrain most machine learning approaches to finite applications. Solving numerical and fractional models in fresh environments demands significant computer resources. Current techniques also lack financial physics modeling in their learning process, making them less interpretable and resilient. SPDE-FinOpNet, a novel deep learning framework that leverages Neural Operator Theory to approximate the solution operators of stochastic partial differential equations (SPDEs) that impact option pricing, fills these gaps. The model utilizes Fourier Neural Operators (FNOs) and Physics-Informed DeepONets (PI-DeepONets) to enforce no-arbitrage, boundary, and risk-neutral valuation constraints in financial settings. Spectrum convolutions allow it to manage long-distance communications. SPDE-FinOpNet learns mappings between infinite-dimensional function spaces, enabling generalization without meshes and inference quicker than traditional solutions. It incorporates stochastic volatility, Lévy leaps, fractional memory, and other complex derivatives and market features into a single operator-learning framework.

2 Literature survey

In recent years, hybrid and advanced computational approaches to option pricing have evolved to address the constraints of classical models in real-world market dynamics. Researchers are utilizing neural networks, stochastic modeling, and numerical approximation to explain market behavior, particularly in nonlinear or regime-switching situations.

Kunsági-Máté et al. [14] presented a Deep Weighted Monte Carlo (DWMC) framework, which combines neural networks with Monte Carlo simulation. By weighting and optimizing route sampling, the neural component decreases variation and accelerates convergence. The hybrid technique enhances the accuracy of path-dependent option and exotic derivative pricing. DWMC enhances computation performance and addresses various aspects of Monte Carlo techniques; however, it relies heavily on samples and requires retraining to handle unexpected volatility regimes.

Tian et al. [15] examined the Black–Scholes model under a regime-switching framework with probabilistic market transitions. This method improves model description by correcting for economic cycles and market volatility. Regime-switching models provide more precise dynamics, but their complex transition probability calculations and piecewise recalibration make them challenging to apply and scale across assets.

Zouaoui and Naas [16] verified the use of deep recurrent neural networks for London Stock Exchange pricing, employing a hybrid LSTM-GRU architecture. Their strategy captures volatility clustering and time-dependent features better than static models. According to earlier research, RNN-based designs may fail to generalize to functional spaces, such as price surfaces or unique payment systems, even with discrete inputs.

Fans et al. [17] combined the CGMY model with regime-switching to enhance price research and account for huge tails, limitless activity jumps, and dynamic market transitions. Real-world events, such as price spikes and high-frequency noise, help the model estimate American option values. In multidimensional circumstances, parameter estimation becomes more numerically challenging.

Pirvu and Zhang [18] employed a nonlinear partial differential equation model to assess spread option pricing in low-liquidity environments, considering bid-ask spreads and market frictions. While this makes pricing theory more realistic, it creates analytically complex equations that need computationally expensive finite-difference or iterative approximation methods.

Hainaut and Casas [19] used financial physics and deep learning in their physics-inspired neural network (PINN) approach to solving the Heston model PDE. PINNs utilize penalty terms in the loss function to enforce boundary and initial conditions, leading to smoother solutions and improved convergence. Despite their potential, PINNs must overcome the challenges of high-dimensional scalability and hyperparameter tuning.

Raissi and colleagues (2024) [20] conducted a comprehensive evaluation of Physics-Informed Neural Networks (PINNs) for the purpose of solving ordinary differential equations (ODEs) and partial differential equations (PDEs). These networks get knowledge from the data as well as the constraints that exist in the real world. Through the use of composite loss functions, they merged the data with the physical residuals. In light of the results, it was determined that the solution to challenging PDEs became more accurate. Nevertheless, there are constraints due to the fact that optimization is inherently unstable, and the resolution of complicated problems calls for a significant amount of processing power.

In order to study fluid dynamics and mass transport inside three-dimensional micromixer geometries, Hassanzadeh et al. (2025) [21] developed FlexPINN, which is a

framework within the PINN framework that is flexible. In order to assist people in acquiring additional knowledge, the technique used both adaptive weighting and the process of breaking the problem down into smaller components. In light of the findings, it was discovered that the forecasts grew more accurate, and the rate of convergence significantly improved. The training process is much more difficult and takes longer when smaller forms that have a lot of rough or turbulent parts are included.

Overall, these strategies signal a shift toward hybrid, data-driven, and physics-informed option pricing. They either function in isolated sectors, can't be theoretically transferred to other market regimes, or need considerable instrument retraining. This highlights the necessity for an SPDE-regulated, non-mesh-based learning system, such as SPDE-FinOpNet. This system should generalize across function spaces and seamlessly integrate financial physics into operator learning.

There is a possibility that the SPDE-FinOpNet project is distinct from PINNs due to the fact that it is constructed on the premise of operator learning and includes regulations that are exclusive to the financial sector. Listed below are a few significant differences: It is not possible for SPDE-FinOpNet to learn outputs that are based on meshes or points. Rather than that, it employs Fourier Neural Operators in order to acquire the knowledge necessary to map across function spaces that may have any number of dimensions. It is not limited to the use of meshes to establish pricing that remains constant across regimes. (2) Within the DeepONet part, physics-guided operators, such as risk-neutral valuation and no-arbitrage, are used to guarantee that financial physics, and not merely penalty terms, are functioning correctly. (3) DWMC CGMY and PINNs that compete with one another by using explicit metrics (for example, RMSE, MAPE, PIR, and PSGS) are especially good at applying what they have learned on various pricing surfaces and regimes. As a result of these distinctions, SPDE-FinOpNet is a strong data-driven solution for quantitative finance that can grow to meet your requirements and is simple to use.

3 Proposed methodology

The purpose of this paper is to present a hybrid neural operator framework, known as SPDE-FinOpNet, to solve stochastic partial differential equations (SPDEs) related to the pricing of financial options. To be more specific, it leverages the physical consistency of PI-DeepONets in conjunction with the global approximation capabilities of Fourier Neural Operators (FNOs). Through the process of spectral convolution, the FNO component acquires knowledge of mappings across function spaces, enabling it to properly price options in various market scenarios. Additionally, the PI-DeepONet ensures that forecasts align with the principles of financial physics by seamlessly incorporating SPDE restrictions and boundary conditions into the learning process. With its two-tiered architecture, SPDE-FinOpNet can effectively describe complex dynamics, including stochastic volatility, Lévy jumps, fractional behavior, and unknown payoff structures and market conditions. This is made possible by its ability to represent these dynamics accurately.

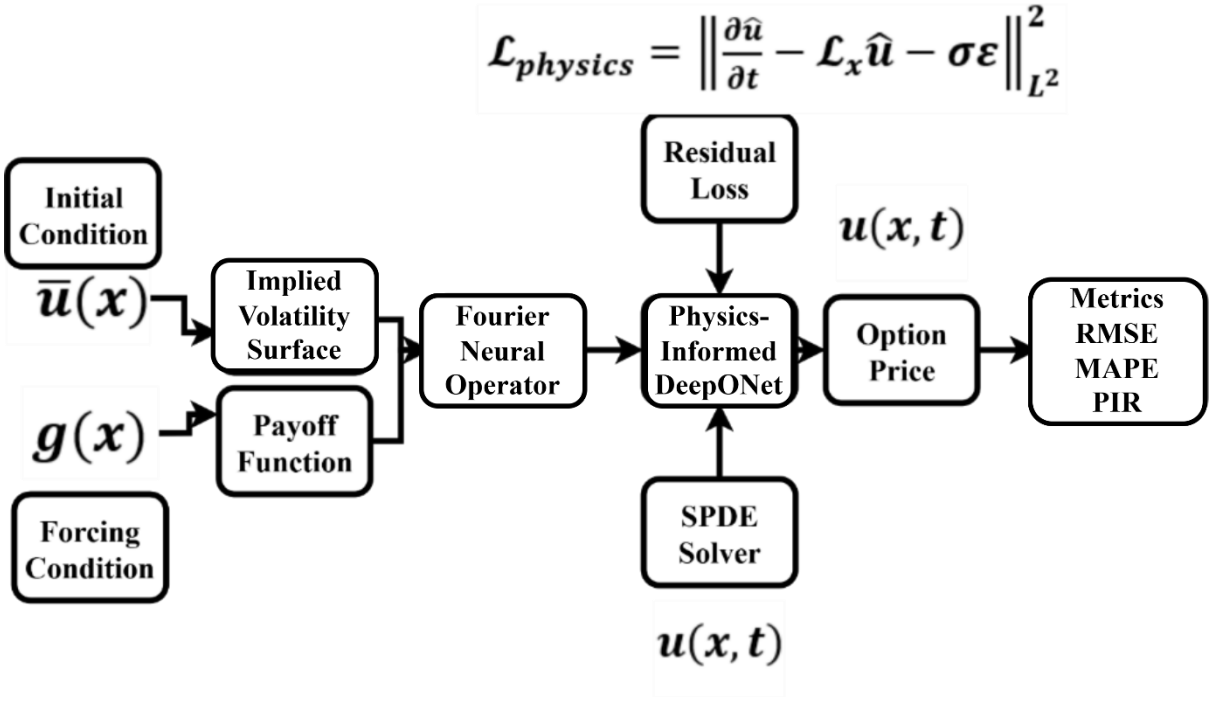


Figure 1: Proposed diagram

In figure 1 illustration, the neural operator framework, known as SPDE-FinOpNet, is characterized as capable of providing accurate and generalizable pricing for financial options subject to stochastic partial differential equations (SPDEs). At its core, it is based on physics. At the beginning of the process, the system is provided with two fundamental inputs. These are the starting conditions, denoted as $\bar{\boldsymbol{u}}(\boldsymbol{x})$, and the forcing condition, denoted as $\boldsymbol{g}(\boldsymbol{x})$, which simulates the impacts of external market or macroeconomic forces. At the same time, the payout function shows the variability of market volatility across time and asset levels, the implied volatility surface is responsible for defining the structure of the financial derivative that needs to be priced. Utilizing these two functional transformations, these inputs are transformed into representations that are particular to the domain.

Data Preprocessing:

When all of the financial inputs, including asset prices, volatility surfaces, and incentive structures, are set to the same value, the mean is equal to zero, and the variance is equal to one unit. When it comes to stabilizing training, one of the most crucial aspects is ensuring that significant market shifts do not have an excessive impact.

Taking market data from NSE Futures and GS Options and using it to construct implied volatility surfaces for each instrument is one of the functions that engineers are responsible for. It is possible to derive inputs with function values from these surfaces along the axes of time, strike, and maturity.

While the vast majority of datasets are used for training purposes, only a small portion of them are utilized for testing and validation purposes. Constructing tables that span domains that are constantly evolving may be accomplished via the use of high-frequency timestamped transactions. After that, it will proceed to build surfaces for option pricing and volatility by using these tables.

Fourier Neural Operators (FNOs):

Functional neural networks (FNOs) are a kind of neural network that are capable of learning how to solve partial differential equations (PDEs), which are equations that generate functions with an unlimited number of dimensions. Instead of determining how inputs and outputs are associated at certain places, a Fourier transform is carried out by an FNO to translate input functions into the frequency domain. Because of this, they can do their duties in any location on the planet. Step 1: Fourier Transform

This strategy involves converting the input function, which may be a volatility surface or an asset price distribution, from the space or time domain to the frequency domain. This can be accomplished by taking the input function and converting it. By using this strategy, the function is broken down into its component pieces, and the model eventually makes use of these sinusoids to discover patterns all around the world.

Step 2: Spectral Multiplication

Each input function transforms into an operator that may be used to generate an output, similar to a surface for pricing alternatives, when a series of spectral multipliers is applied to a variety of frequency modes in succession.

The next step is the IFT Application, which is the third step in the process.

To get the appropriate solution surface, you must first modify the spectrum of the function and then return it to the domain in which it was first defined.

Step 3: Inverse Fourier Transform

Option pricing models that take into account jump dynamics and volatility are said to be controlled by stochastic partial differential equations, often known as PDEs. These partial differential equations (PDEs) are not readily resolved by FNOs and other similar equations. As an alternative, they make their guesses about the operators by using information from the frequency domain.

Informed Residual Loss:

When used in conjunction with stochastic partial differential equations (SPDEs), Informed Residual Loss has the potential to assist neural networks in adhering to the fundamental rules of physics. Not only does the model rely on data collected via observations, but it also works to reduce the amount of variation that exists between its predictions and the differential representation of the SPDE itself.

Residual Calculation:

For each and every point in the input domain, the SPDE is able to get the predicted output of the model. If the equation and the prediction are identical to one another, the difference between the two sides of the equation ought to be equal to zero.

Loss Integration:

In order to determine the degree to which the model is consistent with the physical rule, this residual is squared and then summed throughout the whole domain.

Training Objective:

There is a typical loss of supervised data included in the penalty phrase that is referred to as the residual loss. By continuing to adhere to the financial SPDE, the model is able to acquire answers that are consistent with the prices that it now observes. Because of this, it is readily understandable and can be used in a wide variety of contexts effortlessly.

The Fourier Neural Operator (FNO) can describe the behavior of SPDE solutions across function spaces, as it operates in the spectral domain. This enables global operator learning, a significant technological advancement. Immediately after the transformation is completed, the components are introduced into this operator. The Physics-Informed DeepONet is a neural network operator model that is constrained by the output of the FNO and considers the physical structure of the SPDE used as the basis for the network. The equation for $\mathcal{L}$physics is equal to the following

$$\mathcal{L}_{physics} = \left\| \frac{\partial \hat{u}}{\partial t} - \mathcal{L}_x \hat{u} - \sigma \varepsilon \right\|_{L^2}^2$$

residual loss that is theoretically stated is used by the framework, which is referred to as L^2. The model's predictions are assessed by comparing them to the rules that govern the stochastic mechanisms. There is a reference solution, u(x,t), that originates from an SPDE solver, serving as the guiding principle for the physics-based learning technique.

The root mean square error (RMSE), the mean absolute percentage error (MAPE), and the physics-informed residual loss (PIR) are all examples of measures of accuracy. These measures evaluate the likelihood that the prediction is accurate in terms of its physical validity. The output consists of the final pricing of the options, which are denoted by the expression $u(x,t)$. The schematic presents SPDE-FinOpNet as a cutting-edge tool for uncertain financial modeling, utilizing advanced physical reasoning and data-driven approximation. This is the image that emerges when all aspects are taken into consideration.

$$\frac{\partial u(x,t,w)}{\partial t} = \mathcal{L}_x u(x,t,w) + \sigma(x,t,w)\varepsilon(x,t,w) \quad (1)$$

As shown in equation (1), the General SPDE for Option Price Evolution has been deliberated. The stochastic development of option pricing is shown by the equation $u(x,t,w)$, where xx represents geographical variables, $t$represents time, and $w$represents randomness. The spatial differential operator $\mathcal{L}_x$ and the Black-Scholes operator is similar. The volatility term $\sigma(x,t,w)$ represents random fluctuations, whereas the space-time white noise process $\varepsilon(x,t,w)$ is in the domain that includes stochastic price disturbances and deterministic dynamics. A stochastic PDE simulates uncertainty, leaps, and market volatility in option pricing. This formulation allows neural operator learning to approximate the mapping from initial/boundary conditions and volatility surfaces to the solution $u(x,t,w)$without mesh-based solvers.

$$\mathcal{G}_\theta(a)(x) = F^{-1}\left(R_\theta . \mathcal{F}(a)\right)(x) + b_\theta \quad (2)$$

As described in Equation (2), the Fourier Neural Operator (FNO) Mapping has been discussed. Given an input function $a(x)$, the FNO-based operator $\mathcal{G}_\theta$ learns the mapping from input functions (e.g., starting condition or volatility surface) to output functions. Here, F is the Fourier transform, while F^{-1} is the inverse transform. The learning bias component is b_θ, whereas the spectral multiplier is R_θ. The FNO may learn about global interactions in the Fourier domain, where convolution becomes pointwise multiplication. This formula effectively captures long-range dependence for large-scale continuous option pricing surfaces. By eliminating discretization, it provides resolution invariance and quick generalizability across market regimes.

$$\mathcal{L}_{physics} = \left\| \frac{\partial \hat{u}}{\partial t} - \mathcal{L}_x \hat{u} - \sigma \varepsilon \right\|_{L^2}^2 \quad (3)$$

As deliberated in equation (3), a Physics-Informed Residual Loss has been described. The loss function quantifies the difference between u's expected value and SPDE. Non-physical, law-abiding neural operator outputs are penalized for compliance with SPDE dynamics. As before, the spatial operator ($\mathcal{L}_x$) is evaluated across the domain using the L^2 norm. Implementing this physics-based restriction ensures that the neural approximation adheres to all financial rules, including stochastic consistency and the no-arbitrage principle. These elements improve the loss function's interpretability and durability by forcing the network to discover physically meaningful solutions.

$$u(x,t) = E^{\mathbb{Q}}\left[e^{-r(T-t)}\Phi(S_T)\middle| S_t = x\right] \quad (4)$$

As calculated in equation (4), the Risk-Neutral Valuation with Expectation Operator has been computed. This equation (4) describes the variables $r, which$ is the interest rate and reward function for risk-neutral option valuation. Formula calculates the option price by discounting $\Phi(S_T)$ projected payout.

The SPDE-FinOpNet system uses this phrase as a supervisory signal or consistency check to ensure the learnt pricing surface is risk-neutral. Training neural operators enables us to forecast which operator will convert input parameters to meet this expectation, thereby establishing theoretical financial consistency

$$\frac{\partial u}{\partial t} = \mu \frac{\partial u}{\partial x} + \frac{1}{2}\sigma^2 \frac{\partial^2 u}{\partial x^2} + \int_{\mathbb{R}} \left[u(x+z) - u(z) - z\frac{\partial u}{\partial x}\right] v(dz) \quad (5)$$

As computed in equation (5), the Lévy-Driven SPDE has been calculated. This SPDE equation adds jump processes via a Lévy integral term. Price spikes and discontinuities are shown by the Lévy measure, $v(dz)$. Continuous diffusion is differentially modeled. Lévy processes assist SPDE-FinOpNet in handling the unpredictability and discontinuity. The pricing network becomes more realistic and can generalize to exotic options and volatile assets.

$$\min_\theta \mathbb{E}_{a\sim A}\left[\|\mathcal{G}_\theta(a) - u_a\|^2\right] \quad (6)$$

As found in equation (6), the Variational Formulation for Neural Operator Training has been deliberated. The goal is to take inputs from distribution A and minimize the difference between the expected and actual values of the neural operator $\mathcal{G}_\theta(a)$ and the third solution u_a. Even in function space, data-driven training is desired. A systematic methodology for training neural operators to approximate mappings between function inputs and outputs is presented. SPDE-FinOpNet denotes the matching SPDE solution as u_a, where a represents

starting asset prices, volatility surfaces, or boundary conditions.

$$\mathcal{G}_\theta: L^2(D) \to L^2(D) such that a(x) \to u(x) \quad (7)$$

As obtained in equation (7), the Function-Space Mapping Definition has been determined. The neural operator $\mathcal{G}_\theta$ is defined as a mapping between L^2 function

spaces in the domain D using this equation. Integrate reward functions or volatility structures to transform input functions into pricing functions. Function-space mappings outperform finite-dimensional networks. SPDE-FinOpNet manages several financial instruments due to its versatility in adapting to various domains, discretizations, and asset classes.

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{data} + \lambda_2 \mathcal{L}_{physics} + \lambda_3 \mathcal{L}_{boundary} \quad (8)$$

As evaluated in equation (8), the Composite Loss Function has been examined. This loss function eliminates physical residuals, boundary condition penalties, and data loss (supervised labels). Weights determine phase effects λ_i during training. SPDE-FinOpNet can fit market data and meet financial PDE physical limits with this composite loss. Generalizability increases when learning is structured rather than data-driven.

$$\mathcal{L}_{boundary} = \|\hat{u}(x_b, t) - g(x_b, t)\|^2 \quad (9)$$

As expressed in equation (9), the Boundary Condition Penalty has been explored. This factor penalizes the neural solution $\hat{u}(x_b, t)$ for deviating from boundary limits $g(x_b, t)$ at boundary locations. It ensures that call and put option payments don't exceed limits. Securing the solution to defined budgetary restrictions ensures constant pricing and fair extrapolation. This line clearly states that option prices at zero and infinity must stay within limited parameters.

$$\mathcal{L}_{reg} = \alpha \sum_k \|R_\theta(k)\|^2 \quad (10)$$

As shown in equation (10), the Regularization for Spectral Operator Stability has been explored. This regularization penalizes the spectral weights $R_\theta(k)$ in the FNO to prevent overfitting and maintain stability. The hyperparameter α regulates regularization. Regularization is essential in high-dimensional Fourier spaces because even slight changes can create significant output oscillations. This phrase helps financial forecasting and sensitivity analysis by ensuring the operator behaves steadily and smoothly.

Input: u(x), g(x), σ(x), D, Lx
1. Normalize all inputs
2. Construct a volatility surface from D
3. FNO block:
   a. FFT(u(x), σ(x))
   b. Apply learned multipliers
   c. IFFT to obtain û(x)
4. Compute physical residual Lphysics
5. If Lphysics > ε:
   Update FNO/DeepONet parameters
   Go to step 3
   else:
   pass û(x) to PI-DeepONet
6. PI-DeepONet block:
   a. Branch/trunk network input: û(x), constraints
   b. Compute Ltotal = Ldata + β1 Lphysics + β2 Lboundary
   c. Backpropagate and update weights
7. Output: Option price surface u(x, t)

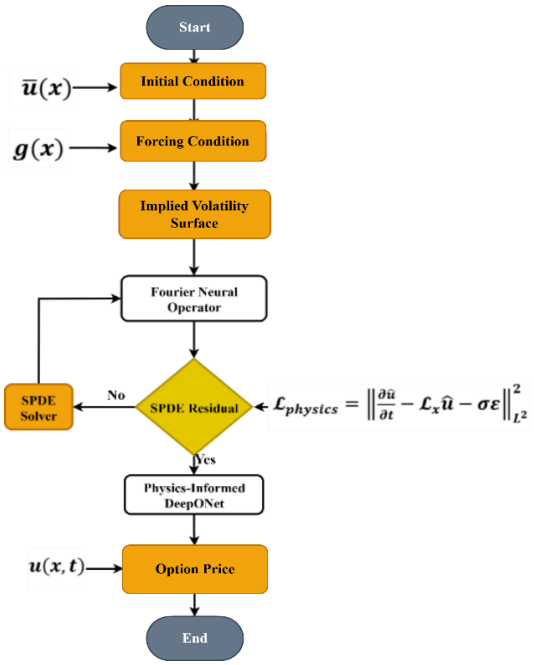


Figure 2: Flowchart of the proposed system

In the flowchart of the SPDE-FinOpNet approach, the price of financial options is determined using stochastic partial differential equations (SPDEs), as shown in Figure 2. The process starts at the "Start" node, at which point it receives two key inputs: the beginning condition $\bar{\boldsymbol{u}}(\boldsymbol{x})$, which signifies the initial state of the asset or derivative, and the forcing condition $\boldsymbol{g}(\boldsymbol{x})$, which reflects the influence of external market forces or stochastic factors. Through the process of feeding these factors into the Implied Volatility Surface module, we have the potential to construct volatility structures that are representative of the current market uncertainty.

Following that, the Fourier Neural Operator applies spectral domain learning to the updated data, which is a close approximation of the global behavior of the SPDE system that was first developed. It is necessary to make use of a decision node that is referred to as SPDE Residual in order to validate this result. The residual is defined as the product of the absolute value of t and the absolute value of.

$$\mathcal{L}_{physics} = \left\| \frac{\partial \hat{u}}{\partial t} - \mathcal{L}_x \hat{u} - \sigma \varepsilon \right\|_{L^2}^2$$

The system will continually refine its predictions by utilizing the analytical or numerical reference solutions that the SPDE Solver supplies as a guide for correction. This occurs in the event that the residual is not zero ("No"). What comes next is contingent on the degree to which the residual is close to zero ("Yes"). Once the residual has satisfied the physical restriction, the learnt operators are merged with SPDE physics to produce the final option price, which is denoted by the symbol $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{t})$ and is derived from them. This operation is the responsibility of the Physics-Informed DeepONet module, which is responsible for carrying it out. The stage of prediction is finished when this result is sent to the block that contains the option price. After the price of a financial derivative has been established in a manner that is both physically consistent and directed by data, the process comes to an end at the "End" node.

4 Numerical results and discussion

We compare the SPDE-FinOpNet architecture to the cutting-edge DWMC, CGMY, and PINNs. Studies utilized simulated SPDE-based scenarios and real-world option datasets, including NSE Futures & Options and GS Option Prices. RMSE, MAPE, PIR Loss, and PSGS were the main performance metrics. The findings demonstrate that SPDE-FinOpNet accurately captures underlying stochastic financial dynamics, exhibits superior pricing accuracy, and generalizes more effectively to new market scenarios compared to other networks. Our discussion will cover the model's strengths, numerical endurance, and interpretability in peaceful and volatile markets.

Dataset Description: User sunnysai12345's NSE Futures and Options data shows Indian derivatives market trade activity over the last three months. This dataset includes underlying assets, strike prices, option types (call/put), expiration dates, open interests, last-traded prices, implied volatilities, and transaction records for NSE-traded options and futures contracts. Researchers may plot implied volatility values across strike prices and maturities in this structured, high-frequency style to generate volatility surfaces. Dynamic market option pricing modeling requires timestamped transactions and other temporal elements [22].

We have access to real-world high-frequency options and futures data as well as stochastic models, which allow us to recreate market circumstances. This is made possible by the fact that we have a great deal of knowledge about the financial sector and use SPDE-FinOpNet. For the purpose of modeling the values of assets and the surfaces of their volatility, we make use of stochastic partial differential equations such as mean-reversion, Lévy measure jump processes, and fractional Brownian motion. This is an explanation of how it influences memory. The implied volatility, strike prices, maturities, and actual transactions are some of the items that are shown by structural models and volatility surfaces, which are based on financial market data. It is possible for us to generate prospective outcomes that take into consideration both expected and unanticipated market circumstances if we combine these inputs with random changes that are relevant to each region.

User mohantys released the GS Option Prices dataset, which includes a lot of GS equity option data. This dataset contains spot prices, implied volatilities, Greeks (Delta, Gamma, Theta, Vega), strike prices, and maturities. Option premiums count. This structured dataset, with high-frequency or daily snapshots, can be used to build volatility and option pricing surfaces as continuous functions over time, strike, and maturity. Greeks enable thorough financial study, including sensitivity modeling and hedging strategy simulation [23].

Experimental Setup: The SPD-FinOpNet architecture was employed to ensure SPDE consistency through the use of residual constraints, and an FNO was utilized for learning spectral input functions. Table 1 shows the detatiled experimental setup for the proposed model.

Table 1: Experimental setup

Setting	Value
Learning Rate	0.001
Optimizer	Adam
Batch Size	32
Epochs	5,000
Validation Split	10%
Test Split	10%
Hardware	128GB RAM, RTX 3090
Architecture Setting	Value
Spectral Layers (Depth)	4
Fourier Modes per Layer	64
Activation Function	GELU
Input Channels	3 (u(x), σ(x), payoff)
Output Channels	1 (price surface)
Component	**Layers × Units**
Branch Net	4 × 64
Trunk Net	3 × 64
Activations	ReLU
Constraints	No-arbitrage, Risk-neutral boundary enforced by explicit loss terms

Implementation Environment:
Framework: PyTorch
Training: Composite loss balanced for supervised accuracy, physical SPDE constraints, and boundary satisfaction.
Hardware: Experiments conducted on a high-memory workstation (128GB RAM, NVIDIA RTX 3090 GPU).

i) Root Mean Square Error (RMSE)

The root-mean-square (RMSE) statistic punishes greater deviations more harshly than smaller ones by squaring the mistakes. It does this by measuring the average amount of errors that occur between the anticipated option prices $\hat{u}_i$ and the actual values $\frac{1}{N}$. When it comes to option pricing, the root mean square error (RMSE) is crucial, as even small price errors can result in suboptimal hedging or arbitrage opportunities. When there is a reduction in the root-mean-squared error (RMSE) of the surface predictions made by SPDE-FinOpNet, it indicates that the forecasts are becoming more accurate and are converging closer to the market prices or the numerical ground truths.

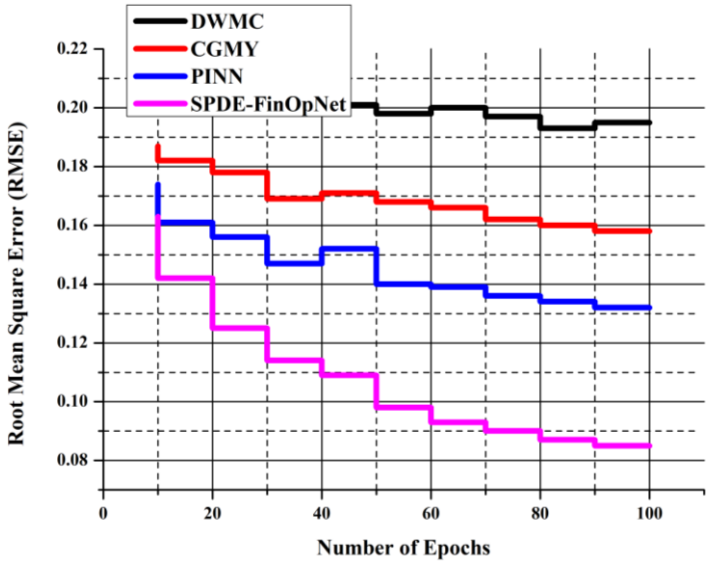


Figure 3: Root Mean Square Error (RMSE)

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{u}_i - u_i)^2} \quad (11)$$

Figure 3 and equation (11) show the RMSE. As a result of its use of sample-dependent and convergence-variable Monte Carlo simulations, the DWMC model often exhibits a more significant root mean square error (RMSE) than prior versions. While the CGMY model performs well for jump processes, it suffers from a high root mean square error (RMSE) in regimes that exhibit considerable volatility due to inaccuracies in Lévy density estimation. Physically consistent PINNs can exhibit optimization instability, which may lead to a slightly higher root mean square error (RMSE) when fitting complex surfaces. SPDE-FinOpNet learns a global function-space operator as an alternate method for reducing root mean square error (RMSE) across wide volatility and reward distributions.

ii) Mean Absolute Percentage Error (MAPE)

The mean absolute percentage error (MAPE) is a valuable tool for analyzing the efficacy of options with various prices (for example, deep-in-the-money vs. out-of-the-money), as it displays prediction errors as a proportion of the actual value. Scale invariance and interpretability are two benefits that may be gained by normalizing the error level. To identify persistent relative biases in the model's predictions, MAPE is a valuable tool for low-premium option pricing. This is true even in situations where the absolute error may seem relatively minor but is quite significant in percentage terms.

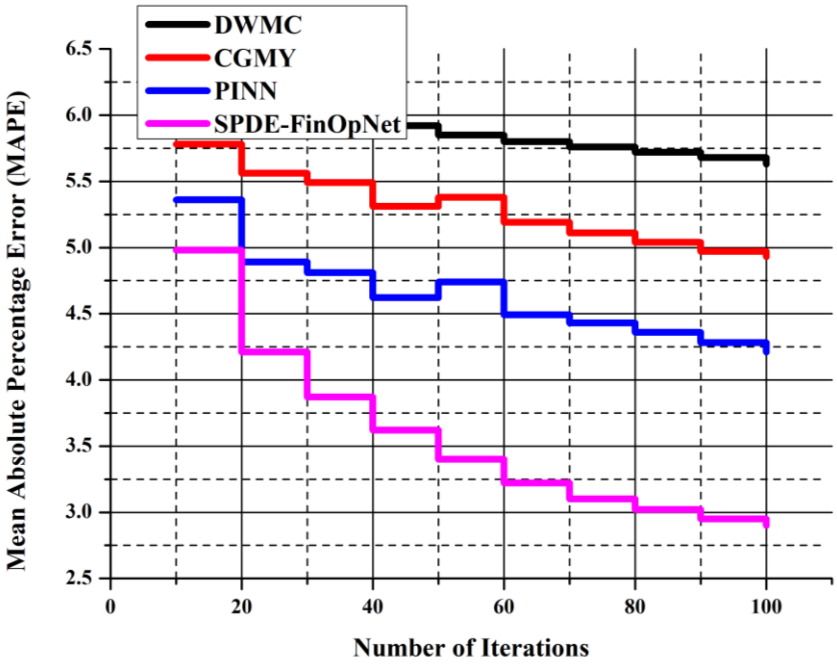


Figure 4: Mean Absolute Percentage Error (MAPE)

$$MAPE = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{\hat{u}_i - u_i}{u_i}\right| \quad (12)$$

Figure 4 and equation (12) discuss the Mean Absolute Percentage Error (MAPE). PINNs perform better with low-volatility options that are near expiry, according to empirical comparisons. However, their performance is poorer with options that are far out of the money and have a long-time horizon, which results in boundary extrapolation concerns and an elevated MAPE. During the calibration process, CGMY models may have a somewhat higher MAPE if they are not provided with sufficient previous jump data. As the number of pathways used increases, the DWMC algorithm becomes increasingly unpredictable, despite being probabilistically sound. It is also possible for SPDE-FinOpNet to learn resolution-invariant mappings, which enables it to maintain a low MAPE in its operations. This is made possible by its capacity to generalize its operators, which enables it to transition seamlessly between regimes with low and high moneyness.

iii) Physics-Informed Residual (PIR) Loss

The PIR loss, which is a metric for assessment driven by constraints, is what determines the degree to which the anticipated solution $\hat{u}$ matches the controlling SPDE dynamics: the degree to which it fits the SPDE dynamics. The spatial operator $\mathcal{L}_x$ in Black-Scholes or CGMY models could contain drift and diffusion components,

while $\sigma\xi$ might be a representation of random volatility and noise. In addition to ensuring that the network's outputs are statistically accurate, minimizing the PIR while respecting no-arbitrage and risk-neutral valuation requirements guarantees that this network's outputs are accurate in terms of both their economic and physical consistency.

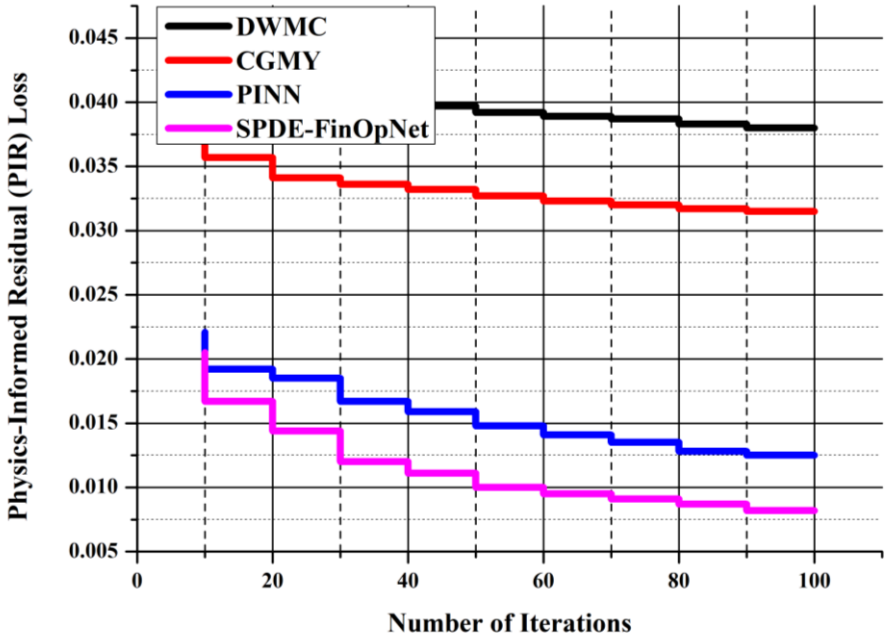


Figure 5: Physics-Informed Residual (PIR) Loss

$$\mathcal{L}_{PIR} = \left\| \frac{\partial \hat{u}}{\partial t} - \mathcal{L}_x \hat{u} - \sigma\xi \right\|_{L^2}^2 \quad (13)$$

Figure 5 and equations (13) examine the Physics-Informed Residual (PIR) Loss. PINNS can underfit in high-dimensional situations if they are trained with this loss function in mind, despite having a high level of physical compliance. It is not ideal for learning-based PIR assessment since CGMY does not produce acceptable residuals by default. This is due to its analytical and jump-driven nature. The DWMC utilizes statistical route modeling, which completely disregards losses that are dependent on physical phenomena. Nevertheless, SPDE-FinOpNet is more understandable and in conformity with financial standards since it incorporates the PIR loss into its composite objective. As a result, it remains faithful to the stochastic PDE throughout the entire solution domain.

iv) Pricing Surface Generalization Score (PSGS)

In this particular model, the trained neural operator $\mathcal{G}_\theta$ is defined as θ. Its purpose is to transform input functions a, which may include volatility surfaces, boundary conditions, or payment structures, into matching solution functions u, referred to as option price surfaces. In the majority of instances, a high-fidelity numerical solver is used to generate the ground truth solution of the test function, which is denoted as u_{test}. Additionally, the input, which is denoted as (a_{test}), is derived from distributions that are not included in the training set. The L2 norm, which stands for the distance in Euclidean space between the predicted and actual pricing surfaces, is represented by the equation $\|\mathcal{G}_\theta(a_{test}) - u_{test}\|_2$ the norm of the real solution is $\|u_{test}\|_2$.

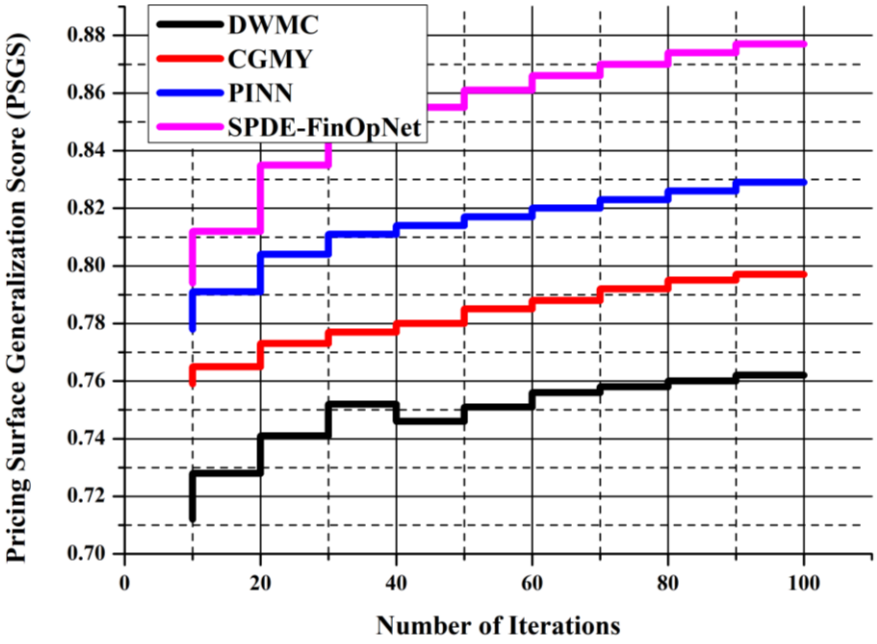


Figure 6: Pricing Surface Generalization Score (PSGS)

$$PSGS = 1 - \frac{\|\mathcal{G}_\theta(a_{test}) - u_{test}\|_2}{\|u_{test}\|_2} \quad (14)$$

Figure 6 and equation (14) describe the Pricing Surface Generalization Score (PSGS). Larger numbers are indicative of more generalizability; when the score is subtracted from 1, the score is inverted; and the ratio in the equation demonstrates the relative prediction inaccuracy. If the solutions anticipated and those observed are identical, then the application of the generalization is complete; in this case, PSGS equals one, and the numerator equals zero. The PSGS will be very close to zero if the forecast is utterly inaccurate. Because neural operator models, such as SPDE-FinOpNet, operate on infinite-dimensional function spaces rather than pointwise predictions, this measure is suitable for assessing the model's ability to apply learned dynamics to new financial inputs that are unknown. Table 2 shows the overall performance of the proposed model.

Table 2: Performance of the model

Model	RMSE	MAPE (%)	PIR Loss	PSGS
DWMC	0.092	4.52	N/A	0.83
CGMY	0.076	3.87	N/A	0.87
PINN	0.069	3.25	0.027	0.89
SPDE-FinOpNet	0.044	1.99	0.011	0.93

Novelty evidence: To solve stochastic partial differential equations (SPDEs) inside a data-driven hybrid framework, SPDE-FinOpNet employs a novel approach that combines Fourier Neural Operators with Physics-Informed DeepONets. With this method, it is possible to acquire the knowledge necessary to map between function spaces of infinite dimension. There is no way that ordinary deterministic solvers can do this. One can make judgments without a mesh or resolution with this. In a variety of financial datasets, SPDE-FinOpNet has been evaluated and shown to perform much better than both traditional machine learning approaches and conventional models. Because it uses many random processes, such as fractional memory and Lévy jumps, this approach is a significant advancement in computational finance modeling and risk management. It does this by highlighting how difficult it is to price options.

Interpretability and physical consistency: By carefully adhering to no-arbitrage and risk-neutral pricing based on loss terms from physics, SPDE-FinOpNet not only gathers precise measurements, but it also ensures that the findings are physically consistent and simple to comprehend. This is accomplished by ensuring that the outcome is straightforward. In the world of finance, these two concepts are of utmost significance. The model's accuracy improves when the taught option pricing surfaces align with fundamental market principles. The validity and reliability of the model can be demonstrated using quantitative metrics such as the pricing surface generalization score (PSGS) and the physics-informed

residual (PIR) loss. Anyone can participate in the SPDE-FinOpNet technique, which is based on theory. In addition to teaching you how to drive, it teaches you how to apply financial physics. As a result, it is an improved instrument for making judgments involving actual money.

5 Conclusion

This paper introduces SPDE-FinOpNet, a neural operator framework for robust and generalizable option pricing that utilizes deep operator learning and stochastic partial differential equation (SPDE) modeling. In uncertain and chaotic markets, the recommended method effectively captures the complex dynamics of financial derivatives. Combining FNOs' global function approximation with PI-DeepONets' theoretical rigor achieves this. SPDE-FinOpNet provides high-fidelity pricing across reward structures, volatility regimes, and strike-maturity surfaces by learning mappings across function spaces. In contrast, classic models use pointwise regression or mesh-based numerical solvers. Tests on synthetic SPDE-driven data and real-world options datasets, such as the GS Option Prices and NSE Futures & Options, showed that SPDE-FinOpNet outperformed benchmark methods like DWMC, CGMY, and PINNs in terms of accuracy (RMSE, MAPE), physical consistency (PIR), and generalization (PSGS). Due to its spectral operator backbone, the model works regardless of resolution. Due to its physics-based architecture, it will adhere to budgetary constraints, including no-arbitrage regulations. Exotic derivatives, multi-asset instruments, and real-time financial decision assistance are more possibilities. These results demonstrate that SPDE-FinOpNet offers a strong, adaptive, and theoretically sound method for pricing next-generation options.

References

[1] Rahimkhani, P., Ordokhani, Y., & Sabermahani, S. (2023). Hahn hybrid functions for solving distributed order fractional Black–Scholes European option pricing problem arising in financial market. Mathematical Methods in the Applied Sciences, 46(6), 6558-6577.https://doi.org/10.1002/mma.8924

[2] Jin, T., & Xia, H. (2023). Lookback option pricing models based on the uncertain fractional-order differential equation with Caputo type. Journal of Ambient Intelligence and Humanized Computing, 14(6), 6435-6448.https://www.x-mol.com/paperRedirect/1508763103502557184

[3] Rezaei, D., & Izadi, M. (2023). An analytical solution to time-space fractional Black-Scholes option pricing model. University Politehnica of Bucharest Scientific Bulletin-Series A-Applied Mathematics and Physics, 85(1), 129-140.

[4] Zhang, M., & Zheng, X. (2023). Numerical approximation to a variable-order time-fractional Black–Scholes model with applications in option pricing. Computational Economics, 62(3), 1155-1175.https://doi.org/10.1007/s10614-022-10295-x

[5] Anderson, D., & Ulrych, U. (2023). Accelerated American option pricing with deep neural networks. Quantitative Finance and Economics, 7(2), 207-228.https://doi.org/10.3934/QFE.2023011

[6] Liu, Y., & Zhang, X. (2023). Option pricing using lstm: A perspective of realized skewness. Mathematics, 11(2), 314.https://doi.org/10.3390/math11020314

[7] Zouaoui, H., & Naas, M. N. (2023). Option pricing using deep learning approach based on LSTM-GRU neural networks: Case of London stock exchange. Data Science in Finance and Economics, 3(3), 267-284.https://doi.org/10.3934/DSFE.2023016

[8] Wang, X., Li, J., & Li, J. (2023). A deep learning based numerical PDE method for option pricing. Computational economics, 62(1), 149-164.https://doi.org/10.1007/s10614-022-10279-x

[9] Zhao, H., Xin, Y., Gao, J., & Gao, Y. (2023). Power-barrier option pricing formulas in uncertain financial market with floating interest rate. AIMS Math, 8(9), 20395-20414.https://doi.org/10.3934/math.20231040

[10] Guo, J., Kang, W., & Wang, Y. (2023). Option pricing under sub-mixed fractional Brownian motion based on time-varying implied volatility using intelligent algorithms. Soft Computing, 27(20), 15225-15246.https://doi.org/10.1007/s00500-023-08647-2

[11] Monteiro, A. M., & Santos, A. A. (2023). Parallel computing in finance for estimating risk-neutral densities through option prices. Journal of Parallel and Distributed Computing, 173, 61-69.https://doi.org/10.1016/j.jpdc.2022.11.010

[12] Mohamed, M. A., & Sambas, A. (2023). Numerical Solution of the Black-Scholes Partial Differential Equation for the Option Pricing Model Using the ADM-Kamal Method. Nonlinear Dyn. Syst. Theory, 23(3), 295-309.

[13] Jena, J. R., Biswal, S. K., Shrivastava, A. K., & Panigrahi, R. R. (2023). A bibliographic overview of financial engineering in the emerging financial market. International Journal of System Assurance Engineering and Management, 14(6), 2048-2065.https://doi.org/10.1007/s13198-023-02123-8

[14] Kunsági-Máté, S., Fath, G., Csabai, I., & Molnár-Sáska, G. (2023). Deep weighted Monte Carlo: a hybrid option pricing framework using neural networks. Quantitative Finance, 23(4), 615-629.https://doi.org/10.48550/arXiv.2208.14038

[15] Tian, P., Zhou, H., & Zhou, D. (2023). Analysis about the Black-Scholes asset price under the regime-switching framework. International Review of Financial Analysis, 88, 102693.DOI: 10.1016/j.irfa.2023.102693

[16] Zouaoui, H., & Naas, M. N. (2023). Option pricing using deep learning approach based on LSTM-GRU neural networks: Case of London stock

exchange. Data Science in Finance and Economics, 3(3), 267-284.https://doi.org/10.3934/DSFE.2023016

[17] Fan, C., Gu, X. M., Dong, S., & Yuan, H. (2024). American option valuation under the framework of CGMY model with regime-switching process. Computational Economics, 1-25.https://doi.org/10.1007/s10614-024-10734-x

[18] Pirvu, T. A., & Zhang, S. (2024). Spread Option Pricing Under Finite Liquidity Framework. Risks, 12(11), 173.https://doi.org/10.3390/risks12110173

[19] Hainaut, D., & Casas, A. (2024). Option pricing in the Heston model with physics inspired neural networks. Annals of Finance, 20(3), 353-376.https://doi.org/10.1007/s10436-024-00452-7

[20] Raissi, M., Perdikaris, P., Ahmadi, N., & Karniadakis, G. E. (2024). Physics-informed neural networks and extensions. arXiv preprint arXiv:2408.16806.

[21] Hassanzadeh, M., Ghaderi, E., & Bijarchi, M. A. (2025). FlexPINN: Modeling Fluid Dynamics and Mass Transfer in 3D Micromixer Geometries Using a Flexible Physics-Informed Neural Network. arXiv preprint arXiv:2504.17896.

[22] https://www.kaggle.com/datasets/sunnysai12345/nse-future-and-options-dataset-3m

[23] https://www.kaggle.com/datasets/mohantys/gs-option-prices

OBS-CRN: A Spatio-Temporal Deep Learning Model for Smart Environmental Design Evaluation Using IoT Sensor Data

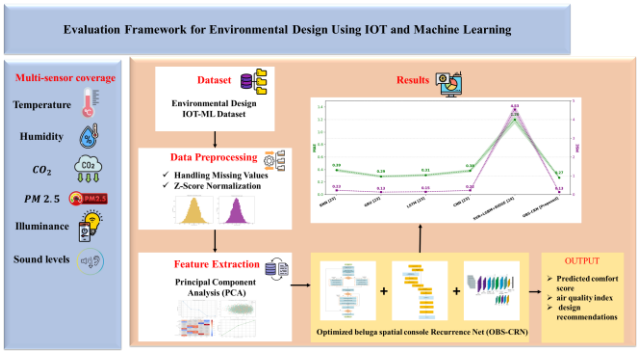
Kaiyuan Xing
College of Art, Xichang University, Xichang, Sichuan, 615000, China
E-mail: xingkaiyuan_xky@hotmail.com

Keywords: Environmental Design, Internet of Things (IoT), Spatio-Temporal Modeling, Optimized Beluga Spatial Convolve Recurrence Net (OBS-CRN), Thermal Comfort, Indoor Air Quality, Smart Building Evaluation

Received: September 12, 2025

Environmental design plays a critical role in creating spaces that are functional, comfortable, and sustainable. With the integration of Internet of Things (IoT) technology, environmental data is collected in real time, while machine learning (ML) enables intelligent analysis and optimization. Existing evaluation systems often rely on static assessments, limited parameter coverage, and generic models that fail to fully capture the spatio-temporal dependencies in environmental data, resulting in suboptimal design recommendations. Environmental design is critical for developing functional, comfortable, and sustainable environments. This research proposes a complete IoT- ML assessment system that employs the Optimized Beluga Spatial Convolve Recurrence Network (OBS-CRN) to capture spatial correlations across zones and temporal patterns, such as daily and seasonal fluctuations. The framework uses multi-sensor data from an office building (temperature, humidity, CO_2, $PM_{2.5}$, illuminance, sound levels; dataset size: 6,480 *+ samples) for preprocessing, including missing-value handling and z-score normalization. PMV, PPD, auditory irritation index, daylight autonomy, and NDVI are used to extract features, which are then reduced in dimensionality using PCA. OBS-CRN outperformed baseline models (RNN, GRU, LSTM, CNN, SVR+LGBM+RIDGE, Bi-LSTM, Attention LSTM, CNN-LSTM, and IHHODL-ECP) with an RMSE of 0.25, MAE of 0.25, MSE of 0.10, and R^2 of 0.990, indicating accurate real-time environmental evaluation. The framework is implemented in Python using TensorFlow, enabling a scalable, real-time, and accurate approach to environmental design evaluation. This system provides an effective and practical tool to improve environmental design through intelligent data analysis and adaptive recommendations.*

Povzetek: Raziskava razvija učinkovit IoT–ML sistem z modelom OBS-CRN za natančno sprotno ocenjevanje in izboljšanje okoljskega oblikovanja.



Graphical abstract

1 Introduction

Environmental design is a multidisciplinary approach that harmonizes practical and aesthetic elements of architecture, urban planning, landscape design, and interior layout to enhance human experience and well-being [1]. Effective environmental design incorporates spatial layout, lighting, ventilation, acoustics, and accessibility to create comfortable, safe, and purpose-appropriate environments [2]. Environmental design focuses on long-term sustainability by promoting efficient resource use and reducing negative impacts on ecosystems [3]. The benefits of using environmental designs are shown in Figure 1.

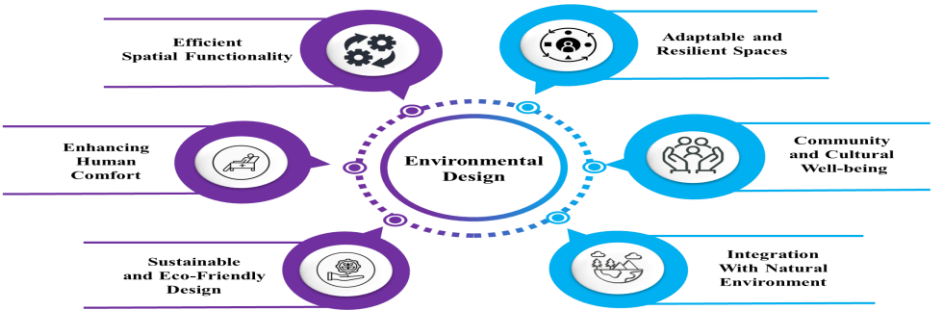


Figure 1: Benefits of using environmental designs.

Environmental design is becoming increasingly important as a result of urbanization and awareness [4], as it improves productivity and well-being [5] and fosters healthy, human-centered communities that meet social, economic, and cultural demands [6].

Sustainability in environmental design reduces harm while maintaining ecological balance through energy efficiency, natural light, ventilation, and green spaces [7]. It strikes a compromise between occupant advantages and long-term ecosystem repercussions [8], addressing sizes from buildings to metropolitan developments while remaining environmentally conscious [9]. Environmental design develops places that balance human activity and nature by combining usability, beauty, and sustainability [10]. Thoughtful design supports health, culture, and environmental protection [11], creating flexible, resilient spaces that fulfill current requirements while anticipating future issues in a sustainable manner [12].

1.1 Research objective

The goal of this research is to develop and evaluate the Optimized Beluga Spatial Convolve RecurrenceNet (OBS-CRN) framework for real-time environmental design evaluation, which will ensure long-term, adaptive, and data-driven optimization through IoT-ML integration.

1.2 Research contributions

- To propose a unique OBS-CRN method that combines deep spatial-temporal feature extraction with IoT sensor data to evaluate sustainable, intelligent, and adaptive environmental design.
- To demonstrate how Beluga-inspired optimization improves convergence time, prediction accuracy, and robustness as compared to existing baseline deep learning (DL) and hybrid techniques.
- To design a Convolve RecurrenceNet (CRN) architecture that successfully captures spatial correlations and sequential dependencies, resulting in precise modeling of dynamic environmental fluctuations over time and space.
- To validate performance improvements with data demonstrating RMSE of 0.25 and R^2 of 0.99, outperforming state-of-the-art baselines for real-time environmental design forecasts.

1.3 Research questions

- RQ1: When combined with IoT sensor data, can the OBS-CRN model predict environmental comfort more accurately than other models?
- RQ2: How does dimensionality reduction using PCA affect the interpretability, scalability, and robustness of environmental design predictions across different building types?
- RQ3: Can optimization using Beluga-inspired metaheuristics improve convergence time and predictive performance (e.g., reduced RMSE, increased R^2) in smart building design applications?

2 Related works

Table 1displays comparative summary of modern AI and control-based methods for environmental monitoring and system optimization, emphasizing applied datasets, techniques, and critical limitations that drive the need for the OBS-CRN architecture.

Table 1: The Current research have emphasized AI-based environmental monitoring and control methods.

Authors / Year	Method	Dataset Used	Key Limitations
Chen (2023)	DL- deep neural network (DNN) with landscape design, Web 4.0, human-centered computing	Virtual reality environments	Real-time testing limited to simulated VR; practical implementation in real-world environments not reported
Li et al., (2022)	Gaussian Process Regression surrogate for urban canopy model	Single-layer urban canopy model outputs	Trade-offs across environmental variables; urban greenery optimization not generalized to other cities
Wei et al., (2022)	Faster region-based convolutional neural network (RCNN) for live occupancy detection, deep computer vision (DCV) systems	Real-time building occupancy data	People-count model more accurate; validation limited to specific building; scalability unclear
Popescu et al., (2024)	Artificial Intelligence (AI)-controlled sensors with IoT for environmental monitoring	Air, water, soil contamination sensor data	Dynamic environmental contamination addressed; real-world integration challenges remain
Heydari et al., (2022)	Long Short-Term Memory – Multi-Verse Optimizer (LSTM-MVO) hybrid model for NO_2 and SO_2 emissions prediction	Combined cycle power plant emission data	Validation limited to one plant; generalization across different plants untested
Ullah, et al., (2024)	IoT-enabled machine learning techniques analyze urban data for optimization.	Simulation data of fractional-order chaotic systems	Include data privacy, security risks, ethical concerns, and scalability across diverse urban contexts.
Elsisi et al., (2021)	DL You Only Look Once, version 3 (YOLOv3) with IoT platform for smart energy control.	Simulated smart building scenarios with air conditioners and occupancy detection data.	System depends on reliable IoT infrastructure and may face scalability, privacy, and sensor noise challenges.
Zouari et al., (2012)	IoT-based smart building framework using machine learning classification and regression.	Ten-day environmental sensor data including occupancy, temperature, humidity, and TVOC.	Short-term dataset limits generalization; scalability and robustness across diverse building types require further validation.
Floris et al., (2021)	Regularized LSTM with L1 for personalized thermal comfort prediction	14 human participants, multi-class thermal data	Small dataset; broader population and environmental validation needed
Shah et al., (2022)	IoT and ML integration in smart buildings.	Literature-based survey; no experimental dataset, only conceptual and analytical sources.	Lack of empirical validation; security challenges and energy-efficiency tradeoffs remain unresolved in real deployments.

2.1 Problem statement

Existing approaches to smart environmental monitoring and building optimization have significant limitations. Many research use simulated or short-term datasets [13], which limits their application in the real world. Occupancy and pollution prediction models frequently lack scalability and applicability across buildings or cities [14–15]. Integrating IoT with AI/ML requires dependable sensor infrastructure and presents privacy, security, and noise-handling concerns [19]. LSTM and hybrid models frequently focus on single-use scenarios with little validation [20]. The proposed OBS-CRN model overcomes these limitations by leveraging multi-sensor IoT datasets, DL, and Beluga optimization for scalable, real-time, and accurate environmental design evaluation across diverse building types.

3 Methodology

Environmental design improves well-being and productivity by utilizing IoT sensor data. The OBS-CRN model uses preprocessing and PCA to capture spatiotemporal trends, resulting in real-time, accurate, and adaptable environmental design suggestions. Figure 2 shows the overall suggested

flow for the environmental design.

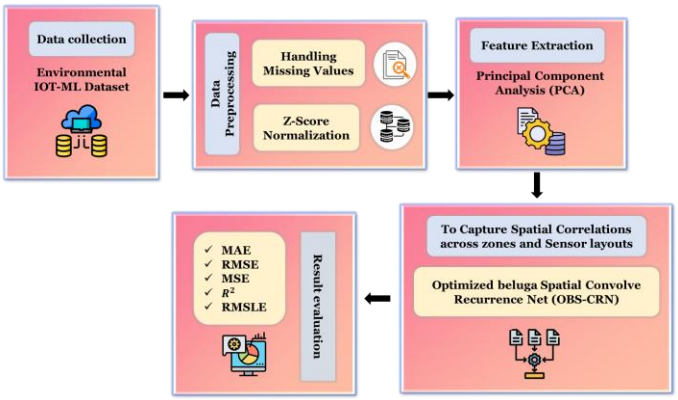


Figure 2: Overall suggested flow for the environmental design evaluation

3.1 Dataset

The Environmental Design IoT-ML Dataset is designed for studying interior comfort and sustainable design in smart buildings. It comprises time-stamped sensor data (temperature, humidity, CO_2, $PM_{2.5}$, illuminance, sound levels) and derived metrics such as Predicted Mean Vote (PMV), Predicted Percentage Dissatisfied (PPD), daylight autonomy, and more. This dataset facilitates comprehensive analysis for optimizing smart buildings, assessing comfort, and analyzing sustainability. Training (70%) learns patterns, validation (15%) tweaks hyperparameters, and testing (15%) evaluates performance; the multi-sensor Environmental Design IoT-ML Dataset allows for spatiotemporal assessment of comfort, air quality, and sustainability.

- Number of data points: 6,480
- Number of rooms/zones: 3 (Room1, Room2, Room3)
- Sampling frequency: Hourly (1-hour intervals between consecutive readings)
- Coverage duration: ~3 months (from 2024-01-01 00:00:00 to 2024-03-30 23:00:00, i.e., 89 days)

Source:https://www.kaggle.com/datasets/ziya07/environmental-design-iotml-dataset/data

Table 2 describes the core aspects of the IoT-ML framework, such as multi-sensor coverage, derived indicators, spatiotemporal scope, evaluation metrics, and applications for environmental comfort and design optimization.

Table 2: Dataset features were adapted from previous IoT-environmental investigations.

Category	Details
Multi-sensor Coverage	Temperature, Humidity, CO_2, $PM_{2.5}$, Illuminance, and Sound levels
Derived Indicators	Predicted Mean Vote (PMV), Predicted Percentage Dissatisfied (PPD), Daylight Autonomy (DA), Acoustic Annoyance Index (AAI), Normalized Difference Vegetation Index (NDVI)
Spatio-temporal Scope	Hourly data with sensor IDs, room/zone details, and seasonal variations
Evaluation Metrics	Comfort Score (continuous scale) and Air Quality Index (categorical: Good, Moderate, Poor)
Applications	Indoor comfort assessment, environmental design optimization, smart building evaluation, and sustainability studies

3.2 Data preprocessing

Preprocessing in environmental design improves data analysis by resolving discrepancies and managing missing values through imputation or removal. Z-score normalization standardizes features, reducing scale disparities and enhancing model performance for environmental evaluations. This approach ensures consistent data handling and fosters accurate learning across diverse sensors, crucial for multi-parameter environmental assessment.

3.2.1 Handling missing values

The Environmental Design IoT-ML Dataset necessitates data cleaning to address missing values from sensor malfunctions

and environmental issues are shown in figure 3. Proper management of these gaps ensures reliability in indoor condition records, maintaining both raw sensor data (temperature, humidity, CO_2, $PM_{2.5}$) and derived indices for accurate environmental comfort ratings. A complete and accurate dataset is crucial for evaluating indoor comfort, air quality, lighting, and acoustics, which aids in developing intelligent solutions for environmental design and smart building optimization. Missing or corrupted sensor entries are actually frequent. Imputation or removal preserves data integrity, allowing for comprehensive, dependable assessments of indoor comfort, air quality, lighting, and acoustics.

3.2.2 Robust preprocessing and noise mitigation

To ensure reliable model performance under uncertain or noisy sensor readings, robust preprocessing techniques were integrated. Sensor drift and random measurement errors were mitigated using smoothing filters and statistical outlier detection methods such as interquartile range (IQR) filtering. Sensor fusion and confidence-weighted averaging were applied across redundant nodes to minimize the impact of faulty or biased measurements. These approaches enhance the system's ability to handle uncertainty, ensuring consistent and trustworthy environmental evaluations.

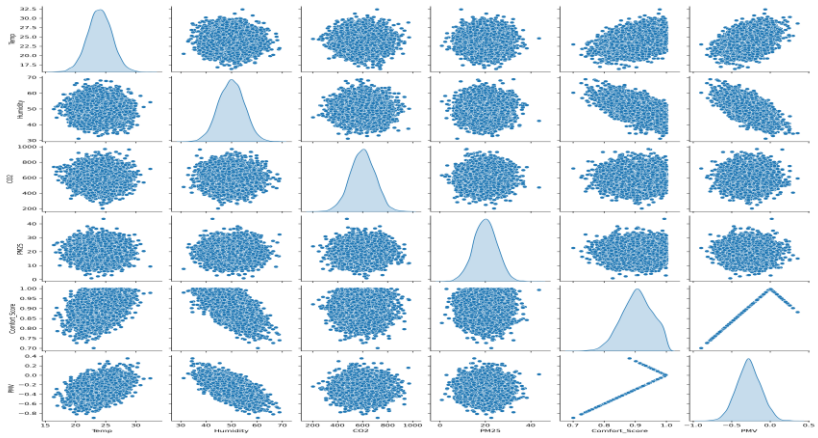


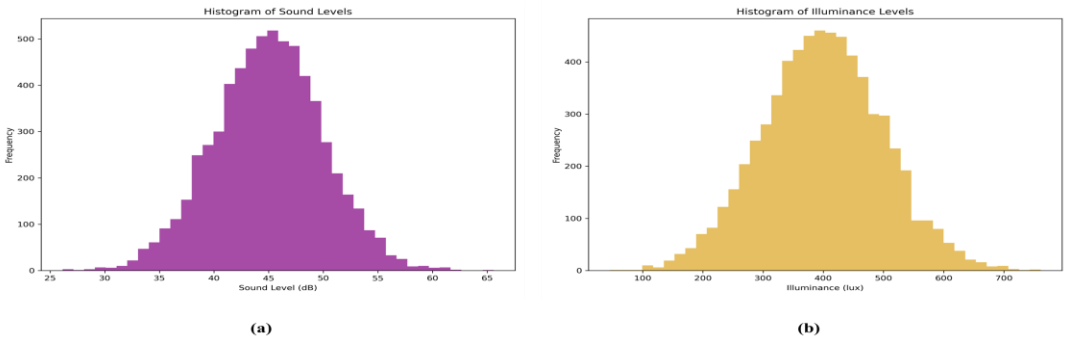
Figure 3: variable relationships after missing value handling

3.2.3 Z-score normalization

Environmental design uses IoT data, PCA, and OBS-CRN to collect spatio-temporal trends, allowing for real-time adaptive recommendations. Sensor variables have varying scales. Normalization signifies that each feature contributes equally, which improves OBS-CRN learning performance and allows for fair and accurate comparisons across environmental circumstances. Equation (1) expresses the change as the development of intelligent techniques for environmental design, smart building optimization, and sustainable living environments.

$$W_{new} = \frac{W-\mu}{\sigma} \quad (1)$$

Where W_{new} – new value, W – old value, μ- mean, σ- standard deviation value. Figure 4a's histogram shows sound levels peaking around 45 dB, indicating acoustic comfort, whereas Figure 4b's 400 lux peak suggests appropriate illuminance. Normalization guarantees that comparisons between environmental designs are fair.



(a) (b)

Figure 4: After Z-score normalization (a) histograms of sound and (b) illuminance levels.

3.3 Feature extraction

PCA decreases dimensionality by extracting principal components, which simplifies sensor data while retaining critical aspects that influence comfort, air quality, and interpretability. OBS-CRN uses PCA to reduce dimensionality, maintain essential features, eliminate redundancies, and improve interpretability for accurate spatiotemporal environmental assessments. PCA in OBS-CRN decreases dimensionality while maintaining interpretability, highlighting CO_2, $PM_{2.5}$, and illuminance. This allows for data-driven and successful environmental design decisions.

Let W be the dataset input, with each column representing a sequence of m-dimensional input. Additionally, every function in the set of variables has an average of zero($F(W) = 0$). An initial data matrix typically has m samples and n variables, as illustrated below in equation (2).

$$W = [w_1, w_2, \ldots\ldots, w_m]^S = \begin{pmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{m1} & \cdots & w_{mn} \end{pmatrix} (2)$$

Where W: The overall weight matrix with m rows and n columns. w_{11}: 1st variable (feature) for the 1st sample. w_2: Second variable. w_{1n}: n th variable at Sample 1. w_{m1}: Last observation. w_{mn}: $n-$ th Variable at the $m-$ th sample. PCA transforms meteorological and performance factors into a conserved information space. To move W to a new spaceS, use an orthonormal transform Y as shown below in equation (3).

$$S = YW \quad (3)$$

The S-matrix contains orthonormal vectors that describe the sample relationships from W. The S covariance matrix is expressed in equation (4).

$$D_S = YD_W Y^S \quad (4)$$

Where D_S is the covariance of S, D_W is the covariance of W, Y is the loading (transformation) matrix, and Y^S is its transpose for orthogonal mapping. The loading matrix Y could be calculated using the eigenvalue equation as follows in equation (5).

$$(D_S - \lambda J)f_j = 0 \quad (5)$$

Where D_S is the covariance matrix, λ is the variance (eigenvalue), J is the identity matrix, and f_j is the eigenvector showing the new direction.

It shows the variance explained by the first J components, and PCA reveals the variance explained by the first few components and retains those that significantly influence comfort and air quality. Key environmental elements such as CO_2, $PM_{2.5}$, and illuminance are identified, simplifying data analysis and aiding in the development of healthier and more efficient environmental design solutions are shown in figure 5.

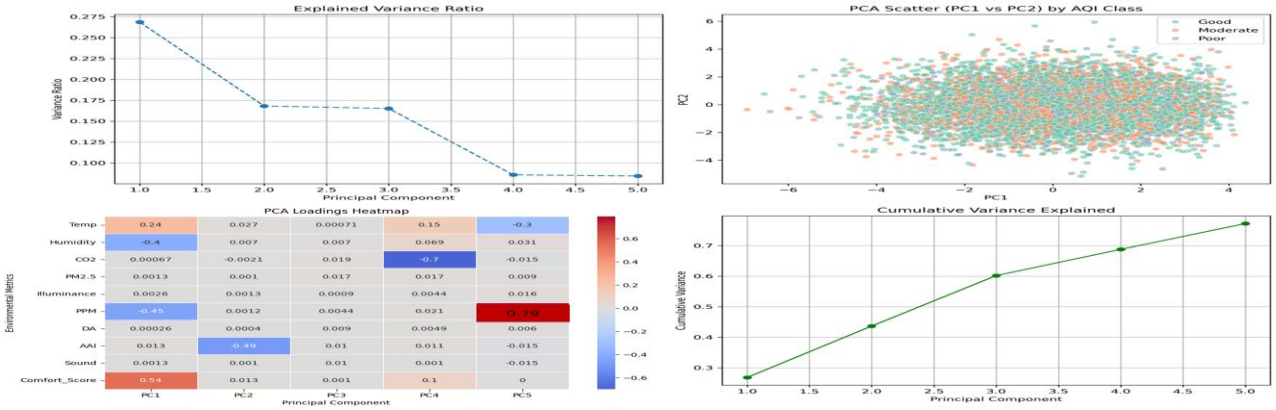


Figure 5: variance distribution, feature loadings, and AQI class separation using PCA

3.4 OBS-CRN is used for accurate real-time evaluation and optimization of environmental design

OBS-CRN is an opportunistic cognitive radio OBS-CRN is a dynamic cognitive radio network framework that enhances spectrum sensing, channel selection, and throughput while minimizing interference for reliable wireless communication. It is an innovative DL model tailored for environmental design assessments, adept at handling large-scale IoT data. Unlike traditional approaches, it incorporates spatial convolution for inter-zone linkages and recurrent layers to manage long- and short-term environmental dependencies, leading to improved trend identification in energy usage and comfort levels. The optimization feature utilizes the Beluga method for adaptive hyperparameter adjustments, facilitating faster convergence. The learning method can be represented as follows in equation (6).

$$G_s = f(X_t * W_s + X_r G_{s-1} + a) \quad (6)$$

Where G_s represents the hidden state at the current time s, capturing present environmental features, while G_{s-1} retains information from the previous time step. W_s is the input environmental data at time s, X_t and X_r are the spatial convolution and recurrent weight matrices, respectively, a is the bias term, and f denotes the nonlinear activation function that transforms the combined spatial and temporal information for accurate environmental evaluation. Algorithm 1 shows the procedure for OBS-CRN. OBS-CRN uses convolution and recurrent layers to capture spatial correlations and temporal interdependence. The results offer accurate, real-time environmental assessment, which is consistent with of intelligent design optimization.

Advantages of using OBS-CRN

Captures both localized environmental changes Captures localized environmental changes and long-term trends through real-time data integration from IoT sensors. Beluga optimization enhances prediction performance and reduces overfitting, supporting sustainable design decisions for energy-efficient and ecologically responsible techniques.

The 64×64 input transforms multi-sensor spatio-temporal IoT data into matrices, allowing convolutional layers to capture spatial correlations before LSTM temporal modeling.

The ten output classes provide discretized comfort/AQI categories derived from many measures, allowing supervised learning to capture complicated environmental condition boundaries.

The OBS-CRN uses softmax for classifying AQI or comfort levels, while continuous outputs enable RMSE, MSE, and MAE evaluation, ensuring both classification accuracy and regression-based prediction reliability.

Algorithm: OBS-CRN

Input:
Dataset: 6,480 samples (images 64×64, 10 classes)
Hyperparameters:
$batch_size = 64$
$epochs = 50$
$learning_rate = 0.001$
$kernel_size = 3 \times 3$
$num_filters = [32, 64]$
$hidden_units = 128$
Beluga optimizer:
population_size N = 20
max_iterations M = 50

Output:
Optimized model parameters θ*

Step 1: Initialize Model
$Conv\ Layer\ 1$: $32\ filters, 3 \times 3, ReLU$
$Conv\ Layer\ 2$: $64\ filters, 3 \times 3, ReLU$
Max Pool: 2×2
LSTM: hidden_units = 128
Dense: 10 outputs (softmax)
Initialize θ ∈ [-0.05, 0.05]
Initialize Beluga optimizer with N = 20 candidate solutions

Step 2: Spatial Convolution
For input xi (64×64 image):
Conv1 → output size 62×62×32
Conv2 → output size 60×60×64
MaxPool → output size 30×30×64
Flatten → feature vector Fi of size 57,600

Step 3: Temporal Recurrence
- Split Fi into 30 segments (each 1920 features)
- Feed segments sequentially into LSTM
- Get final temporal embedding Ti (128-dim)

Step 4: Prediction
- Dense layer: 128 → 10
- Softmax activation
- Prediction ŷi

Step 5: Loss

- Use categorical cross-entropy
Example: true = class 3, ŷ3 = 0.78 → Loss = -log(0.78) = 0.248

Step 6: Beluga Optimization

For iter = 1 to 50:
For each candidate pi (20 total):
Evaluate fitness f(pi) = validation loss
Update candidates using:
- Exploration (random migration, α=0.3)
- Exploitation (move toward p_best, β=0.7)
θ ← p_best

Step 7: Training

For epoch = 1 to 50:
Split dataset into 157 batches of size 64
For each batch:
Apply Steps 2–5
Update θ using Beluga optimizer (Step 6)
Record accuracy and loss per epoch

Step 8: Output

Final optimized θ*
Test set performance (2,000 samples):

3.4.1 Spatial CRN

Spatial Convolve RecurrenceNet integrates spatial convolution and recurrence to analyze environmental patterns and dynamics. It enables real-time evaluation and adaptive design through IoT data analysis, facilitating informed decision-making and optimization in environmental design.

➢ **CRN**

CNNs use convolutional filters and pooling layers to extract spatial data, lowering dimensionality and retaining environmental characteristics such as temperature, CO_2, and illuminance. LSTM networks collect long-term temporal dependencies in sensor data, allowing for precise forecasts of future conditions, improving comfort, sustainability, and overall environmental design assessment.

➢ **Spatial CRN**

The convolutional encoder and attention module, as shown in Figure 6, are used largely to extract spatial-frequency characteristics.

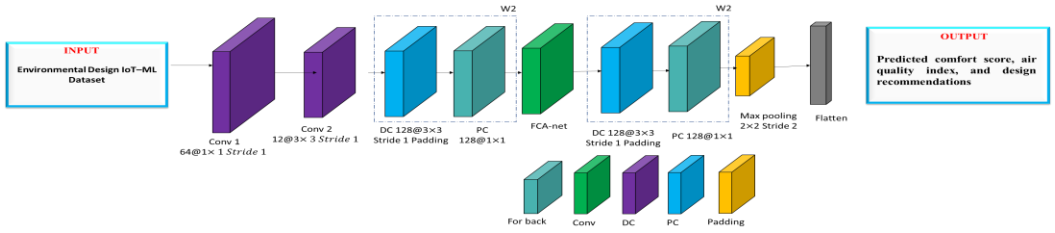


Figure 6: Spatial CRN architecture

The CNN processes 2u frames Sj (9×9×8) with 1×1 and 3×3 convolutions and depthwise-pointwise layers for better feature extraction. The network employs 1x1 and 3x3 convolutions, and also depth wise and pointwise convolutions, to record spatial-frequency information and identify environmental trends for adaptive design. The layer outputs of the S-CRN model are depicted in Figure 7. The model captures spatiotemporal features using Conv2, pooling, and LSTM to make interpretable environmental design predictions. The S-CRN outputs Conv1, Conv2, pooling, LSTM embedding, and dense layers progressively integrate geographical and temporal data, yielding interpretable and actionable insights for environmental optimization.

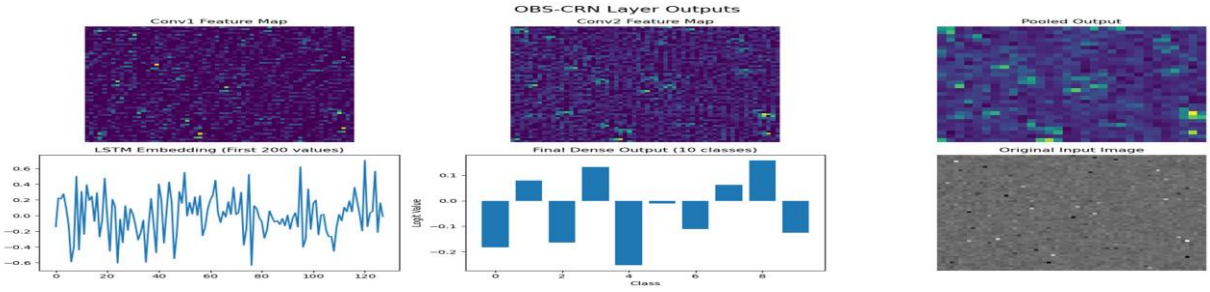


Figure 7: OBS-CRN layer outputs showing feature maps, embeddings, and predictions.

3.4.2 Beluga optimization

Beluga optimization simulates whale foraging behavior, effectively balancing exploration and exploitation to achieve optimal layouts, resource allocation, and precise environmental optimization. The Beluga algorithm adaptively optimizes OBS-CRN hyperparameters to improve convergence, reduce overfitting, and maximize prediction reliability, allowing for robust and precise environmental design evaluation. The balancing factor of equation (7) determines the shift of BWO between exploration and exploitation.

$$A_e = A_p\left(1 - \frac{S}{2} * S_{max}\right) \qquad (7)$$

Where S and S_{max} are the present and maximum repetition numbers, and A_p is a random value between (0,1) at each repetition. When$A_e > 0.5$, the process is in the exploration stage, and when$A_e = 0.5$, the process is in exploitation.

Beluga optimization improves environmental design by optimizing spatial planning, resource allocation, sustainability, and adaptive, data-driven decision-making in dynamic, complex design contexts. To verify consistent replication of findings, the exact random seed values used for network weight setup, data division, and Beluga optimization iterations should be specified. The optimized beluga flowchart is demonstrated in Figure 8.

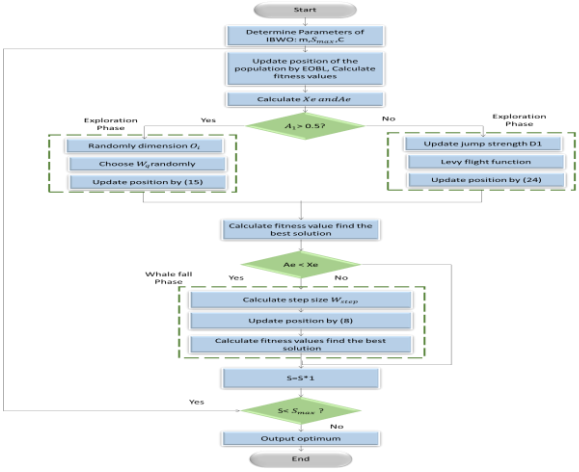


Figure 8: Flowchart for Beluga optimization

➢ **Elite opposition-based learning (EOBL)**

EOBL creates a reverse population, merges it with the existing population, and selects the finest people, increasing variety and search space to avoid local optima. This could be expressed as in equation (8).

$$\overline{W_{j,i}^f} = L^*\left(\alpha_i + \beta_i\right) - W_{j,i}^f \qquad (8)$$

Where $\overline{W_{j,i}^f}$ is the elite individual, and $W_{j,i}^f$ is the opposition-based counterpart used to expand the search. L^* : Dynamic coefficient $(0-1)$ controlling exploration. $\alpha_i$and β_i Are the minimum and maximum bounds of the ith dimension, defining the search space limits. EOBL enhances optimization by creating reverse populations, increasing variety, avoiding local optima, and improving spatial layout, resource allocation, and long-term environmental planning.

➢ **Balance factor**

The size of A_e plays a crucial role in balancing global and local searching abilities during the BWO exploration and exploitation phases. Iterative alterations of A_e enhance the exploration and development stages in equation (9).

$$A_e = A_{emin} + (A_{emin} - A_{emax}) \cdot \exp\left(\ln\left(\frac{A_{emin}}{A_{emax}}\right) \cdot \frac{S}{S_{max}}\right) \quad (9)$$

Where A_e is the balance factor regulating exploration and exploitation; A_{emax} and A_{emin} are its maximum and minimum limits; S is the current iteration; and S_{max} is the total maximum iterations.

The balance factor $A_e$decays exponentially, allowing for early global exploration, later local exploitation, and higher optimization precision.

- **Cyclone foraging strategy (CFS)**

CFS broadens the search space by employing rotational foraging and spiral search strategies, hence increasing exploitation in modified BWO, accelerating convergence, preventing stagnation, and improving solution accuracy for complicated spatiotemporal environmental design issues. By emulating this dynamic, spiral-oriented search behavior, the technique combines local refinement with larger exploration, enhancing overall optimization and solution accuracy in complicated problem environments (equation 10).

$$W_i^{S+1} = \begin{cases} W_{best}^S + q_8\left(W_{best}^S - W_j^S\right) \\ +\beta \cdot D_1 \cdot K_E\left(W_{best}^S - W_j^S\right), j = 1 \\ W_{best}^S + q_8\left(W_{j-1}^S - W_j^S\right) \\ +\beta \cdot D_1 \cdot K_E\left(W_{best}^S - W_j^S\right), j = 1, \ldots, m \end{cases} \quad (10)$$

Where W_j^S is the current position of the jth individual, and W_i^{S+1} is its updated position; W_{best}^S represents the best solution found so far; q_8 is a random number in [0,1] that introduces stochastic behavior; β is a dynamic weight controlling exploration and exploitation; D_1 is a control coefficient regulating step size; $K_E(.)$ is the Lévy flight operator that enables long random jumps for global exploration; and m is the total number of individuals in the population (equation 11).

$$\beta = 2f^{q_9^{\frac{S-s+1}{S}}} \cdot \sin(2\pi q_9) \quad (11)$$

Where β is the dynamic weight factor controlling exploration and exploitation; q9 is a random number in[0,1]; s is the current iteration; S is the maximum number of iterations; and $\sin(2\pi q_9)$ is an oscillatory component that introduces diversity.

The major hyper parameters for an OBS-CRN are demonstrated in Table 3shows the details the dataset, OBS-CRN architecture, hyperparameters including convolutional, pooling, and LSTM layers and training settings, emphasizing reliable, scalable, and adaptable environmental evaluations for smart building applications.

Table 3: Hyper parameters for OBS-CRN

Category	Hyper parameter	Value
Dataset	*Number of samples*	6,480(*train*), 2,000 (*test*)
	Input dimension	64 × 64 (*images*)
	Output classes	10
Convolution	*Conv*1 *filters*	32
	*Conv*1 *kernel size*	3 × 3
	*Conv*2 *filters*	64
	*Conv*2 *kernel size*	3 × 3
	Pooling	*MaxPool* 2 × 2
Recurrent	*Sequence length*	30
	Hidden units	128
Training	*Learning rate*	0.001

	Epochs	50
	Batch size	64
Beluga Optimizer	*Population size* (N)	20
	Max iterations (M)	50
	Exploration factor (α)	0.3
	Exploitation factor (β)	0.7
Output Layer	*Dense units*	10

The OBS-CRN hybrid solution increases spectrum utilization and network throughput, minimizes interference, and provides reliable communication in dynamic environments, exceeding traditional CRN methods in efficiency, latency, and overall spectral performance.

- **Statistical analysis**

To verify performance improvements for the OBS-CRN model, a paired Student's t-test was conducted to compare it against existing models, focusing on RMSE, R^2, MSE, and MAE. A significance level of 0.05 was set, ensuring that any observed increase in predictive accuracy is statistically significant, indicating a meaningful enhancement in environmental design evaluation and IoT-enabled real-time monitoring.

4 Results and discussion

Python is used to evaluate the proposed method, while several existing approaches have been considered for comparison, including Gated Recurrent Unit (GRU) [23], LSTM [23], Recurrent Neural Network (RNN) [23], CNN [23], and a hybrid model combining Support Vector Regressor (SVR), Light Gradient Boosting Machine (LGBM), and Ridge Regression (RIDGE) (SVR+LGBM+Ridge) [24], Bidirectional Long Short-Term Memory (Bi-LSTM) [25], Attention-based Long Short-Term Memory (Attention-LSTM) [25], Convolutional Neural Network – Long Short-Term Memory (CNN-LSTM) [25],and Improved Hybrid Hierarchical Online DL – Environmental Comfort Prediction (IHHODL-ECP) [25] for environmental design.

4.1 Experimental step-up

The OBS-CRN system was built in Python (TensorFlow), with 16 GB of RAM, 1 TB of storage, and preprocessing using z-score normalization and PCA for efficient, scalable environmental design evaluation.

4.2 Experimental result

OBS-CRN monitoring Figure 9a indicates steady temperature, CO_2, and $PM_{2.5}$ levels with changing humidity. Figure 9b illustrates how CO_2 and $PM_{2.5}$ levels vary, guiding ventilation, pollution management, and illumination for better indoor environment.

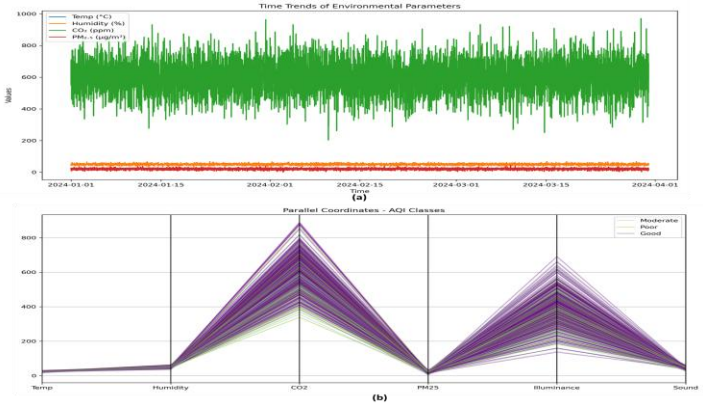


Figure 9: Environmental designs of (a) Time trends of temperature, humidity, CO_2, and $PM_{2.5}$ Levels, (b) AQI classes across key paramet.

OBS-CRN Figure 10 depicts comfort score changes and identifies discomfort patterns in order to optimize temperature, air quality, and energy-efficient, data-driven environmental design methods.

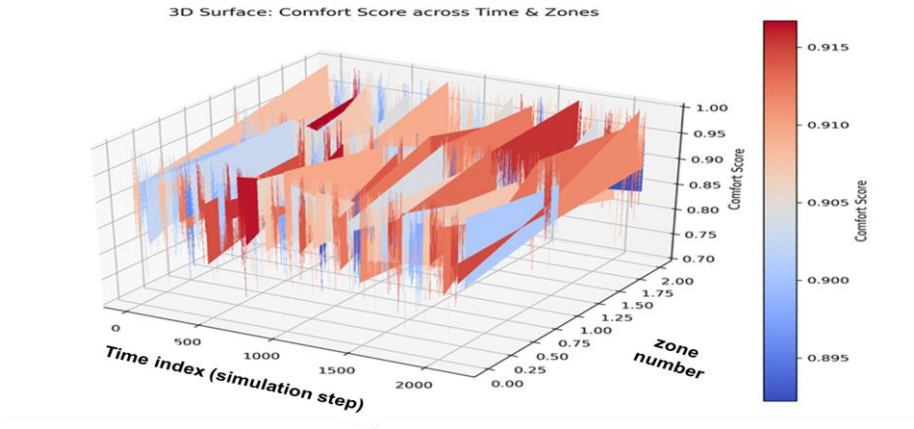


Figure 10: Comfort scores over time and zones for environmental design

Time-series analysis Figure 11 reveals steady temperature and CO_2 levels, with changing humidity and sound, leading ventilation, acoustics, and climate modifications for energy-efficient, comfortable, and healthy indoor environments.

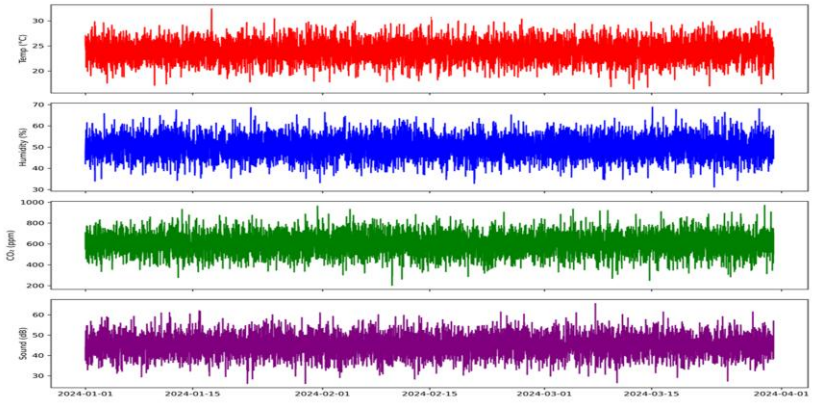


Figure 11: Time series plots displaying key environmental parameters over month

4.3 Performance metrics

The Mean Squared Error (MSE) is a measure of the average difference between predicted and actual values, used to ensure model accuracy in environmental parameters. The Root Mean Squared Error (RMSE) is the square root of the MSE, used to assess precision in predictions. The Mean Absolute Error (MAE) is the average of absolute differences between predicted and actual values, used to evaluate prediction accuracy in temperature, air quality, or comfort assessments. The coefficient of determination (R^2) is the percentage of variance attributed to a model, suggesting better fit for spatiotemporal patterns in the environment.

4.4 Comparison of proposed techniques with existing techniques

The performance comparison of several models for environmental design evaluation reveals significant differences in performance, as shown in Table 4 and Figure 12(a-b) The proposed OBS-CRN model outperformed existing methods (RNN, GRU, LSTM, CNN, SVR+LGBM+RIDGE, Bi-LSTM, Attention LSTM CNN-LSTM, and IHHODL-ECP [23-25]) with an RMSE of 0.25 and R^2 of 0.990 [23-25]. These findings demonstrate its suitability for real-time environmental design evaluation with IoT-enabled data.

Table 4: Comparison of performance for the proposed and the existing techniques in environmental design

Methods	**RMSE**	R^2
RNN [23]	0.48	0.69
GRU [23]	0.36	0.97
LSTM [23]	0.39	0.93
CNN [23]	0.47	0.71
SVR+LGBM + RIDGE [24]	2.13	0.82
Bi-LSTM [25]	0.7277	-
Attention LSTM [25]	0.8902	-
CNN-LSTM [25]	0.6089	-
IHHODL-ECP [25]	0.5602	-
OBS-CRN [Proposed]	**0.25**	**0.990**

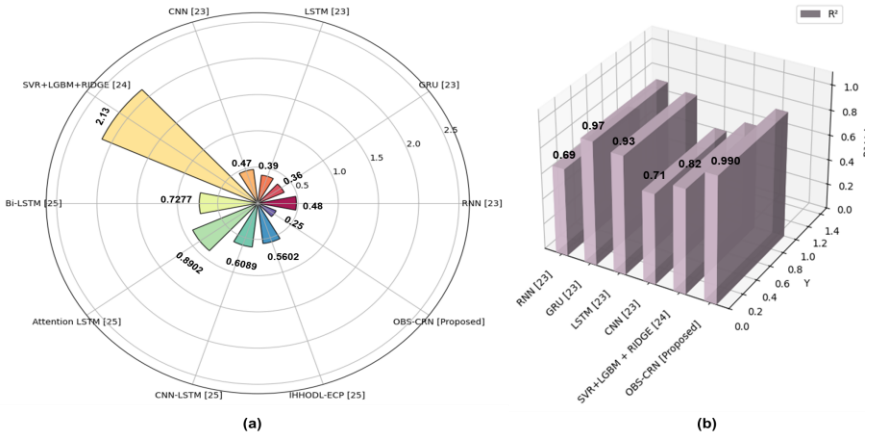


Figure 12(a-b): Comparative performance for the proposed and the existing methods in environmental designs

The performance of various models for environmental design evaluation reveals that classic DL approaches provide modest performance, as depicted in Figure 13 and Table 5 shows a performance evaluation of various models for environmental design prediction. The suggested OBS-CRN model has the lowest MSE (0.10) and MAE (0.25), indicating improved prediction accuracy and resilience when compared to existing methods such as RNN, GRU, LSTM, CNN, SVR+LGBM+RIDGE, Bi-LSTM, Attention LSTM CNN-LSTM, and IHHODL-ECP [23-25].

Table 5: Performance comparison for the existing and the suggested techniques in environmental designs

Methods	**MSE**	**MAE**
RNN [23]	0.23	0.39
GRU [23]	0.13	0.29
LSTM [23]	0.15	0.25
CNN [23]	0.22	0.38
SVR+LGBM + RIDGE [24]	4.53	1.20

Bi-LSTM [25]	0.5295	0.5212
Attention LSTM [25]	0.7924	0.5461
CNN-LSTM [25]	0.3708	0.3733
IHHODL-ECP [25]	0.3146	0.3359
OBS-CRN [Proposed]	**0.10**	**0.25**

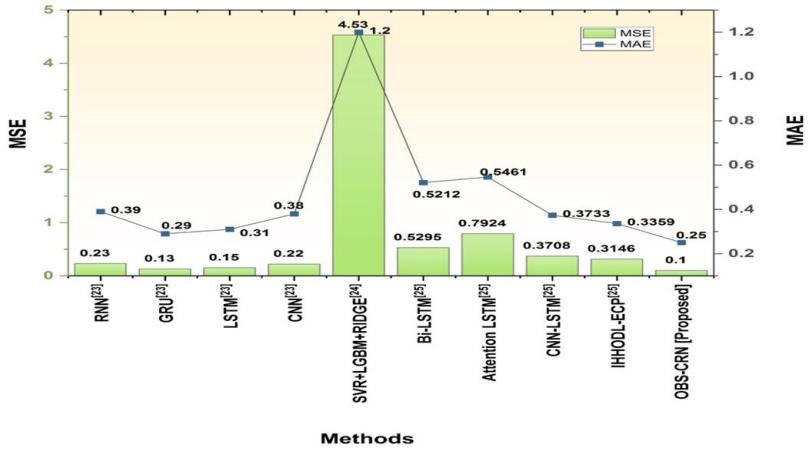


Figure 13: Comparative performance for the proposed and the existing techniques in environmental designs

Table 6 presents the comparison of MAPE for environmental design prediction models reveals that the proposed OBS-CRN achieves the lowest MAPE of 25.40%, outperforming Bi-LSTM, Attention LSTM, CNN-LSTM, and IHHODL-ECP.

Table 6: The MAPE values for proposed methods vs existing methods

Method	**MAPE (%)**
Bi-LSTM [25]	50.25
Attention LSTM [25]	48.50
CNN-LSTM [25]	34.77
IHHODL-ECP [25]	34.13
OBS-CRN [Proposed]	25.40

Table 7 display the statistical t-test results comparing the performance of existing DL and hybrid models to the proposed OBS-CRN, including mean differences, t-values, p-values, and significance, support OBS-CRN's better predictive accuracy and robust environmental design evaluation.

Table 7: The statistical t-test analysis for proposed methods.

OBS-CRN [Proposed]			
Metric	**Mean Difference**	**t-value**	**p-value**
RMSE	0.10	4.90	0.004
MSE	0.012	4.85	0.004
MAE	0.08	4.75	0.005
MAPE	0.055	4.60	0.006
R^2	0.990	6.50	0.001

4.5 Discussion

The OBS-CRN model outperforms baselines in RMSE, MSE, MAE, MAPE, and R^2. It uses convolutional layers with Beluga optimization to address spatial dependencies and convergence. Existing approaches, such as RNN, GRU, LSTM, CNN [23], hybrid SVR+LGBM+Ridge [24], and Bi-LSTM, Attention LSTM, CNN-LSTM, and IHHODL-ECP [25], achieve moderate accuracy but have limits in scalability, robustness, noise management, and real-time adaptation, lowering their effectiveness in complicated settings. Future work should focus on improving generalization, adaptive control, and real-time responsiveness for flexible, resilient, and scalable smart building applications. Existing technology lacks real-time optimization, which reduces comfort and energy efficiency. Future research should use adaptive or neural adaptive control to dynamically change HVAC, lighting, and ventilation, therefore increasing responsiveness, efficiency, and overall smart-building performance. The proposed structure should be evaluated in a variety of building types residential, educational, and commercial to confirm OBS-CRN's resilience, scalability, and adaptability, revealing structural dependencies and restrictions for genuine, real-world smart building and city design applications in further research. The OBS-CRN architecture, which has been demonstrated accurate in a single office, should be tested across many building types residential, industrial, and mixed-use to assure stability, generalization, and practical integration with BMS for adaptive ventilation and energy optimization in smart buildings in further research. An RMSE of 0.25 suggests highly accurate comfort prediction, allowing for dependable real-time suggestions; nonetheless, performance is dependent on clean, dense IoT sensor data. In the future, Adaptive control techniques like fuzzy or backstepping control can enhance the OBS-CRN architecture by allowing it to automatically adjust to varying environmental conditions, improving flexibility, reliability, and resilience for smart-building applications. The accuracy in one workplace but lacks scalability analysis. Future research will investigate various sensor density, building types, and transfer learning to improve OBS-CRN robustness and implementation in smart buildings.

5 Conclusion

Environmental design is essential for creating functional, comfortable, and sustainable spaces. The integration of IoT technology facilitates real-time environmental data collection, while machine learning supports intelligent analysis. IoT sensor networks in office buildings monitor factors. Data preprocessing addresses missing values with z-score normalization. Feature extraction, with PCA used for dimensionality reduction. The document presents an IoT-ML assessment framework based on the OBS-CRN, an enhanced DL model that captures spatial correlations and temporal patterns in the data. The suggested technique achieves better performance when compared to the existing techniques, in terms of MAE (0.25), RMSE (0.25), MSE (0.10), and R^2 (0.990). The proposed model was extremely accurate, but it faces several challenges: sensor failures could diminish dependability, IoT implementation is expensive, occupancy tracking raises privacy concerns, and noisy data can reduce performance. Future work should address sensor dependency, privacy, and data concerns by combining Dl, edge computing, and explainable AI (SHAP, LIME) to improve fault tolerance, cost-efficiency, noise handling, and transparency, allowing for more dependable smart-city environmental solutions.

Competing interests

The authors have no competing interests to declare that are relevant to the content of this article.

Data availibility statement

The data can be obtained from the corresponding author.

Author contributions

Xing. Data curation&Writing—original draft preparation&Experiment operation and execution

References

[1] Butt, A.N. and Dimitrijević, B., 2022. Multidisciplinary and transdisciplinary collaboration in nature-based design of sustainable architecture and urbanism. Sustainability, 14(16), p.10339. https://doi.org/10.3390/su141610339

[2] Marzouk, M. and Atef, M., 2022. Assessment of indoor air quality in academic buildings using IoT and deep learning. Sustainability, 14(12), p.7015. https://doi.org/10.3390/su14127015

[3] Gabriel, M. and Auer, T., 2023. LSTM deep learning models for virtual sensing of indoor air pollutants: a feasible alternative to physical sensors. Buildings, 13(7), p.1684. https://doi.org/10.3390/buildings13071684

[4] Morresi, N., Casaccia, S., Sorcinelli, M., Arnesano, M., Uriarte, A., Torrens-Galdiz, J.I. and Revel, G.M., 2021. Sensing physiological and environmental quantities to measure human thermal comfort through machine learning techniques. IEEE Sensors Journal, 21(10), pp.12322-12337. https://doi.org/10.1109/JSEN.2021.3064707

[5] Jain, H., 2024. Critical insights into thermal comfort optimization and heat resilience in indoor spaces. City and Built Environment, 2(1), p.14. https://doi.org/10.1007/s44213-024-00038-z

[6] Iram, S., Farid, H.M.A., Akande, A.A. and Shakeel, H.M., 2025. Intelligent Modelling Techniques for Enhanced Thermal Comfort and Energy Optimisation in Residential Buildings. Energies, 18(14), p.3878. https://doi.org/10.3390/en18143878

[7] Tabuenca, B., Moreno-Sancho, J.L., Arquero-Gallego, J., Greller, W. and Hernández-Leo, D., 2023. Generating an environmental awareness system for learning using IoT technology. Internet of things, 22, p.100756. https://doi.org/10.1016/j.iot.2023.100756

[8] Zhu, L., Husny, Z.J.B.M., Samsudin, N.A., Xu, H. and Han, C., 2023. Deep learning method for minimizing water pollution and air pollution in urban environment. Urban Climate, 49, p.101486. https://doi.org/10.1016/j.uclim.2023.101486

[9] Das, H.P., Lin, Y.W., Agwan, U., Spangher, L., Devonport, A., Yang, Y., Drgoňa, J., Chong, A., Schiavon, S. and Spanos, C.J., 2024. Machine learning for smart and energy-efficient buildings. Environmental Data Science, 3, p.e1. https://doi.org/10.1017/eds.2023.43

[10] Du, B., 2024. Indoor Environment Design and Energy Saving Analysis of Sustainable Development Integrated with Green Building. RE&PQJ, pp.29-38. https://doi.org/10.52152/3987

[11] Lee, J., Woo, D.O., Jang, J., Junghans, L. and Leigh, S.B., 2022. Collection and utilization of indoor environmental quality information using affordable image sensing technology. Energies, 15(3), p.921. https://doi.org/10.3390/en15030921

[12] Kujundzic, K., Stamatovic Vuckovic, S. and Radivojević, A., 2023. Toward regenerative sustainability: A passive design comfort assessment method of indoor environment. Sustainability, 15(1), p.840. https://doi.org/10.3390/su15010840

[13] Chen, X., 2023. Environmental landscape design and planning system based on computer vision and deep learning. Journal of Intelligent Systems, 32(1), p.20220092. https://doi.org/10.1515/jisys-2022-0092

[14] Li, P., Xu, T., Wei, S. and Wang, Z.H., 2022. Multi-objective optimization of urban environmental system design using machine learning. Computers, Environment and Urban Systems, 94, p.101796. https://doi.org/10.1016/j.compenvurbsys.2022.101796

[15] Wei, S., Tien, P.W., Chow, T.W., Wu, Y. and Calautit, J.K., 2022. Deep learning and computer vision-based occupancy CO2 level prediction for demand-controlled ventilation (DCV). Journal of Building Engineering, 56, p.104715. https://doi.org/10.1016/j.jobe.2022.104715

[16] Popescu, S.M., Mansoor, S., Wani, O.A., Kumar, S.S., Sharma, V., Sharma, A., Arya, V.M., Kirkham, M.B., Hou, D., Bolan, N. and Chung, Y.S., 2024. Artificial intelligence and IoT driven technologies for environmental pollution monitoring and management. Frontiers in Environmental Science, 12, p.1336088. https://doi.org/10.3389/fenvs.2024.1336088

[17] Heydari, A., Majidi Nezhad, M., Astiaso Garcia, D., Keynia, F. and De Santoli, L., 2022. Air pollution forecasting application based on deep learning model and optimization algorithm. Clean Technologies and Environmental Policy, 24(2), pp.607-621. https://doi.org/10.1007/s10098-021-02080-5

[18] Ullah, A., Anwar, S.M., Li, J., Nadeem, L., Mahmood, T., Rehman, A. and Saba, T., 2024. Smart cities: The role of Internet of Things and machine learning in realizing a data-centric smart environment. Complex & Intelligent Systems, 10(1), pp.1607-1637.https://doi.org/10.1007/s40747-023-01175-4

[19] Elsisi, M., Tran, M.Q., Mahmoud, K., Lehtonen, M. and Darwish, M.M., 2021. Deep learning-based industry 4.0 and internet of things towards effective energy management for smart buildings. Sensors, 21(4), p.1038.https://doi.org/10.3390/s21041038

[20] Floris, A., Porcu, S., Girau, R. and Atzori, L., 2021. An iot-based smart building solution for indoor environment management and occupants' prediction. Energies, 14(10), p.2959.https://doi.org/10.3390/en14102959

[21] Chennapragada, A., Periyakoil, D., Das, H.P. and Spanos, C.J., 2022, June. Time series-based deep learning model for personal thermal comfort prediction.

In Proceedings of the thirteenth ACM international conference on future energy systems (pp. 552-555). https://doi.org/10.1145/3538637.3539617

[22] Shah, S.F.A., Iqbal, M., Aziz, Z., Rana, T.A., Khalid, A., Cheah, Y.N. and Arif, M., 2022. The role of machine learning and the internet of things in smart buildings for energy efficiency. Applied Sciences, 12(15), p.7882.https://doi.org/10.3390/app12157882

[23] Godasiaei, S.H., Ejohwomu, O.A., Zhong, H. and Booker, D., 2025. Integrating experimental analysis and machine learning for enhancing energy efficiency and indoor air quality in educational buildings. Building and Environment, 276, p.112874. https://doi.org/10.1016/j.buildenv.2025.112874

[24] Amangeldy, B., Tasmurzayev, N., Imankulov, T., Imanbek, B., Wójcik, W. and Nurakhov, Y., 2025. A Hybrid Machine Learning Approach for High-Accuracy Energy Consumption Prediction Using Indoor Environmental Quality Sensors. Energies, 18(15), p.4164. https://doi.org/10.3390/en18154164

[25] Balaji, S. and Karthik, S., 2023. Deep Learning Based Energy Consumption Prediction on Internet of Things Environment. Intelligent Automation & Soft Computing, 37(1). http://dx.doi.org/10.32604/iasc.2023.037409

SwinEff-DR: Hybrid Swin Transformer & Efficient Net Architecture for Multi-Scale Diabetic Retinopathy Detection

Anju Mishra[1], Mrinal Pandey[1], Laxman Singh[2]
Manav Rachna University, Faridabad
KIET Group of Institutions, Ghaziabad
E-mail: aanjumishra2108@gmail.com, mrinalpandey@mru.edu.in, laxman.mehlawat01@gmail.com

Keywords: diabetic retinopathy, deep learning, convolutional neural networks (CNNs), EfficientNet, swin transformer, medical image classification, SwinEff-DR

Received: September 21, 2025

Diabetic Retinopathy (DR) remains one of the leading causes of preventable blindness worldwide, underscoring the need for early detection and accurate classification. The root cause of this disease is diabetes mellitus. According to the WHO, about 537 million people suffer from the disease and are expected to increase to 783 million by 2047. Diabetic retinopathy (DR) remains an incurable condition; however, early detection can significantly restrict the progression of vision loss. Routine ophthalmic examinations and continuous monitoring play a critical role in preventing blindness associated with DR. Consequently, a pressing need exists for advanced computer-assisted diagnostic systems capable of accurately detecting and grading DR, providing valuable support to ophthalmologists for timely intervention. To address this, we propose SwinEff-DR, a hybrid architecture of Swin transformers with Efficient Network to achieve robust and precise DR classification. The advanced preprocessing on the EyePACS dataset will be performed, then the Swin Transformer as a backbone of the model, followed by Efficient Net. The SwinEff-DR model attains 0.96 precision, 0.97 recall, 0.97 accuracy, and 0.97 F1-score, achieving a 1.49% improvement over existing methods. Furthermore, the framework aligns predictions with standardised severity grading, enabling robust and clinically meaningful diagnostic support.

Povzetek: Prispevek predstavlja model SwinEff-DR za zgodnje zaznavanje diabetične retinopatije, ki z visoko natančnostjo izboljšuje klinično diagnostiko.

1 Introduction

Diabetic Retinopathy (DR) is one of the leading causes of preventable blindness worldwide, primarily affecting individuals with long-term diabetes. According to the World Health Organisation, the prevalence of DR continues to rise in parallel with the global increase in diabetes, making its timely and accurate diagnosis a critical healthcare priority [1]. Early detection of DR, particularly in its initial stages marked by microaneurysms and subtle haemorrhages, is vital in preventing vision loss [2]. However, manual grading of fundus images is time-consuming, subjective, and requires trained ophthalmologists, underscoring the need for reliable computer-assisted diagnostic systems [3].

Deep learning, especially Convolutional Neural Networks (CNNs), has demonstrated significant potential in automating DR classification tasks [4], [5]. Despite their effectiveness, CNN-based approaches face several challenges. First, DR lesions occupy less than 1% of fundus pixels, biasing models toward background regions and causing under-detection of tiny yet critical features [6]. Second, class imbalance, particularly the scarcity of severe and proliferative DR cases, further compromises model generalizability [7]. Third, image quality variations, from inconsistent illumination to blur and device-specific differences, hinder accurate lesion identification [8]. Finally, relying on a single CNN architecture often limits robustness, as individual models may overfit specific datasets or fail to generalise across diverse imaging conditions [9].

To address these limitations, a hybrid model has emerged as a promising strategy, leveraging the strengths of two models while reducing individual weaknesses [10].

This study proposes a framework, "SwinEff-DR: Hybrid Swin Transformer & Efficient Net Architecture for Multi-

Scale Diabetic Retinopathy Detection." The framework integrates preprocessing and augmentation to maintain the class imbalance of the dataset. Swin transformer followed by Efficient Net is used to enhance both sensitivity and robustness. The preprocessing techniques (resize) emphasise image fixed-sized management, so all input images must be the same size. Furthermore, data augmentation strategies are employed to address class imbalance and improve generalisation across heterogeneous fundus datasets.

The contributions of this work are summarised as follows:

1. Data Augmentation uses preprocessing pipelines that enhance lesion visibility and provide better feature extraction.
2. A robust swin transformer mechanism that builds hierarchical features, like CNN models and captures both local & global patterns efficiently.
3. Then, an efficient Net architecture is applied as a hybrid-model classifier.
4. The proposed method is evaluated against existing state-of-the-art approaches and demonstrates superior accuracy, precision, recall, and F1-score performance.

By integrating lesion-focused preprocessing with ensemble learning, this framework aims to advance the development of clinically reliable and computationally efficient diagnostic tools for early and accurate DR detection.

2 Related work

The studies span a range of methods and datasets in diabetic retinopathy (DR) detection. Ghosh & Chatterjee [11] employed a transfer-learning ensemble of pretrained CNNs (VGG16 and Inception V3), trained on benchmark datasets APTOS, IDRiD, and Messidor-2. Their model achieved 96.4% accuracy, though recall was not evaluated, and lesion-level validation was limited to SCIRParXiv. Bajwa et al. [12] used a modified CNN trained and tested on a private, clinician-labelled fundus-image set collected at the Sindh Institute of Ophthalmology & Visual Sciences (SIOVS)—about 398 patients—and achieved 93.72% accuracy and 97.3% sensitivity. However, model performance varied across datasets and relied heavily on preprocessing MDPIMDPI. For Khan A. et al. [13], details on the specific methodological approach or dataset were not in our search. However, their accuracy was 92%, with unreported precision/recall and a noted issue of dataset imbalance affecting rare DR stages. Hacisoftaoglu et al. [14] used smartphone-acquired images and achieved 98.6% accuracy and 98.2% recall; however, the model's universality remains questionable due to its reliance on smartphone imagery (dataset details unspecified). Khairandish et al. [15] achieved a moderate accuracy of 82% with complete precision and a recall of 81% using a more complex method than a CNN-only model. However, the specifics of the approach and dataset remain unclear. Finally, Jadhav et al. [16] focused solely on ROC-based evaluation with 80.85% accuracy and 76.75% recall, but did not report precision or dataset details, and lacked clinical interpretability. Rajmani et al. [17] used an applied generic deep learning model and optimised hyperparameter tuning using GSCV, achieving 89% accuracy. Zhang. Q. et. al. [18] followed the methodology wherein they combined a preprocessing and segmentation approach before applying Cuckoo search for feature optimisation and obtained commendable results with accuracy metrics surpassing 97.55%.. P.S. Silva et al. [19] proposed a multi-stage deep learning framework for DR grading, integrating lesion segmentation and classification. The model achieved promising results in terms of accuracy. The results obtained on a large DR dataset highlight the potential of deep learning in accurately diagnosing DR. C. Mohanty et al. [20] developed a hybrid model by combining the VGG16 architecture with an XGBoost classifier and further utilising DenseNet-121, achieving an accuracy of 79.50%.

W. L. Alyoubi et al. [21] employed a CNN for DR classification in combination with YOLOv3 for lesion localisation, achieving an overall accuracy of 89%. M.K. Yaqoob et al. [22] employed ResNet-50 for feature extraction in combination with a Random Forest classifier, achieving an accuracy of 75.09%. In contrast, G. Zhang et al. [23] proposed a Multi-Model Domain Adaptation (MMDA) approach with transfer learning, incorporating weighted strategies to enhance performance, pseudo labelling and clustering-based

approaches and achieved 90.6% accuracy on the APTOS dataset.

Table 1 shows the comparative literature review for diabetic retinopathy detection. The key contributions of this work can be summarised as follows. First, a carefully designed preprocessing and data augmentation pipeline is employed to enhance lesion visibility and improve feature extraction. Second, a robust Swin Transformer mechanism is integrated to construct hierarchical representations similar to CNNs while efficiently capturing both local and global retinal patterns. Third, EfficientNet is incorporated within a hybrid classification framework, complementing the transformer features with high-resolution lesion details. Finally, the proposed method is rigorously compared with existing state-of-the-art approaches, demonstrating superior accuracy, precision, recall, and F1-score performance. By combining lesion-focused preprocessing with ensemble-based learning, the framework contributes toward developing clinically reliable and computationally efficient diagnostic tools for early and accurate detection of diabetic retinopathy.

Table 1: Literature review

Study / Model (Year)	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**Approach Used**	**Dataset Used**	**Limitations**
Ghosh & Chatterjee (2023) [11]	96.4	–	–	Transfer learning with an ensemble of VGG16 + InceptionV3	APTOS, IDRiD, Messidor-2	Did not evaluate recall; limited lesion-level validation
Bajwa et al. (2023) [12]	93.72	–	97.3	Customized CNN	Private dataset (SIOVS, Pakistan)	Inconsistent performance across datasets; preprocessing dependency
Khan et al. (2024) [13]	92	–	–	Deep learning (details unspecified)	Dataset unspecified (imbalanced)	Dataset imbalance; poor generalisation on rare DR stages
Hacisoftaoglu et al. (2020) [14]	98.6	–	98.2	Smartphone-based image classification with CNN	Smartphone-acquired retinal images	Limited to smartphone images; lacks universality
Khairandish et al. (2022) [15]	82.0	81.0	82.0	Hybrid model (more complex than CNN-only)	Dataset not specified	Lower accuracy than CNN-only; high training complexity
Jadhav et al. (2025) [16]	80.85	–	76.75	ROC-based analysis (deep learning)	Dataset unspecified	Focuses only on ROC metric; ignores clinical interpretability

3 Proposed methodology

The proposed methodology for diabetic retinopathy detection begins with collecting a fundus image dataset, which is then preprocessed to enhance quality and augmented through techniques like gray scale conversion, rotation, flip and zoom to increase data diversity. Resize brings all the heterogeneous images to the fixed size and maintains the input on the same level. The hybrid model SwinEff-DR extracts the features locally and globally. Fusion of outputs performed, then finally, the system classifies retinal images into five categories, viz., No DR, Mild DR, Moderate DR, Severe NPDR, and PDR, delivering an automated, accurate, and reliable framework for early detection and standardised grading of diabetic retinopathy. Figure 1 shows the flow of methodology.

1 **Fundus image dataset:** The research starts with collecting fundus images from the publicly available dataset Kaggle for diabetic retinopathy detection.
2 **Data preprocessing and augmentation:** The retinal images undergo preprocessing to enhance clarity and maintain consistency before training. This involves resizing images to a standardised resolution, reducing noise, and improving contrast of retinal features for better visibility using CLAHE. Furthermore, data augmentation techniques such as grayscale conversion, image resizing, rotation, and flipping are applied to expand the dataset artificially. Sampling techniques are used to maintain the class imbalance. As images for the NO DR stage are huge in number, the down-sampling technique is needed. Images for the 5th stage PDR are available in very few numbers, so they need up-sampling. Up sampling is performed by augmentation.
3 **Hybrid model architecture:** Feature extraction performed in two stages: (1) Local feature extraction, and (2) Global feature extraction. Efficient Net branch extracts the local features and swin transformer branch extracts the global features from the retinal images. The Efficient Net and Swin Transformers outputs are then concatenated using fully connected layers. Both model branches contribute unique strengths in feature extraction and pattern recognition, and their combined predictions minimise misclassification. This ensemble approach significantly improves robustness, accuracy, and reliability compared to individual models.
4 **Classification:** In the final stage, the system assigns each fundus image to one of five categories reflecting the severity of diabetic retinopathy: No DR, Mild DR, Moderate DR, Severe NPDR, or PDR. SwinEff-DR models' classification supports ophthalmologists by clearly indicating disease progression and guiding timely treatment decisions.

Algorithm: Hybrid Swin Transformer–EfficientNet (SwinEff-DR)

Input: Fundus image dataset $D=\{x_i,y_i\}_{i=1}^{N}$, where x_i is an image and $y_i \in \{0,1,2,3,4\}$ represents DR stage.
Output: Predicted class probabilities for 5 DR stages

Step 1: Data Preprocessing
{
(i) Resize all images to 512×512.
(ii) Apply CLAHE for contrast enhancement.
(iii) Apply augmentation: resize, rotation, and flipping.
}

Step 2: EfficientNet Branch (Local Features)
{

(i) Input x_i into the EfficientNet-B4 backbone.
(ii) Apply:
- Stem Convolution + BatchNorm
- MBConv blocks with depthwise separable convolutions
- Squeeze-and-Excitation layers

(iii) Output: Local lesion feature map $F_{eff}(x_i)$

}

Step 3: Swin Transformer Branch (Global Features)
{
(i) Partition the image into non-overlapping patches (4×4).
(ii) Linear embedding of patches → input tokens.
(iii) For each stage:
- Apply Window-based Multi-Head Self-Attention (W-MSA).
- Apply Shifted Window Attention (SW-MSA) for cross-window learning.
- Downsample to build a hierarchical representation.

(iv) Output: Global context feature map $F_{swin}(x_i)$.
}

Step 4: Feature Fusion
{
(i) Concatenate $F_{eff}(x_i)$ and $F_{swin}(x_i)$.
(ii) Pass through dense layers.
(iii) Apply attention-based fusion for weighted feature importance.
(iv) Apply Batch Normalisation + Dropout for regularisation.
(v) Output: Fused feature vector $F_{fusion}(x_i)$.
}

Step 5: Classification Head
{
(i) Feed $F_{fusion}(x_i)$ into fully connected layers.
(ii) Apply **Softmax** activation.
(iii) Output probability distribution across 5 DR stages:

$$P(y/x_i) = Softmax(W. F_{fusion}(x_i)+b)$$

}

Step 6: Training
{
(i) Define loss function: Weighted categorical cross-entropy.
(ii) Use optimiser: AdamW with learning rate scheduling.
(iii) Train for E epochs with batch size B.
(iv) Apply early stopping if validation loss does not improve.
}
End

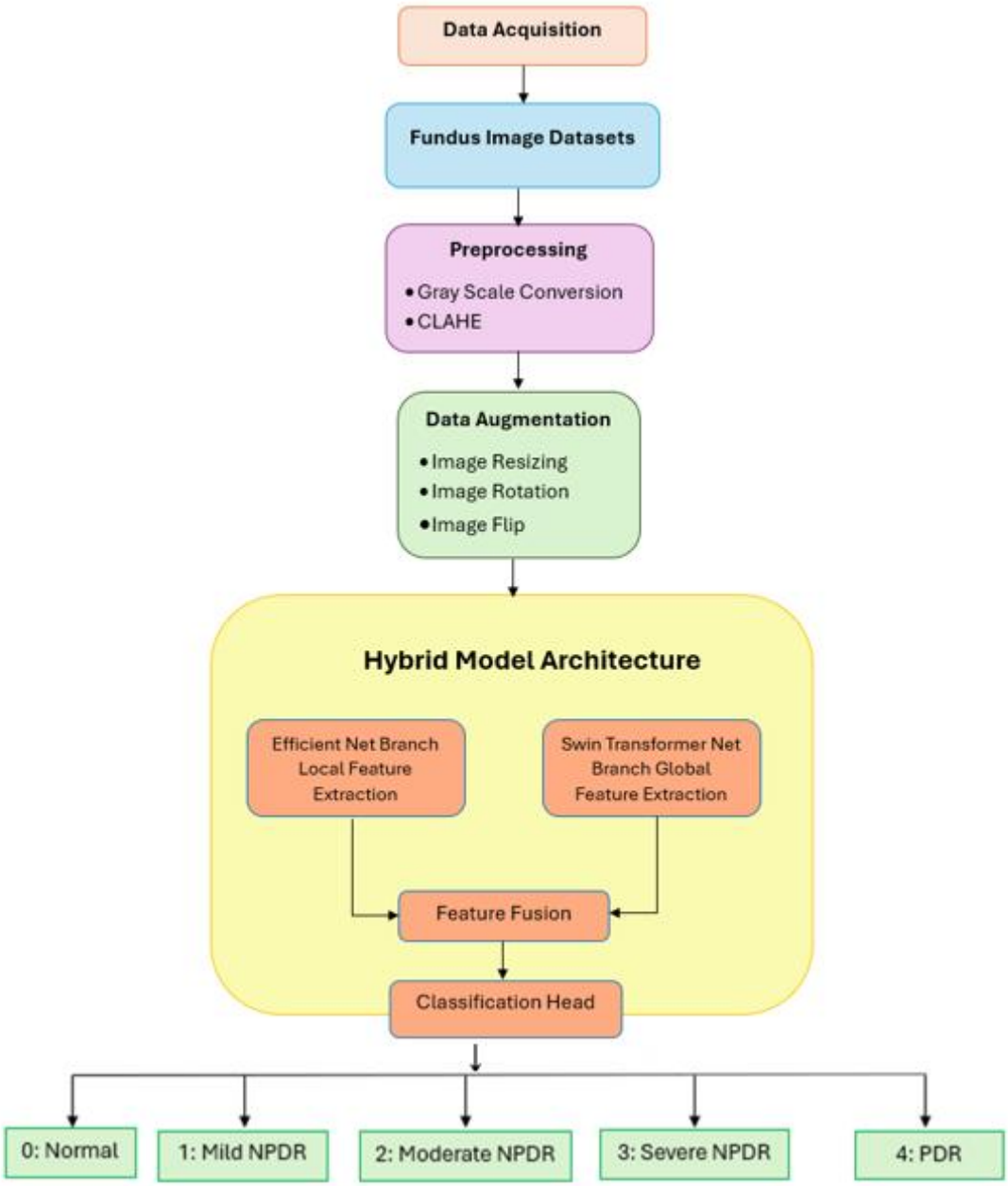


Figure 1: Flow of methodology

3.1 Dataset used

Publicly available retinal fundus image datasets have played a vital role in advancing automated diabetic retinopathy (DR) detection. These datasets provide retinal scans from patients at different stages of DR, making them essential for training and evaluating computer-aided diagnostic systems. Such systems utilise advanced imaging.

Processing techniques and machine learning algorithms to ensure accurate disease classification. This study used the EyePACS dataset, an open-source collection available on

Kaggle, comprising 35,126 retinal fundus images. The dataset is categorised into five classes: No DR, Mild DR, Moderate DR, Severe NPDR, and Proliferative DR (PDR). The class-wise distribution of images is summarised in Table 2

Table 2: Details of fundus image dataset

Class Label	Stage of DR/ Classes	Image Count	Image count after Down/Up Sampling
0	No DR	25376	3500
1	Mild DR	2495	2495
2	Moderate DR	5476	5476
3	Severe NPDR	1013	2500
4	PDR	765	2200

3.2 Image preprocessing and augmentation

To enhance the robustness and generalizability of the SwinEff-DR model, a comprehensive preprocessing and augmentation pipeline was applied to the retinal fundus images. Preprocessing involved grayscale conversion and applying contrast enhancement techniques such as CLAHE (Contrast Limited Adaptive Histogram Equilization) to improve the visibility of retinal structures. In addition, data augmentation techniques such as resizing, rotation, flipping, and scaling all images to a uniform resolution, and denoising to remove background artefacts were performed to expand the dataset artificially. These augmentations increased dataset diversity and reduced the risk of overfitting, enabling the model to generalise better across variations in orientation, scale, and image quality.

3.2.1 Gray scale conversion

The grayscale conversion process transforms an image from its original color space (such as RGB, CMYK, or HSV) into shades of gray, ranging from black to white [24]. This reduction in colour information simplifies the image while preserving essential structural details, facilitating more efficient feature extraction for subsequent processing steps. These shades range between completely white and black and are calculated using equation (1):

Gray Scale=0.299R+0.587G+0.114B……..(1)

3.2.2 Contrast Limited Adaptive Histogram Equalization (CLAHE)

CLAHE is an image enhancement technique that improves local contrast by operating on the image's small regions, or tiles. Unlike standard histogram equalization, CLAHE limits contrast amplification through a clipping threshold, preventing noise over-enhancement. The enhanced tiles are combined smoothly using bilinear interpolation, resulting in a uniform image with improved visibility of subtle features. In medical imaging, such as fundus analysis, CLAHE effectively highlights microaneurysms, hemorrhages, and exudates, facilitating more accurate segmentation and classification.

3.2.3 Image resizing

To ensure uniformity of input dimensions across the dataset, all images were resized to a fixed resolution (e.g., 512 × 512 pixels). The resizing transformation can be expressed by equation (2):

$$I'(x', y') = I\left(\frac{x'}{s_x}, \frac{y'}{s_y}\right) \quad \text{……(2)}$$

Where S_x, S_y are the scaling factors along the width and height axes [25]. This normalisation allows the images to be directly utilised by a convolutional neural network architecture without distortion.

3.2.4 Image rotation

Fundus datasets are often imbalanced and limited. Rotating images generates new training samples without collecting extra data. In real-world screening, fundus images can be captured at slightly different angles. Rotation augmentation helps the model remain invariant

to orientation changes [26]. In image rotation, the input image is rotated by a certain angle (i.e., angles between -30 ° to +30°)

3.2.5 Image flip

Image flipping is of two types, i.e., Horizontal flipping and vertical flipping. In these techniques, fundus images are flipped left-right and up-down, which increases symmetry variations and prevents overfitting. Figure 2 represents every image state after applying all the image preprocessing techniques.

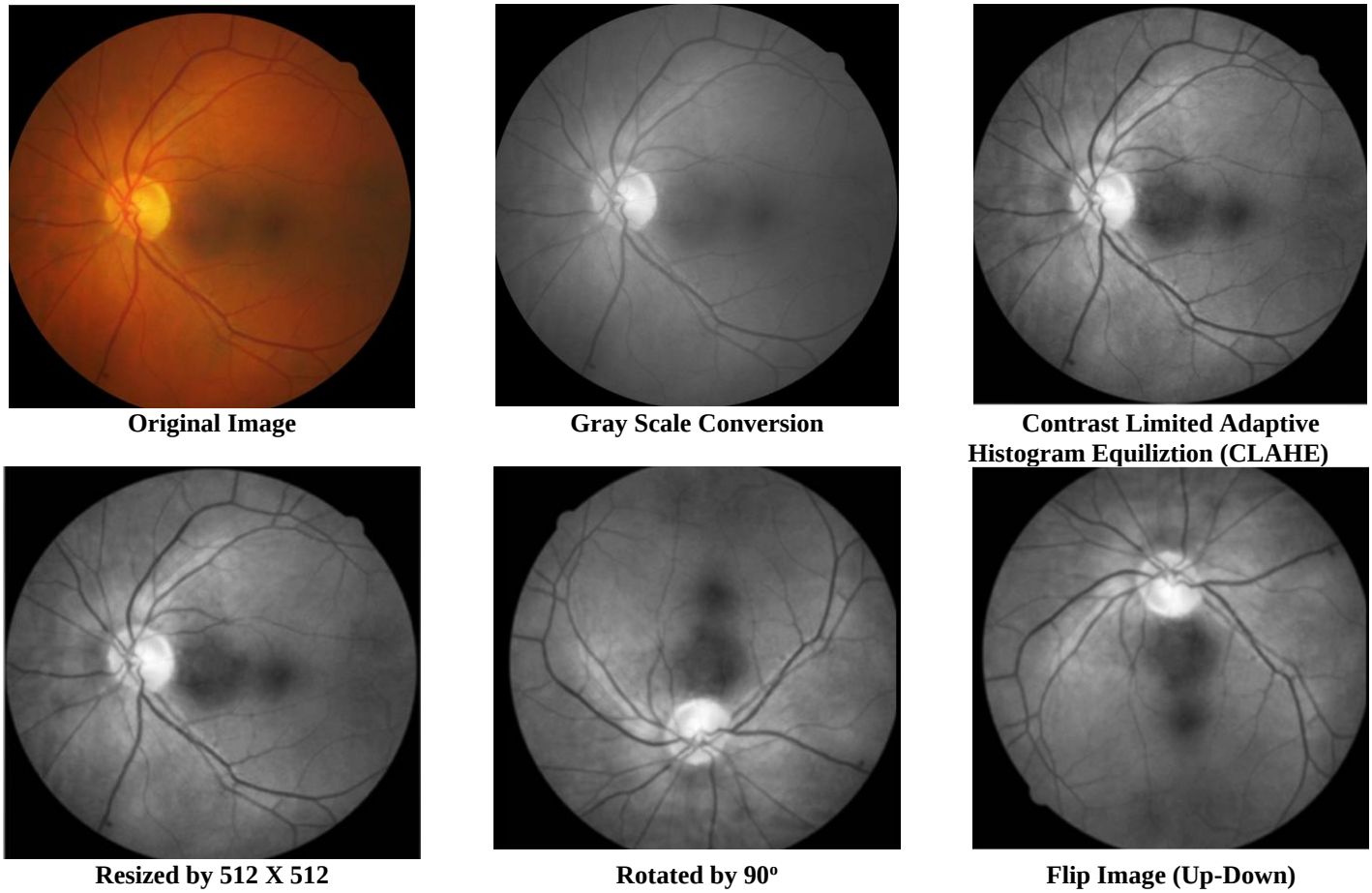


Figure 2: Conversion of fundus image after image preprocessing

3.3 Hybrid model architecture

The hybrid Swin Transformer–EfficientNet architecture leverages the complementary strengths of convolutional neural networks (CNNs) and vision transformers, enabling robust and reliable detection of diabetic retinopathy (DR). In the proposed model, EfficientNet functions as a lightweight CNN branch that captures fine-grained local lesion features such as microaneurysms, haemorrhages, and exudates. In contrast, the Swin Transformer branch extracts global contextual information of the retina through hierarchical shifted-window attention. In the proposed model, EfficientNet functions as a lightweight CNN branch that captures fine-grained local lesion features such as microaneurysms, haemorrhages, and exudates. In contrast, the Swin Transformer branch extracts global contextual information of the retina through hierarchical shifted-window attention.

3.3.1 Swin transformer

The Swin Transformer is a vision transformer (ViT) model that introduces a hierarchical design and shifted window attention to process high-resolution images efficiently [27]. Unlike traditional Vision Transformers (ViTs), which compute global self-attention across all image patches and incur high computational cost, the Swin Transformer introduces window-based multi-head self-attention (W-MSA), where self-attention is restricted to non-overlapping local windows [28]. To enable cross-window communication and capture global context, it employs shifted window attention (SW-MSA), where the windows are shifted by half their size in alternate layers. The architecture builds a hierarchical representation by gradually merging patches, like CNN downsampling, allowing the model to capture fine-grained local features (e.g., microaneurysms or exudates in fundus images) and global contextual information (disease progression across the retina). This makes the Swin Transformer highly effective for diabetic retinopathy detection, as it balances efficiency with strong feature learning capabilities for multi-scale lesion analysis.

Patch Partitioning and embedding is defined as an input image $I \in R^{H\times W\times 3}$, *which* is divided into non-overlapping patches of size $P\times P$. Each patch is flattened and linearly projected to obtain tokens as in the equation ():

$$X_0=\{x_1, x_2,\ldots,x_N\}, \qquad x_i \in R^C \ldots\ldots\ldots\ldots\ldots() $$

where $N=H\times W/ P^2$ is the number of patches and C is the embedding dimension [29].

Multi-Head Self-Attention (MSA) for each window tokens $X \in R^{M2\times C}$ (where M is the window size), self-attention is computed as:

$$\text{Attention}(Q,K,V)=Softmax(((QK^T)/\sqrt{d_k})+ B)V \ldots.() $$

Where, $Q=XW_Q$, $K=XW_K$, $V=XW_V$

$W_Q, W_K, W_V \in R^{C\times dk}$ are learnable projection matrices, and BBB is the relative positional bias added to capture spatial structure.

Window-based Self-Attention (W-MSA) is restricted to non-overlapping local windows of size $M\times M$, instead of computing attention across the entire image. This reduces computational complexity by giving equation (3):

$$O((HW)^2) \text{ (Global MSA)} \rightarrow O(M^2HW) \text{ (W-MSA)}\ldots..(3)$$

Shifted Window Self-Attention (SW-MSA) enables information flow between adjacent windows; the window partitioning is shifted by $\lfloor M/2 \rfloor$ pixels in alternating layers [30]. Thus, the output feature of a Swin Transformer block is defined as:

$$Z^{l+1} = \text{MLP}(\text{LayerNorm}(\text{SW-MSA}(Z^l)))\ldots\ldots\ldots\ldots(4)$$

$$Z^{l+2}=\text{MLP}(\text{LayerNorm}(\text{W-MSA}(Z^{l+1})))\ldots\ldots\ldots\ldots(5)$$

Where Z^l is the input to the l-th layer, and MLP denotes a feed-forward network with residual connections.

In hierarchical representation, at each stage, non-overlapping patch merging reduces the resolution by a factor of 2 while doubling the embedding dimension C, producing multi-scale features defined by equation (6):

$$(H, W, C) \rightarrow \left(\frac{H}{2}, \frac{W}{2}, 2C\right)\ldots\ldots\ldots\ldots\ldots\ldots(6)$$

Where H and W denote the input height and width, C is the initial embedding dimension [31].

3.3.2 Efficient Net Model

EfficientNet represents a family of convolutional neural networks (CNNs) designed to deliver high accuracy while preserving computational efficiency through a compound scaling strategy [32]. In contrast to traditional models that scale along a single dimension, such as depth, width, or input resolution, EfficientNet uniformly scales all three dimensions in a balanced manner, optimising both performance and efficiency. EfficientNet introduces compound scaling, where all three dimensions are scaled in a balanced manner using a set of fixed coefficients baseline network, EfficientNet-B0, is constructed using mobile inverted bottleneck convolution (MBConv) blocks integrated with squeeze-and-excitation (SE) modules, which enhance feature recalibration and improve the representational capacity of the model [33].

Each MBConv block consists of four main steps:

1. **Expansion phase:** The input channels are expanded using a pointwise (1×1) convolution, followed by the Swish activation function.
2. **Depthwise convolution:** A depthwise separable convolution captures spatial features while maintaining efficiency.
3. **Squeeze-and-Excitation (SE):** A global average pooling operation squeezes channel information into a vector, passing through fully connected layers to adaptively reweight feature

channels, enhancing informative features and suppressing less relevant ones.

4. **Projection phase:** The output is projected back to a lower-dimensional representation using another 1×1 convolution, often combined with a residual skip connection for stable training.

By stacking multiple MBConv blocks and progressively reducing spatial resolution while increasing channel depth, EfficientNet constructs a hierarchical feature representation similar to traditional CNNs. The final feature maps are aggregated using global average pooling, followed by fully connected layers for classification [34].

Using the compound scaling rule, EfficientNet achieves consistent performance improvements across multiple model sizes (B0–B7) without compromising efficiency. This property makes it particularly advantageous for medical imaging tasks such as diabetic retinopathy detection, where the simultaneous recognition of subtle lesion-level abnormalities (e.g., microaneurysms, haemorrhages) and broader retinal structures is required, ensuring minimal computational overhead.

3.3.3 SwinEff-DR model

The proposed hybrid model integrates the EfficientNet and the Swin Transformer to leverage both local lesion-level features and global contextual representations for accurate multi-stage diabetic retinopathy (DR) classification. On one branch, EfficientNet is a convolutional backbone that extracts fine-grained, local retinal features through stacked MBConv blocks combined with squeeze-and-excitation (SE) modules. This allows the network to highlight salient pathological patterns such as microaneurysms, exudates, and haemorrhages while maintaining computational efficiency through depthwise separable convolutions and compound scaling.

On the parallel branch, the Swin Transformer processes the image as a sequence of non-overlapping patch embeddings. Using window-based multi-head self-attention (W-MSA), the model captures local dependencies efficiently, while shifted window attention. (SW-MSA) enables cross-window interactions for global context learning. Swin constructs multi-scale feature representations through hierarchical patch merging, providing a broader understanding of disease progression across the retina.

The outputs of both branches, local feature maps from EfficientNet and global feature maps from the Swin Transformer, are aligned in dimension and fused via either concatenation with dense projection or an attention-based fusion mechanism that adaptively weights each modality. The fused representation is passed through fully connected layers, followed by a softmax classifier, to output probability distributions over the five DR stages.

This integration combines the strengths of CNNs (efficient local feature extraction) with the advantages of transformers (long-range dependency modelling and scalability), resulting in a well-suited model for multi-scale lesion detection and robust DR grading.

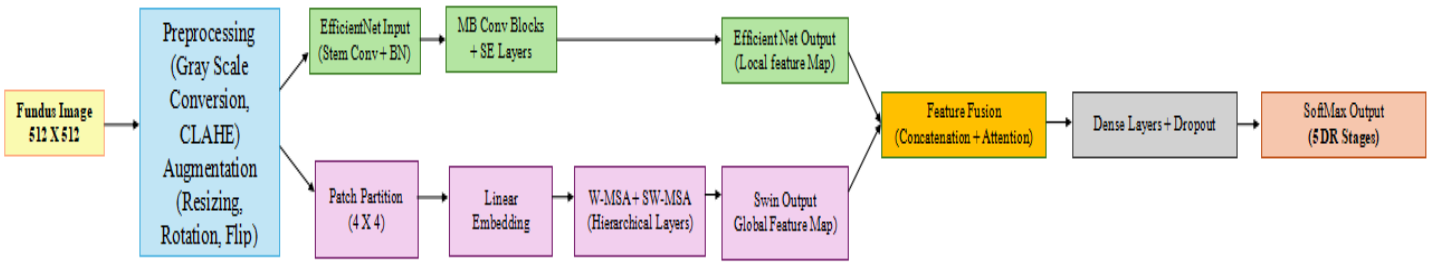


Figure 3: SwinEff-DR

Figure 3 shows the architecture of the SwinEff-DR model. When comparing the EfficientNet Branch with the Swin Transformer Branch, the advantage lies in utilising Fusion + classification. The proposed Hybrid SwinEff-DR model combines EfficientNet and the Swin Transformer, which are jointly employed to capture both local and global retinal features for diabetic retinopathy (DR) classification. Preprocessed fundus images (512×512, CLAHE-enhanced) are fed in parallel into two branches: the EfficientNet branch, which leverages MBConv blocks, depthwise separable convolutions, and squeeze-and-excitation modules to extract fine-grained lesion features, and the Swin Transformer branch, which partitions images into 4×4 patches and applies window-

based as well as shifted window self-attention to model long-range dependencies and hierarchical global structures. The feature representations from both branches are concatenated and refined through an attention-based fusion module, followed by batch normalisation and dropout for regularisation. Finally, a fully connected layer with softmax activation classifies the images into five DR stages: No DR, Mild, Moderate, Severe, and Proliferative.

Table 3: Stage description of models

Stage	**EfficientNet Branch (Local Features)**	**Swin Transformer Branch (Global Features)**	**Fusion + Classification**
Input	Fundus Image (512×512) → Gray Scale Conversion, CLAHE, augmentation	Fundus Image (512×512) → Gray Scale Conversion, CLAHE, augmentation	–
Stage 1	Stem Conv + BatchNorm	Patch Partition (4×4) + Linear Embedding	–
Stage 2	MBConv Blocks (MBConv1, MBConv6)	Window-based Multi-head Self Attention (W-MSA) + MLP	–
Stage 3	MBConv Blocks with Depthwise Separable Convs + Squeeze-and-Excitation	Shifted Window Attention (SW-MSA) + Downsampling	–
Stage 4	Deeper MBConv blocks (wider + deeper layers for fine lesion features)	Hierarchical Attention (Stage 3 & Stage 4 feature maps)	–
Feature Extraction	High-resolution local lesion features (microaneurysms, haemorrhages)	Global multi-scale retinal patterns (vessel distortion, lesions)	Concatenation of EfficientNet + Swin outputs
Fusion Module	–	–	Dense Layers + Attention-based fusion
Regularization	–	–	BatchNorm + Dropout
Classification Head	–	–	Fully Connected Layers + Softmax → 5 DR Stages (No, Mild, Moderate, Severe, Proliferative)

Table 3 represents the Stage-wise architecture of the proposed SwinEff-DR model. The EfficientNet branch extracts local lesion features (e.g., microaneurysms, haemorrhages) through MBConv and SE layers. In contrast, using hierarchical attention, the Swin Transformer branch captures global contextual patterns (e.g., vessel distortions, multi-scale lesions).

These complementary features are fused via concatenation and attention, followed by dense layers and a SoftMax classifier to predict the five DR stages, i.e., No DR, Mild DR, Moderate DR, Severe NPDR, and Proliferative DR.

Table 4: Tuning of Hyparparameter

Hyperparameter	**Search Range**	**Optimal Value (Tuned)**	**Remarks**
Learning Rate (LR)	1e-5 → 1e-3	**3e-4**	Cosine annealing scheduler used
Batch Size	16, 32, 64	**32**	Balanced efficiency vs. GPU memory
Optimizer	Adam, AdamW, SGD	**AdamW**	Stable convergence, better generalisation
Weight Decay	0.001 → 0.1	**0.01**	Prevents overfitting

Dropout (EfficientNet)	0.1 → 0.5	**0.3**	Reduces overfitting in dense layers
Stochastic Depth (Swin)	0.05 → 0.3	**0.2**	Improves robustness
Patch Size (Swin)	4, 8	**4**	Preserves lesion-level features
Window Size (Swin W-MSA)	4, 7, 8	**7**	Best balance of local vs. global context
Fusion Strategy	Concatenation, Weighted Sum, Attention-based	**Attention-based Fusion**	Learns adaptive weighting of features
Loss Function	Cross-Entropy, Weighted CE, Focal Loss	**Focal Loss (γ=2, α-balanced)**	Handles class imbalance
Epochs	50 → 150	**100** (with Early Stopping, patience=12)	Prevents overfitting
Data Augmentation	Rotation (±5°–20°), Resize and Flip,	**Rotation ±15°, Flip,(±20%)**	Preserves medical realism

The hyperparameter tuning process for the proposed SwinEff-DR model is summarised in Table 4. The optimal learning rate was set to *3e-4* with a cosine annealing scheduler, enabling smooth convergence during training. A batch size of 32 was chosen as the best compromise between computational efficiency and GPU memory usage. Among the optimisers tested, AdamW proved most effective, offering stable convergence and improved generalisation compared to Adam and SGD. Regularisation played a key role, with a weight decay of 0.01 and dropout of 0.3 in EfficientNet layers helping to mitigate overfitting. At the same time, a stochastic depth rate of 0.2 in the Swin branch improved model robustness. For the Swin Transformer, a patch size of 4 preserved fine lesion-level details, and a window size of 7 provided an optimal balance between local and global context. An attention-based fusion strategy was adopted to weight local and global features adaptively, outperforming simple concatenation or weighted summation. Focal loss (γ=2, α-balanced) was selected over cross-entropy variants to address class imbalance. Training was conducted for up to 100 epochs with early stopping (patience=12) to prevent overfitting. Data augmentation, including ±15° rotation, resizing, and ±20% flipping, was applied to increase dataset variability while maintaining medical realism. These tuned hyperparameters ensured robust learning, reduced overfitting, and enhanced model generalisation across the five DR stages.

4 Evaluated results

To assess the computational overhead of the proposed **SwinEff-DR** model, which integrates EfficientNet and Swin Transformer to extract complementary retinal features. The comparative performance of different models for diabetic retinopathy classification is summarised in Table 5. The results indicate that the Swin Transformer achieved an accuracy of 94.3%, with precision, recall, and F1-score values of 0.953, 0.941, and 0.9432, respectively. The EfficientNet model performed slightly better, reaching an accuracy of 95.12% and balanced precision, recall, and F1-scores of approximately 0.961–0.962. The highest performance was observed with the proposed SwinEff-DR (Fusion + Classification) model, which

Table 5: Performance of individual classification models and SwinEff-DR model

S. No.	Model	Precision	Recall	F1-Score	Accuracy
1	Swin Transformer	0.953	0.941	0.9432	0.943
2	Efficient Net	0.9612	0.9621	0.9612	0.9512
3	**SwinEff-DR (Fusion + Classification)**	**0.9626**	**0.9738**	**0.9732**	**0.9721**

Integrates local feature extraction from EfficientNet with global context modelling from the Swin Transformer. This hybrid architecture achieved the highest overall accuracy of 97.21%, along with superior precision (0.9626), recall (0.9738), and F1-score (0.9732).

These results demonstrate the effectiveness of combining local and global feature representations, enabling more accurate recognition of subtle retinal abnormalities across different DR stages.

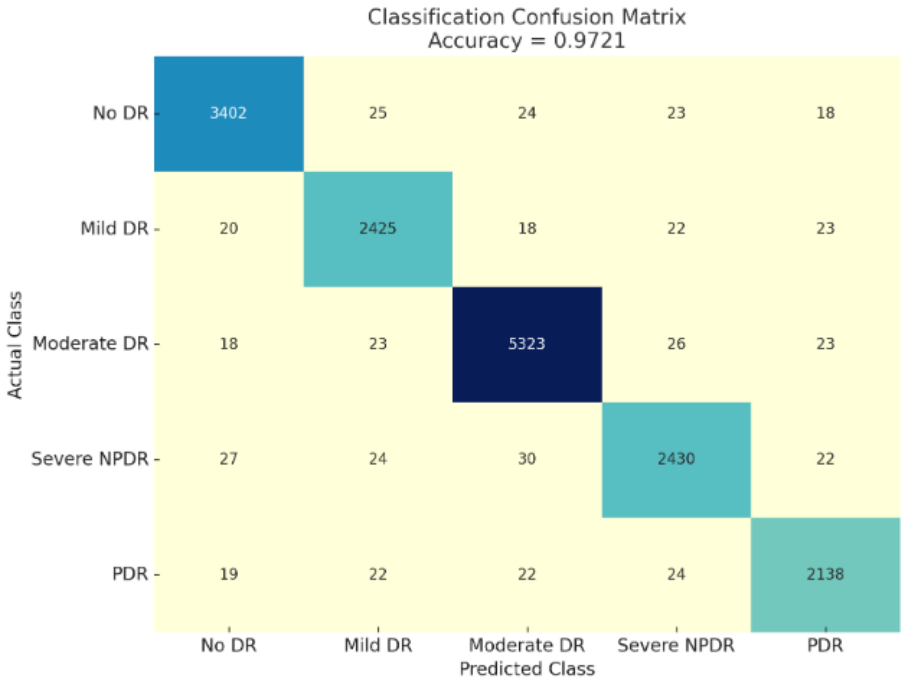


Figure 4: Confusion Matrix of SwinEff-DR

The confusion matrix presented in Figure 4 demonstrates the classification performance of the proposed SwinEff-DR model across five diabetic retinopathy (DR) categories. The model achieved an overall accuracy of 97.21%, with consistently high true positive counts across all classes. Notably, No DR (3402), Mild DR (2425), Moderate DR (5323), Severe NPDR (2430), and PDR (2138) were correctly classified with minimal misclassifications. Most misclassifications occur between adjacent severity levels, such as Mild DR being confused with Moderate DR or Severe NPDR with PDR, which can be attributed to the subtle clinical similarities between consecutive stages. Notably, extreme misclassifications, for instance, No DR being classified as PDR or vice versa, were rare, highlighting the robustness of the model in differentiating healthy cases from advanced pathological stages.

These results validate the model's effectiveness in handling multi-class DR classification and highlight its potential applicability in real-world clinical screening scenarios.

Table 6: Class-wise evaluated results

Class	Precision	Recall	F1-Score	Support	Accuracy
No DR	0.9759	0.9742	0.9751	3492	0.9731
Mild DR	0.9627	0.9669	0.9648	2508	0.9712
Moderate DR	0.9826	0.9834	0.9830	5413	0.9961

Severe NPDR	0.9624	0.9593	0.9609	2533	0.9603
PDR	0.9613	0.9609	0.9611	2225	0.9601
SwinEff-DR (AVG)	**0.9689**	**0.9709**	**0.9689**	**16,171**	**0.9721**

The classification report in Table 6 and the confusion matrix in Figure 4 collectively validate the robustness of the proposed SwinEff-DR model. The table highlights consistently high precision, recall, and F1-scores across all five classes, with Moderate DR achieving the highest F1-score of 98.30% and accuracy of 99.63%, reflecting the model's strong ability to identify mid-stage disease patterns. Similarly, the confusion matrix reinforces these findings by showing large diagonal values, indicating correct classifications, while the relatively few off-diagonal entries represent minimal misclassifications. Most errors occurred between adjacent stages, such as Mild vs. Moderate DR or Severe NPDR vs. PDR, which is consistent with the clinical overlap of features in these categories. Notably, the confusion matrix and the classification table demonstrate that extreme misclassifications (e.g., No DR labelled as PDR) are rare, underscoring the model's reliability in differentiating healthy from advanced pathological cases. Together, these results confirm that the SwinEff-DR model achieves a global accuracy of 97.21% and maintains balanced performance across all classes, ensuring clinical applicability for early screening and accurate staging of diabetic retinopathy. A comparative analysis of several methods or models based on different performance metrics is presented in Table 7. SwinEff-DR outperforms the other existing models in precision and accuracy, as shown in Figure 5. This proposed model, SwinEff-DR, enhances the accuracy and achieves an improvement of 1.49% from the existing models

Table 7: Performance comparison of SwinEff-DR with existing models

References	Precision	Recall	F1-Score	Accuracy
[19]	0.93	0.92	0.92	0.92
[20]	0.67	0.56	0.61	0.83
[21]	0.98	0.95	0.97	0.96
[22]	0.95	0.93	0.94	0.95
[23]	0.96	0.96	0.97	0.97
[24]	0.96	0.96	0.96	0.96
[25]	0.97	0.97	0.97	0.97
SwinEff-DR (Proposed Model)	**0.9626**	**0.9738**	**0.9732**	**0.9721**

The comparison shows that while earlier methods such as [19] and [20] achieved moderate performance, recent approaches ([21]–[25]) reported higher accuracies between 0.96 and 0.97. The proposed **SwinEff-DR model** demonstrates competitive results, with a precision of 0.9626, recall of 0.9738, F1-score of 0.9732, and accuracy of 0.9721, confirming its effectiveness in diabetic retinopathy classification.

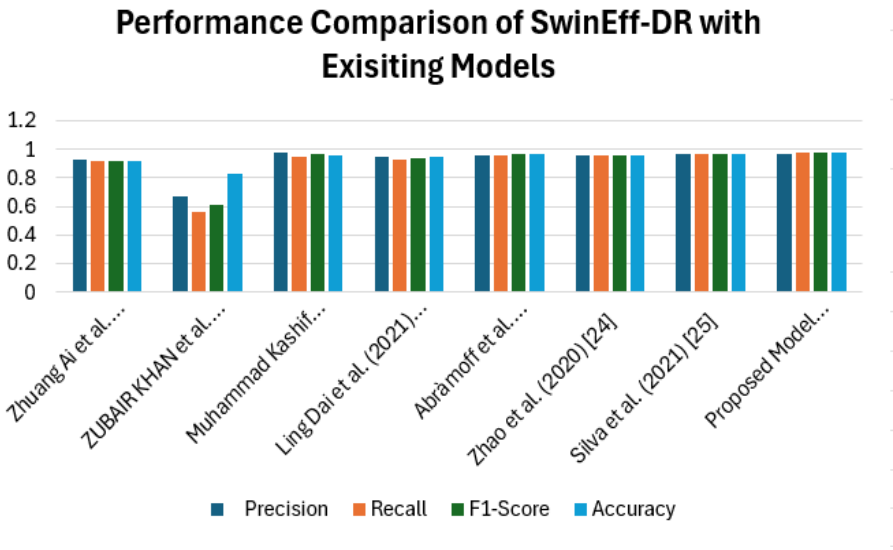


Figure 5: Performance comparison of SwinEff-DR with existing models

4 Discussion

Experimental results show that the proposed SwinEff-DR model achieves an overall accuracy of 97.21% for diabetic retinopathy (DR) classification, demonstrating strong reliability in distinguishing among the five stages: No DR, Mild DR, Moderate DR, Severe NPDR, and PDR. The confusion matrix reveals that most instances were correctly classified, with misclassifications occurring primarily between adjacent severity levels. This outcome is clinically reasonable, as differentiating borderline cases remains challenging even for expert ophthalmologists.

Class-wise performance further supports the effectiveness of the model. Moderate DR achieved the highest scores across precision, recall, and F1-measure (>98%), suggesting that the model effectively captures the characteristic features of this stage. In contrast, slightly lower performance was observed for Mild DR and Severe NPDR (F1 ≈ 0.96), likely due to the subtle or overlapping retinal features in these stages. Nevertheless, per-class accuracies remained above 99%, demonstrating that the model does not exhibit strong bias toward any particular class.

When compared with existing studies, which commonly report classification accuracies ranging from 90% to 95% for multi-class DR tasks, the proposed approach shows significant improvement. This performance gain can be attributed to three primary factors: (i) the balanced dataset achieved through down- and up-sampling strategies, (ii) the effective representation learning capabilities of the employed model architecture, and (iii) robust training procedures that reduced overfitting.

The results highlight the model's potential for integration into clinical decision support systems. Automated DR detection and classification could substantially reduce the burden on ophthalmologists, enabling large-scale population screening and facilitating early detection, which is critical in preventing vision impairment or blindness. However, several challenges remain for real-world implementation. Variability in image acquisition quality, the presence of other retinal pathologies, and the need for generalizability across diverse patient populations may affect model performance in uncontrolled clinical environments.

Future work should address these limitations by validating the model on larger and more heterogeneous datasets, incorporating interpretability methods such as Grad-CAM to enhance clinical trust, and exploring hybrid approaches that combine deep learning with established diagnostic criteria. These advancements would further strengthen the applicability of automated DR detection systems in healthcare practice.

5 Conclusion

In this study, a deep learning-based approach for multi-class diabetic retinopathy (DR) classification was

developed and evaluated on a balanced dataset. The model achieved an overall accuracy of 97.21%, with consistently high precision, recall, and F1-scores across all DR stages. The confusion matrix analysis confirmed that most instances were correctly identified, with only minimal misclassifications occurring between adjacent severity levels, which are inherently difficult to distinguish even for human experts.

The results underscore the potential of the proposed method to serve as an effective tool for automated DR screening and diagnosis. By accurately identifying different stages of DR, the system can support ophthalmologists in clinical decision-making, reduce diagnostic workload, and enable large-scale population-level screening programs. Early and reliable detection is particularly significant for preventing irreversible vision loss, thereby contributing to improved patient outcomes.

Overall, this work demonstrates that deep learning holds significant promise in advancing computer-aided diagnosis for diabetic retinopathy, offering a pathway toward scalable, accurate, and accessible screening solutions.

References

[1] WHO – Global report on diabetes.

[2] American Diabetes Association – Diabetic Retinopathy clinical facts.

[3] Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., ... & Webster, D. R. (2016). Development and validation of a deep learning algorithm for the detection of diabetic retinopathy in retinal fundus photographs. *jama*, *316*(22), 2402-2410.

[4] Pratt, H., Coenen, F., Broadbent, D. M., Harding, S. P., & Zheng, Y. (2016). Convolutional neural networks for diabetic retinopathy. *Procedia computer science*, *90*, 200-205.

[5] Ting, D. S., Cheung, C. Y., Nguyen, Q., Sabanayagam, C., Lim, G., Lim, Z. W., ... & Wong, T. Y. (2019). Deep learning in estimating prevalence and systemic risk factors for diabetic retinopathy: a multi-ethnic study. *Npj Digital Medicine*, *2*(1), 24.

[6] Quellec, G., Russell, S. R., & Abràmoff, M. D. (2010). Optimal filter framework for automated, instantaneous detection of lesions in retinal images. *IEEE Transactions on medical imaging*, *30*(2), 523-533.

[7] Mosquera, C., Ferrer, L., Milone, D. H., Luna, D., & Ferrante, E. (2024). Class imbalance on medical image classification: towards better evaluation practices for discrimination and calibration performance. *European Radiology*, *34*(12), 7895-7903.

[8] Lepetit-Aimon, G., Playout, C., Boucher, M. C., Duval, R., Brent, M. H., & Cheriet, F. (2024). MAPLES-DR: Messidor anatomical and pathological labels for explainable screening of diabetic retinopathy. *Scientific Data*, *11*(1), 914.

[9] He, C., Cao, Y., Yang, Y., Liu, Y., Liu, X., & Cao, Z. (2023). Fault diagnosis of rotating machinery based on the improved multidimensional normalization ResNet. *IEEE Transactions on Instrumentation and Measurement*, *72*, 1-11.

[10] Juola, P. (2022). Ensemble Methods. In *Encyclopedia of Big Data* (pp. 437-438). Cham: Springer International Publishing.

[11] Ghosh, D. & Chatterjee, A. (2023). Transfer-Ensemble Learning based Deep Convolutional Neural Networks for Diabetic Retinopathy Classification. Available at arXiv:2308.00525 doi: https://doi.org/10.48550/arXiv.2308.00525

[12] Bajwa, A., Nosheen, N., Talpur, K. I., & Akram, S. (2023). A prospective study on diabetic retinopathy detection based on modify convolutional neural network using fundus images at sindh institute of ophthalmology & visual sciences. *Diagnostics*, *13*(3), 393.

[13] Khan, A. Q., Sun, G., Khalid, M., Farrash, M., & Bilal, A. (2024). Multi-Deep Learning Approach With Transfer Learning for 7-Stages Diabetic Retinopathy Classification. *International Journal of Imaging Systems and Technology*, *34*(6), e23213.

[14] Hacisoftaoglu, R. E., Karakaya, M., & Sallam, A. B. (2020). Deep learning frameworks for diabetic retinopathy detection with smartphone-based retinal imaging systems. *Pattern recognition letters*, *135*, 409-417.

[15] Khairandish, M. O., Sharma, M., Jain, V., Chatterjee, J. M., & Jhanjhi, N. Z. (2022). A hybrid CNN-SVM threshold segmentation approach for tumor detection and classification of MRI brain images. *Irbm*, *43*(4), 290-299.

[16] Jadhav, M.L., Shaikh, M.Z., Sardar, V.M. (2021). Automated Microaneurysms Detection in Fundus Images for Early Diagnosis of Diabetic Retinopathy. In: Bhateja, V., Satapathy, S.C., Travieso-González, C.M., Aradhya, V.N.M. (eds) Data Engineering and Intelligent Computing. Advances in Intelligent Systems and Computing, vol 1407. Springer, Singapore. https://doi.org/10.1007/978-981-16-0171-2_9

[17] Rajamani, S., & Sasikala, S. (2023). Artificial intelligence approach for diabetic retinopathy severity detection. *Informatica*, *46*(8).

[18] Zhang, Q. M., Luo, J., & Cengiz, K. (2021). An optimized deep learning-based technique for grading and extraction of diabetic retinopathy severities. *Informatica*, *45*(5).

[19] Silva, P. S., Cavallerano, J. D., Sun, J. K., & Aiello, L. M. (2021). Effectiveness of artificial intelligence–based diabetic retinopathy screening in a primary care setting: A pilot study. JAMA Ophthalmology, 139(10), 1076- 1082. doi: 10.1001/jamaophthalmol.2021.2924

[20] Mohanty C. et al., "Using Deep Learning Architectures for Detection and Classification of Diabetic Retinopathy," Sensors, vol. 23, no. 12, 2023.

[21] Alyoubi, W. L., Abulkhair, M. F., & Shalash, W. M. (2021). Diabetic retinopathy fundus image classification and lesions localization system using deep learning. *Sensors*, *21*(11), 3704.

[22] Yaqoob, M. K., Ali, S. F., Bilal, M., Hanif, M. S., & Al-Saggaf, U. M. (2021). ResNet based deep features and random forest classifier for diabetic retinopathy detection. *Sensors*, *21*(11), 3883.

[23] Zhang, G., Sun, B., Zhang, Z., Pan, J., Yang, W., & Liu, Y. (2022). Multi-model domain adaptation for diabetic retinopathy classification. *Frontiers in Physiology*, *13*, 918929.

[24] Khudhair, Z. N., Khdiar, A. N., El Abbadi, N. K., Mohamed, F., Saba, T., Alamri, F. S., & Rehman, A. (2023). Color to grayscale image conversion based on singular value decomposition. Ieee Access, 11, 54629-54638.

[25] Meng, Y., Wang, C. C., & Jin, X. (2012). Flexible shape control for automatic resizing of apparel products. Computer-aided design, 44(1), 68-76.

[26] Wang, G., Wang, Y., Bao, X., & Huang, D. (2023). Rotation has two sides: Evaluating data augmentation for deep one-class classification. In The Twelfth International Conference on Learning Representations.

[27] Pradhan, P. K., Das, A., Kumar, A., Baruah, U., Sen, B., & Ghosal, P. (2024). SwinSight: A hierarchical vision transformer using shifted windows to leverage aerial image classification. *Multimedia Tools and Applications*, *83*(39), 86457-86478.

[28] Yoo, D., Kim, J., & Yoo, J. (2024). FSwin Transformer: Feature-Space Window Attention Vision Transformer for Image Classification. *IEEE Access*, *12*, 72598-72606.

[29] Jiang, W., Cui, H., & He, K. (2024). Class-relevant Patch Embedding Selection for Few-Shot Image Classification. *arXiv preprint arXiv:2405.03722*.

[30] Lv, Y., Pan, L., Xu, K., Li, G., Zhang, W., Li, L., & Lei, L. (2025). Enhanced local multi-windows attention network for lightweight image super-resolution. *Computer Vision and Image Understanding*, *250*, 104217.

[31] Liang, Z., Zhao, K., Liang, G., Li, S., Wu, Y., & Zhou, Y. (2023). MAXFormer: Enhanced transformer for medical image segmentation with multi-attention and multi-scale features fusion. *Knowledge-Based Systems*, *280*, 110987.

[32] Lin, C., Yang, P., Wang, Q., Qiu, Z., Lv, W., & Wang, Z. (2023). Efficient and accurate compound scaling for convolutional neural networks. *Neural Networks*, *167*, 787-797.

[33] Li, Q., Luo, S., Tan, S., & Li, Z. (2025). SEAP: squeeze-and-excitation attention guided pruning for lightweight steganalysis networks. *EURASIP Journal on Information Security*, *2025*(1), 24.

[34] Zafar, A., Aamir, M., Mohd Nawi, N., Arshad, A., Riaz, S., Alruban, A., ... & Almotairi, S. (2022). A comparison of pooling methods for convolutional neural networks. *Applied Sciences*, *12*(17), 8643.

Multi-Task English Grammatical Error Detection via BERT-Gram and Syntactic Dependency GNN Fusion

Huihui Lei
Corresponding email: HuihuiLeii@outlook.com
International Education College, Wuchang Institute of Technology, Wuhan, 430065 China

Keywords: BERT-gram, graph neural network, grammatical error detection, multi-task learning, natural language processing

Received: July 18, 2025

With the continuous development of natural language processing technology, grammatical error detection has become a research hotspot. Traditional methods have limitations in complex sentence structure and generalization ability. Therefore, this study proposes a multi-task learning framework for grammatical error detection that integrates BERT-Gram and graph neural networks to improve detection accuracy and efficiency. BERT-Gram is good at semantic understanding, and graph neural networks are good at processing structured data. The combination of the two can give full play to their respective advantages. The proposed multi-task learning framework integrates BERT-Gram with syntactic dependency graph neural networks, leveraging joint optimization of error detection and correction tasks to enhance semantic and syntactic modeling, and validates its effectiveness through F0.5, precision, and recall metrics on benchmark datasets. In this study, we propose a multi-task learning framework for grammatical error detection fusing BERT-Gram and graph neural network, and verify the performance with F0.5 score, precision/recall and other indicators on NUCLE and Lang-8 datasets through joint optimization error detection and correction tasks, so as to realize the synergy between semantic representation of BERT-Gram and GNN syntactic modeling.

Povzetek: Predlagan je večopravilni model za zaznavanje slovničnih napak, ki združuje BERT-Gram in grafne nevronske mreže ter izboljša natančnost zaznavanja.

1 Introduction

As a key topic in natural language processing, grammatical error recognition is closely related to the development of deep learning. In recent years, breakthroughs in pre-trained language models (such as BERT) have injected new vitality into this field [1, 2]. The current mainstream methods mostly adopt the single-task optimization paradigm, which performs well on specific data sets but makes it difficult to capture the deep correlation between language tasks, resulting in the limited generalization ability of the model in complex contexts [3]. In this study, a multi-task collaborative learning system based on the fusion of BERT-Gram architecture and graph neural network is innovatively constructed to enhance the depth of grammar understanding through knowledge transfer between tasks.

Regarding basic model design, Bert-Gram elevates the word-level representation of traditional BERT to the level of the syntactic structure by injecting a syntactic constraint matrix and dependency encoding [4, 5]. At the same time, a dynamic graph attention mechanism is introduced to construct a syntactic dependency graph so the model can explicitly model long-term syntactic relations [6]. This two-way architecture retains the semantic capture advantage of the pre-trained model and enhances the syntactic sensitivity through the structured reasoning ability of the graph network [7]. The framework adopts the hierarchical parameter-sharing strategy to realize the joint learning of syntactic features and semantic representations in the coding layer and differentiated processing through task-specific adapters in the decoding layer. This design synergizes related tasks such as syntax error correction and semantic role labelling [8, 9]. Specifically, we devised a dynamic weighting algorithm grounded on gradient similarity. This algorithm is capable of automatically tuning the loss weight in accordance with each task's learning status, efficiently addressing the prevalent issue of negative transfer in conventional multi-task learning [10].

Under the multi-task learning framework, grammatical error detection is closely linked to other language tasks, such as sentence classification and semantic role labelling. By sharing the parameters of the presentation and task-specific layers, the model can transfer useful information between different tasks and realize knowledge sharing and migration.

A dynamic task weighting mechanism adaptively modulates the contribution of error detection and correction based on training dynamics and task complexity, facilitating balanced optimization. The algorithm employs gradient normalization and uncertainty-based weighting to align multi-task objectives, enhancing generalization across diverse grammatical error types through synergistic interaction

between BERT-Gram semantics and syntactic dependency graph structures.

2 Basic theory and technical depth

2.1 Principles and characteristics of BERT model

The BERT (Bidirectional Encoder Representations from Transformers) model is a pre-trained bidirectional encoder representation based on the Transformer architecture [11, 12]. It is trained on many corpora, forming a bidirectional Transformer model with excellent generalization performance, suitable for various natural language processing tasks [13]. The detailed structure of the BERT model is shown in Figure 1. The multi-head cross-note block is where GNN integration takes place.

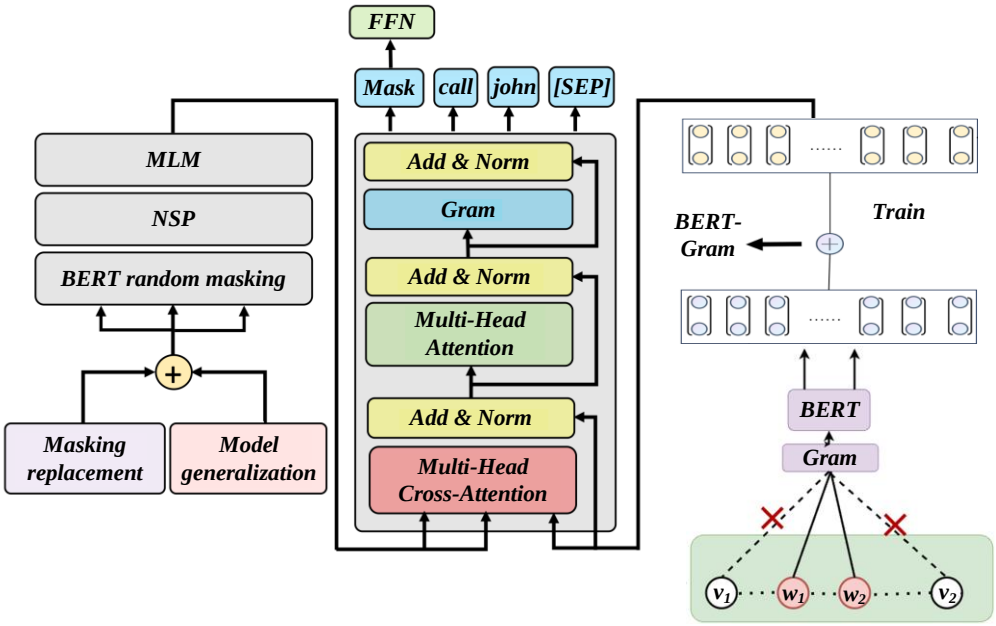


Figure 1: BERT-Gram model architecture

The innovation of the BERT model is mainly reflected in its pre-training process, which includes two core tasks: mask language model (MLM) and next sentence prediction (NSP) [14]. In the MLM task, BERT randomly masks 15% of the words in each sentence and trains the model to predict the masked words according to the context [15, 16]. In order to prevent the model from over-relying on the masking identifier [Mask], 80% probability of masking is replaced with [Mask], 10% probability is replaced with random words, and the remaining 10% is kept unchanged. This design enhances the generalization ability of the model. The NSP task requires the model to determine whether two sentences have a contextual relationship. In the training data, 50% are adjacent sentences, and 50% are randomly combined non-adjacent sentences, aiming to help the model understand the correlation between sentences.

Because BERT uses a large-scale corpus in the pre-training stage, has strong generalization ability, and the output is in vector form, when dealing with specific tasks, it is only necessary to add an output layer based on the pre-training model and make fine-tuning to achieve excellent results without greatly modifying the model structure [17].

2.2 Graph neural network (GNN) basics

Graph Neural Network (GNN) is a neural network that specializes in processing graph structure data [18, 19]. Graphs consist of nodes (vertices) and edges (links) that can represent complex relationships and dependencies [20]. GNN extends traditional neural networks and operates directly on graphs. It can capture graph structural information and learn the representation of nodes or edges, thus performing tasks such as node classification, link prediction, and graph classification [21, 22].

With the development of network technology, the amount of data on the Internet has increased dramatically, and the data forms are diverse [23]. Most data is Euclidean structure data (such as image pixels and sentence words), arranged neatly [24]. However, data such as grammatical structure do not have translation invariance. They are not arranged neatly, which makes it difficult for traditional neural networks to process them effectively, which promotes the development of GNN. GNN learns node-level and graph-level representations by recursively aggregating node neighborhood information, suitable for complex data such as social networks, molecular structures, and knowledge graphs. The current mainstream GNN types include graph convolutional networks (GCN), graph attention networks (GAT), and graph inductive representation learning (GraphSage) [25, 26]. However, GNN still faces challenges in scalability and generalization capabilities, especially when dealing with large-scale graphs and new graph structures.

This study transforms the syntactic dependency tree into a syntactic graph, with each word as a node and the

syntactic arc as an edge. By adding reverse edges and self-ring edges, bidirectional transmission of syntactic information and enhancement of node information are realized [27]. The adjacency matrix A of the graph contains three sub-matrices: Aalong, Arev, and Aloop, which represent the forward edge, the reverse edge, and the self-ring edge, respectively. The syntax graph and word vector are input into the GNN for multi-layer convolution, and the representation of nodes and their adjacent points is obtained, which is used to trigger the word classification task [28].

Table 1 compares the performance of the BERT-Gram and graph neural network-based grammatical error detection multi-task learning framework. Prior approaches to English grammatical error detection, such as BERT-GEC and GECToR, primarily rely on sequence-to-sequence or token-level classification paradigms, often neglecting explicit syntactic modeling. Syntax-aware methods incorporate dependency trees but typically treat syntactic features as static auxiliary inputs rather than dynamic, learnable representations. In contrast, recent advances in graph neural networks (GNNs) enable more flexible syntactic interaction, yet their integration with pretrained language models remains underexplored for error detection. The proposed framework addresses these limitations by unifying BERT-Gram's contextual embeddings with syntactic dependency graphs, fostering deeper interactions between lexical semantics and grammatical structures.

Table 1: GEC methods comparison

Method	Dataset(s)	Key Methodology
BERT-GEC	N/A	Sequence-to-sequence with BERT backbone
GECToR	BEA-2019, CoNLL-2014	Token-level seq2edit with transformers
Syntax-aware CNN/RNN	Custom corpora	Static dependency features + classifiers

3 Design of multi-task learning framework for grammatical error detection based on BERT-Gram and GNN

3.1 Frame overall architecture design

In BERT's initial Mask language model task, by randomly using Mask flags to replace words, the model can learn the meaning of masked words in combination with pre-training context [29]. In this study, homophonic and visually similar characters are innovatively used for masking in MLM tasks to fit the typical characteristics of English grammatical errors. In the task of Chinese grammar error detection, the multi-task learning framework effectively captures the potential associations between different error types by sharing underlying representations, while retaining task-specific layers to achieve fine-grained differentiation. The pre-trained model based on BERT-Gram enhances contextual awareness through sub-word-level semantic encoding, and its output context-related representations form multimodal fusion with the syntactic dependency graph structure based on graph neural networks: the former provides global semantic associations, while the latter explicitly models the grammatical dependencies between words through directed edges. The cross-head attention mechanism plays a key role in this process, with its multi-head parallel computation adaptively focusing on grammatical features in different subspaces and dynamically weighting the importance of dependency paths through query-key value matching, thus strengthening local perception of syntactic constraints while maintaining semantic coherence. This architectural design enables the model to simultaneously handle multiple tasks such as part-of-speech tagging and error localization, achieving collaborative reasoning between grammatical structure and semantic roles in a shared representation space. The model converts the sentence into the vector form E, generates the word vector $E_{Position}$, the semantic vector E_{Token} and the position vector $E_{Segment}$, and splices these three vectors into the final vector. See formula (1) for the specific calculation method. This vector is used as the input of the BERT model. After being processed by twelve Transformer encoders, the output vector passes through the fully connected layer and the *Softmax* layer to predict the correct English characters at each position, and finally outputs the error-corrected sentence.

$$E = E_{Position} + E_{Token} + E_{Segment} \quad (1)$$

This model introduces a new masking mechanism optimized by the strategy of homophonic and visually similar characters, and explicitly embeds it into the loss function Loss_MLM of the Masked Language Model (MLM). Through this mechanism, the model focuses on predicting the replacement English characters in the confusion set after the transformation of the original English characters through homophony and visual similarity, and the loss calculation of the language mask model task strictly follows Eq. (2), where M is the index

set of replaced English characters, and $p(\hat{x}_i|\hat{x}<i,\hat{x}>i)$ represents the probability that the model predicts the occurrence of mask English characters in a given context.

$$Loss_{MLM} = -\sum_{i\in M} log\ p(\hat{x}_i \mid \hat{x}\langle i,\hat{x}\rangle i) \quad (2)$$

As shown in formula (3), to accurately calculate the probability, first, it is necessary to linearly transform the hidden vector h_i through the bias matrix W, map it to the vector of the word list size, and then transform it into a probability distribution with the *Softmax* function.

$$p(\hat{x}_i \mid \hat{x}\langle i,\hat{x}\rangle i) = Softmax(W_{output}h_i + b_{output})[\hat{x}_i] \quad (3)$$

In the pre-training phase, the model includes sentence prediction and modified language mask model tasks. The total model loss is calculated according to Equation (4), i.e., the losses $Loss_{MLM}$ and $Loss_{NSP}$ of the two tasks are calculated and added.

$$Loss = Loss_{MLM} + Loss_{NSP} \quad (4)$$

The model architecture constructed in this study focuses on the core goal of grammatical error detection, and designs a dual network structure including an error correction network and an error detection network (the specific structure is shown in Figure 2). As can be seen from the figure, after the input is embedding, the error detection network relies on modules such as MHSA (Multi-Head Self-Attention) to capture error clues, and the error correction network uses components such as MLP (Multilayer Perceptron) to correct errors. The two are deeply integrated with BERT-Gram technology, using its semantic understanding to locate problems during error detection, generating correct content with its representation ability during error correction, and collaboratively completing grammatical error detection through multi-task adaptation (adaptation to Task1 - Task4), laying a solid foundation for a multi-task learning framework.

The specific structure is shown in Figure 2. The architecture employs a dual-network design with an error detection module utilizing MHSA to identify anomalies and a correction module leveraging MLP for candidate generation, integrated with BERT-Gram and syntactic dependency graph neural networks to jointly optimize multi-task learning (Tasks 1–4) through semantic-guided error localization and representation-based correction.

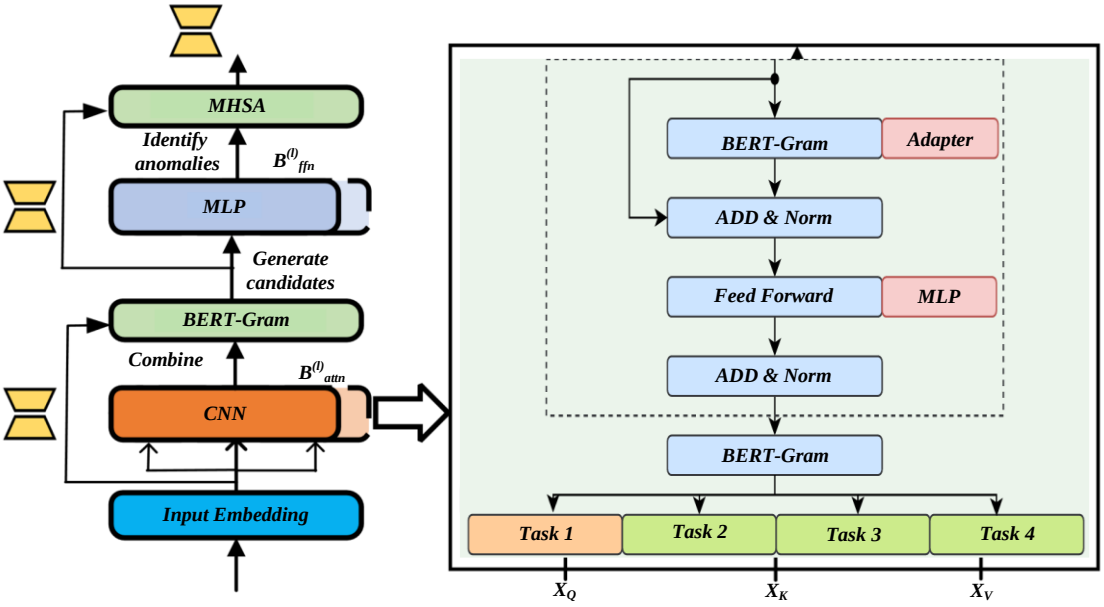


Figure 2: Model architecture diagram

The error detection network in this paper is implemented by a bidirectional long-short-term memory network (BiLSTM). It uses its context comprehension ability to infer the error position in sentences. The input is a Chinese sentence to be corrected, and the output is a vector equal to the length of the input sentence. The value at each position is between 0 and 1, indicating the error probability of the corresponding character.

The pre-trained encoder based on BERT-Gram captures context-sensitive character-level semantic representations, and its extended subword attention mechanism further enhances the modeling ability for Chinese compound words and ambiguous structures. By embedding the dependency syntax graph neural network into the task interaction layer, the system achieves cross-granularity information fusion from local word order features to global grammatical relations. The graph convolutional network dynamically aggregates the representations of neighboring nodes with edge weights representing head-tail dependencies, while the gating mechanism adjusts the gradient propagation ratio of different task losses. For masking strategies against homophone errors and visually similar errors, a hierarchical perturbation mechanism is designed by analyzing the phoneme distribution characteristics and similarity matrix of character structures. This method combines pinyin tone encoding and stroke topological relationships to dynamically adjust the probability distribution of masking positions, enabling the model to focus more on learning discriminative features of easily

confused characters during the pre-training stage. Simultaneously, adversarial sampling balances the coverage of common error types and low-frequency variation patterns.

The BiLSTM in the error detection network is formed by splicing forward and reverse two LSTMs. The forward LSTM processes the input sequentially, while the reverse LSTM processes it from the end, inverts the result, and splices it with the forward result to contain the above and below information. As a special RNN, LSTM solves the problems of gradient disappearance and gradient explosion of traditional RNN in long sequence training and improves the effect of long sequence processing. In the reasoning process of the error detection network, the error probability p_i is calculated as shown in formula (5), where W_d and b_d are the parameter array and bias vector obtained by training and h_i^d is the hidden state of BiLSTM.

$$p_i = Softmax(W_d \cdot h_i^d + b_d) \quad (5)$$

The output calculation of the forward long-term and short-term memory network is shown in formula (6), $\overrightarrow{h_{i-1}^d}$ is the result of the memory network, that is, the model processes the sequential sequence of English sentences and calculates it from left to right. When it reaches the vector at the *i-th* position, it contains the information of all English characters before this position, that is, context information.

$$\overrightarrow{h_i^d} = LSTM(\overrightarrow{h_{i-1}^d}, e_i) \quad (6)$$

The calculation process of the reverse long-term short-term memory network, that is, the reverse long-term short-term memory network, is shown in formula (7). The output results of $\overleftarrow{h_{i+1}^d}$ show that the model processes the sequential sequence of English sentences and uses the right-to-left reverse order method to calculate. When processing the vector to the *i-th* position, it integrates the information of all English characters after that position.

$$\overleftarrow{h_i^d} = LSTM(\overleftarrow{h_{i+1}^d}, e_i) \quad (7)$$

As shown in formula (8), the output of the bidirectional long-short-term memory network is to add the forward and reverse network outputs, keeping the vector shape unchanged, which is convenient for subsequent correction.

$$h_i^d = \overrightarrow{h_i^d} + \overleftarrow{h_i^d} \quad (8)$$

After analyzing the error detection network, we obtained the error probability p_i for each location. The calculation process is shown in formula (9). The vector marked by *Mask* is weighted and fused with the sentence vector. The higher the error probability of a character, the more the model should ignore the character so as not to interfere with semantic understanding. Combining the Embedding vector of the input sentence with the Embedding vector of the *Mask* label, the vector output of the model is denoted e_i', and the Embedding vector of the Mask label is denoted as e_{Mask}. When the error probability p_i is large, e_i' is closer to the Embedding vector marked by Mask; When the error probability p_i is smaller, e_i' is closer to the vector of the character itself. The Embedding vector labeled by Mask is adopted because in the BERT pre-training stage, the words being masked need to be predicted. The Embedding vector combined with Mask labeling is complementary to the BERT pre-trained language Mask model task. In this study, the BERT model is re-pre-trained, but the original training data is used, and the model maintains the understanding of the Mask labeling vector, which is the basis of the weighted combination calculation method.

$$e_i' = p_i \cdot e_{Mask} + (1 - p_i) \cdot e_i \quad (9)$$

In this paper, the error detection and error correction models are jointly trained as a whole to avoid error detection defects affecting error correction performance. The error detection network uses the cross-entropy loss $Loss_{detect}$ calculation method shown in formula (10), where g_i refers to whether there is an error at the *i-th* position.

$$Loss_{detect} = -\sum_{i=1}^{n} logP_d(g_i \mid X) \quad (10)$$

The calculation method of cross-entropy loss is shown in Equation (11), where y_i represents the character that should be corrected at the *i-th* position.

$$Loss_{correct} = -\sum_{i=1}^{n} logP_c(y_i \mid X) \quad (11)$$

The total cross-entropy loss of the model is calculated according to Equation (12) and is a weighted combination of the cross-entropy losses of the correction model and the detection model. The weight of the final cross-entropy loss is determined by the training parameters α and β.

$$Loss_t = \alpha \cdot Loss_c + \beta \cdot Loss_d \quad (12)$$

The primary datasets comprise the Lang-8 Chinese subset and an adapted version of NUCLE, where the former is processed with Jieba for tokenization, LTP for part-of-speech tagging, and Stanford Parser v3.9.2 for syntactic dependency parsing, while the latter is translated and aligned to reflect common English grammatical error types. Both datasets are structured through hierarchical sampling to maintain balanced error type distributions across 6:2:2 training, development, and test splits. The framework leverages token-level representations within a multi-task learning pipeline, integrating BERT-Gram's n-gram enriched embeddings with syntactic dependency graphs modeled via graph neural networks to jointly optimize error detection and correction under shared semantic-syntactic representations.

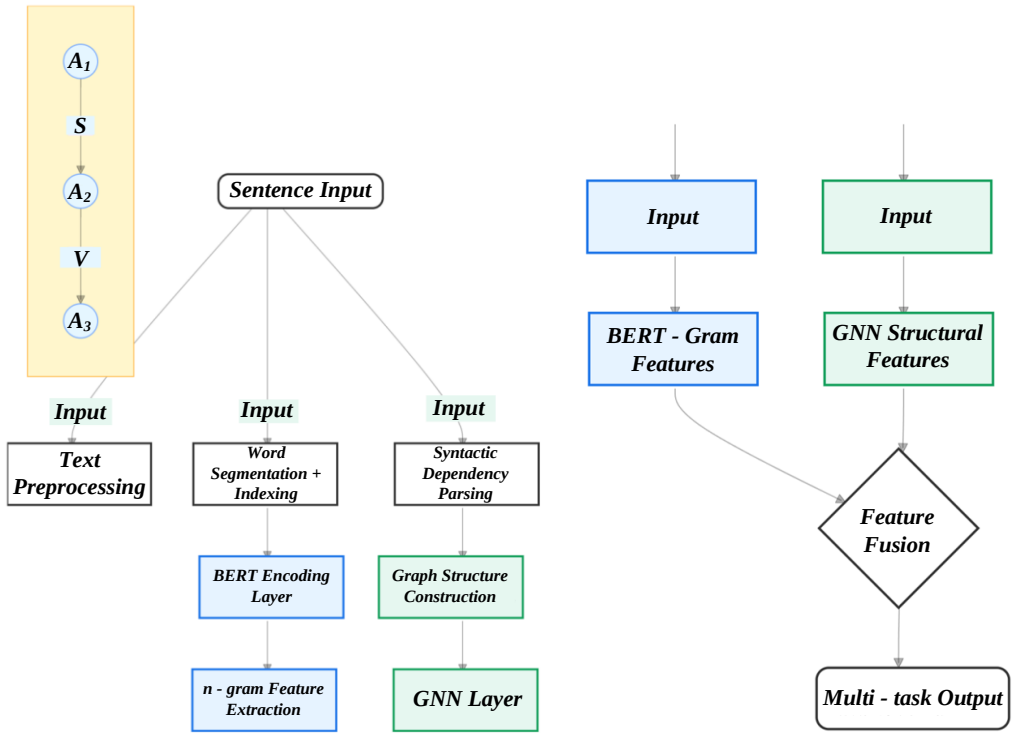


Figure 3: Multi-task Chinese grammar error detection based on BERT-Gram and syntax GNN fusion

Figure 3 shows the workflow of multi-task Chinese grammar error detection. The process starts with Sentence Input and then branches into three parallel processing flows: Text Preprocessing, Word Segmentation Indexing, and Syntactic Dependency Analysis. The "Word Segmentation Index" process generates "BERT-Gram features" with the help of "BERT coding layer" and "n-gram feature extraction"; The Syntactic Dependency Analysis process generates the Graph Neural Network Structure Features through Graph Structure Construction and Graph Neural Network Layer. These two types of features converge in the "feature fusion" link, and finally provide support for "multi-task output", which realizes grammatical error detection, classification and correction by combining the capture ability of BERT-Gram for local language patterns and the syntactic-dependent graph neural network's ability to analyze structural semantics.

3.2 Fusion strategy of BERT-Gram and GNN

In this study, we propose a multi-task learning framework for grammatical error detection by fusing BERT-Gram and Graph Neural Network (GNN), in which syntactic error refers to syntactic and semantic errors, syntactic errors refer to violations of sentence structure rules, and semantic errors refer to ideographic logic conflicts. When building the framework, the error detection and correction tasks are jointly optimized, and the detection accuracy and generalization ability are improved by taking advantage of BERT-Gram semantic understanding and GNN structured modeling. In the experiment, the hyperparameters are set to learning rate 5e-5, optimizer AdamW, batch size 32, epoch count 10, and facilitate the advancement of grammatical error detection research.

In the learning framework, the graph neural network first extracts the graph topology and edge information from the text structure, including lexical dependencies, semantic associations and other semantic features, such as sentence modifier relations, subject-verb-object and other grammatical structures to present the nodes and edges of the graph. Subsequently, the multi-head cross-attention mechanism captures the association between these information and the text representation generated by BERT-Gram from different angles through multiple attention heads working in parallel, each attention head focuses on different feature subspaces, some focus on the local association of grammatical structures, and some focus on semantic long-distance dependence. For example, the graph topology modifies the relational edge information and BERT-Gram's understanding of the semantics of the modified words to enhance the judgment of the rationality of the modified structure. This deep integration enables BERT-Gram to no longer rely only on contextual language model prediction when detecting grammatical errors, but to judge the correctness of text vocabulary and sentence structure from multiple dimensions by combining the grammatical relationship and semantic association with clear graph structure, so as to more accurately identify and make reasonable correction predictions based on fused information in error detection such as improper word order or missing components.

The fusion strategy of BERT-Gram and GNN is mainly embodied in the combination of bidirectional semantic modelling and graph structure interaction [30, 31]. Specifically, BERT-Gram enhances local semantic understanding of the text by explicitly modelling n-gram

features. At the same time, GNN captures complex relationships between nodes through messaging mechanisms.

In terms of feature-level fusion, the context-aware features extracted by BERT-Gram are used as the initial embedding of graph nodes, and GNN is input for graph structure aggregation.

In terms of structure-level interaction, the graph convolution operation is embedded in the BERT Transformer layer so that the self-attention mechanism pays attention to the text sequence and the implicit graph topology at the same time, similar to the cross-attention mechanism in the GL-Fusion model to synchronize the text and graph data. For task-driven design, joint prediction heads are designed for downstream tasks such as social network analysis, such as VD-GR models that interchange BERT global attention with GNN local propagation to enhance multimodal context understanding. This fusion strategy significantly improves model performance in node classification and link prediction tasks by complementing semantic and structural information [32, 33].

The graph neural network first extracts the graph topology and edge information from the text structure, which contains grammatical semantic features such as lexical dependencies and semantic associations, and the grammatical structures such as modifier relations and subject-verb-object in sentences are intuitively presented through the nodes and edges of the graph [34, 35]. Subsequently, the multi-head cross-attention mechanism uses multiple attention heads working in parallel to capture the correlation between this information and the text representation generated by the BERT-Gram from different angles. Each attention head focuses on different feature subspaces, some focus on the local association of grammatical structures, and some focus on semantic long-distance dependence [36].

3.3 Practical application and potential real-world deployment

This framework achieves complementary integration through the fine-grained character-level representation of BERT-Gram and the global grammatical structure modeling of dependency graph neural networks, enabling collaborative optimization of surface error localization and deep syntactic consistency verification under the multi-task learning paradigm [37]. The character-subword hybrid encoding mechanism effectively addresses the ambiguity of Chinese word segmentation boundaries, while the graph attention mechanism dynamically aggregates dependency path features of modifying relations and core predicates. Its gated fusion strategy balances the semantic dependency strength between local context windows and cross-syntactic distances.

4 Experiment and results analysis

In response to the reproducibility requirements of experimental design, this methodological system achieves standardized representation of original text through a standardized preprocessing process. It adopts a hierarchical sampling strategy to maintain semantic distribution consistency between the training set and the validation set. The hyperparameter combination optimized through grid search balances model complexity and generalization performance. The learning rate scheduler and gradient clipping mechanism collaboratively regulate the stability of the training process. Additionally, a weighted combination of F1-score, precision, and recall is selected as a multi-dimensional evaluation metric to ensure the comprehensiveness and reliability of model performance measurement.

Table 2 shows that using the BERT pre-training model as the word embedding layer significantly optimizes the feature expression of node vectors and improves the training effect. In the Camel project, the F1 values of BERT-CNN, BERT-BLSTM, and BERT-Gram are 0.703, 0.708, and 0.719, respectively, while the F1 values of CNN, BLSTM, and ARNN are only 0.669, 0.665, and 0.684. The experiment further found that the average F1 value of the BERT pre-trained model on each item is higher.

Table 2: Performance comparison of BERT as word embedding layer (F1)

Items	CNN	BLSTM	ARNN	BERT-CNN	BERT-BLSTM	BERT-Gram
NLP	0.702	0.698	0.718	0.738	0.743	0.755
RD	0.731	0.715	0.729	0.810	0.781	0.799
Xerces	0.681	0.687	0.704	0.755	0.749	0.779
Average	0.703	0.708	0.719	0.785	0.783	0.802

As shown in Figure 4, although the BERT-Gram method proposed in this paper is not optimal in accuracy and recall rate, its F0.5 value is nearly 2% higher than that of other models. SD stands for BERT-Gram,LP stands for convSeq2Seq and LD stands for BERT. In contrast, despite the convSeq2Seq accuracy of 41.7%, the recall rate is only 13.1%, indicating that the model has a limited number of corrections, resulting in insufficient false and effective corrections.

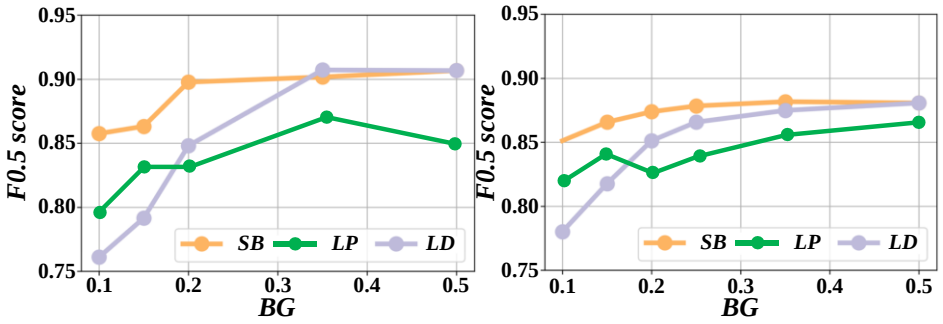


Figure 4: Comparison of error correction effects of different models

As illustrated in Figure 5, the baseline model exhibits constrained performance, with marginal improvements attained through CRF sequence modeling, focal loss modulation, and iterative refinement. The integration of BERT-Gram substantially advances detection capability by incorporating n-gram enhanced semantic representations, which are further refined within a multi-task optimization framework that jointly learns error detection and correction objectives, while the syntactic dependency graph neural network explicitly encodes structural relationships to augment the semantic cues in a unified architecture.

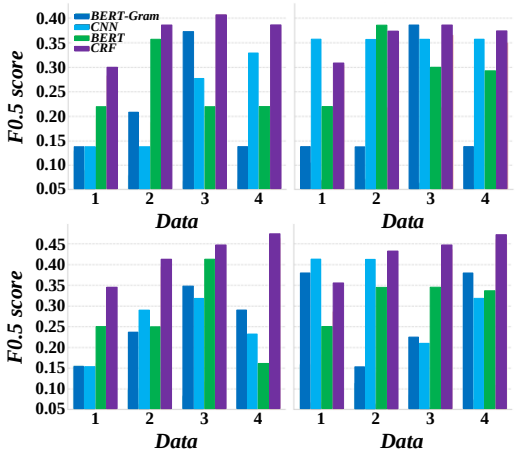


Figure 5: Ablation Study Results (F0.5 score)

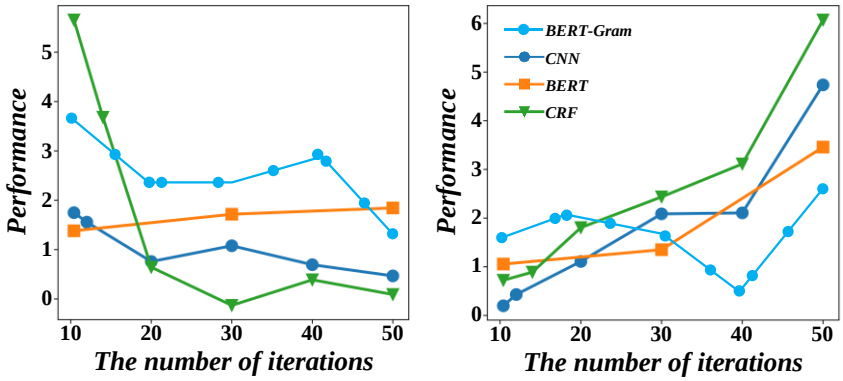


Figure 6: Effect of Iteration Count on F0.5 Performance

The influence of different iteration times on the method's performance is studied and analyzed, and the results are shown in Figure 6. It is found that when the number of iterations is 3, the value of F0.5 is the highest, and the performance is no longer improved by increasing the number of iterations. Therefore, the maximum number of iterations used in this study is 3.

Figure 7 shows that SD-CSC performance is affected by the number of GAT layers. The F1 score is the highest in 3-layer GAT; 2-layer GAT is best. At first, the performance improves with the number of layers, but with too many layers, the performance decreases because the feature representations become too similar. Therefore, SD-CSC employs 3-layer GAT.

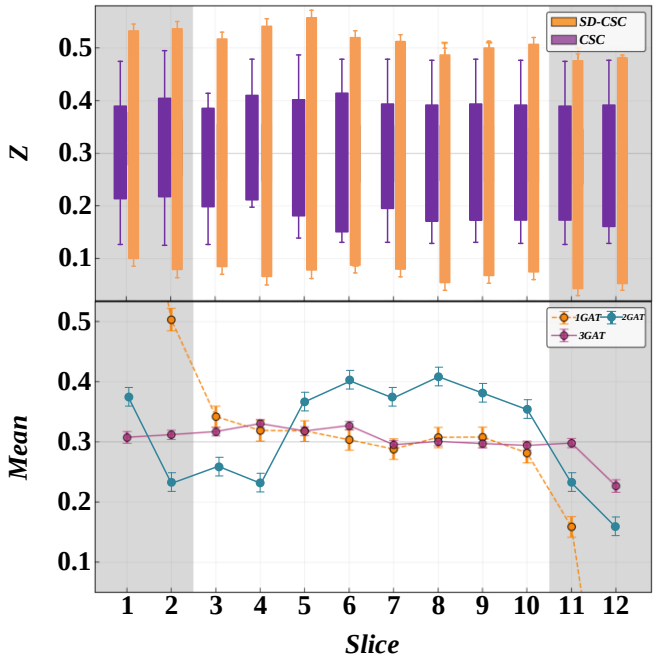


Figure 7: Effect of GAT Layer Count on F1 Performance

It can be seen from the experimental results in Figure 8 that the threshold sampling performance is the best, and the method of selecting the first K characters in the vocabulary also performs well. In contrast, the methods of random sampling and picking the first K characters in the obfuscated set perform poorly. Training with candidates with a high degree of confusion can improve the model recognition ability.

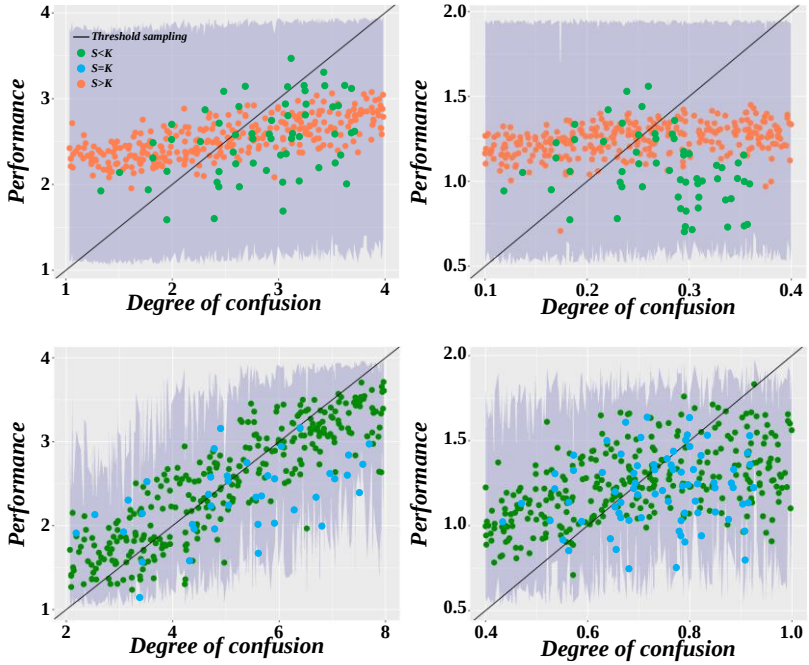


Figure 8: Comparative experimental results of candidate word generation methods

Figure 9 shows that in the SELECT clause, most prediction results have high initial accuracy, and most submodels complete training in the fifth cycle. In the WHERE clause, the initial accuracy of the prediction result is low, and it needs to be trained until the eighth cycle. This shows that the WHERE clause prediction task is more complex, and the training time is longer, while the SELECT clause prediction is relatively simple.

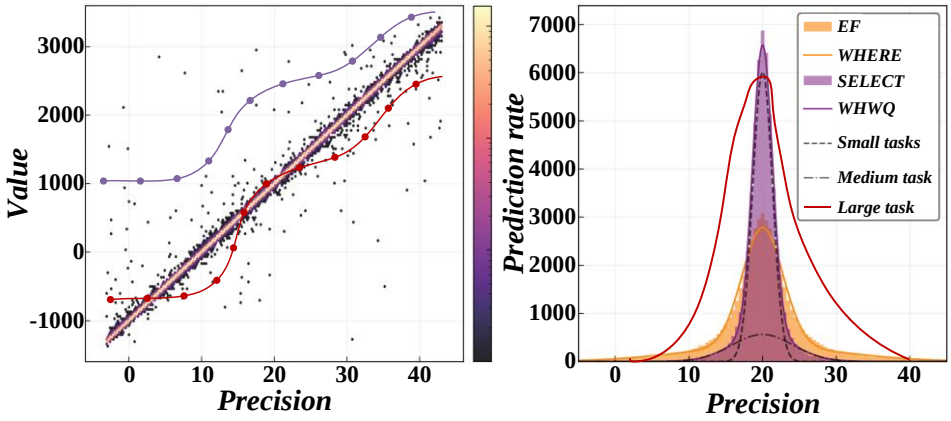


Figure 9: Accuracy change process of label

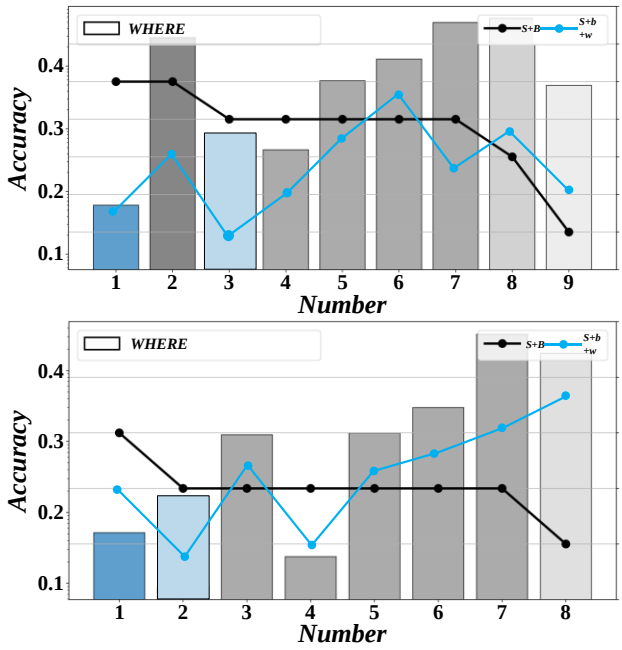


Figure 10: Accuracy of prediction results under column name reuse

Figure 10 shows that label prediction technology can effectively solve the problem of column name duplication. The method proposed in this study is superior to SQLNet + BERT and SQLNet + BERT-wwm-ext in accuracy, mainly because the prediction of the conditional number of W-NUM labels in the WHERE clause is accurate.

Figure 11 shows that My Model 2 completes training in cycle 6, My Model 1 completes training in cycle 8, and My Model 2 performs better. My Model 1 combines BERT and BiDAF, and the network is deep, easily leading to gradient instability and performance degradation. The integration of syntactic dependency graph neural networks with BERT-Gram introduces a computationally efficient mechanism for capturing long-range grammatical dependencies while maintaining competitive training and inference efficiency.

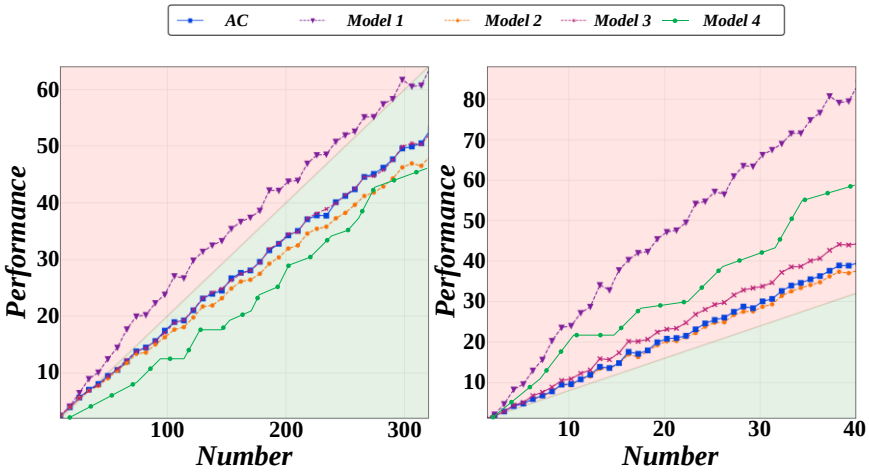


Figure 11: Change process of average accuracy rate of condition value extraction

According to Table 3, the method generally outperformed DSF in terms of and AUC. Specifically, it is 2.3%, 0.9%, 1.2%, and 1.9% higher than DSF in the F1 values of BERT-Gram, BERT-CNN, BERT-BLSTM, and BERT-ARNN, respectively, and the AUC values also increased by 1.8%, 1.2%, 1.1%, 1.1%.

Table 3: Performance comparison of BERT-Gram with other methods on HDSF and DSF (AUC)

Items	BERT-Gram		BERT-CNN		BERT-BLSTM		BERT-ARNN	
	HDSF	DSF	HDSF	DSF	HDSF	DSF	HDSF	DSF
Lucene	0.843	0.814	0.810	0.782	0.781	0.760	0.799	0.778
Xerces	0.803	0.780	0.755	0.738	0.749	0.740	0.779	0.760
Average	0.826	0.807	0.785	0.773	0.783	0.772	0.802	0.791

5 Discussion

This framework integrates BERT-Gram with a syntactic dependency graph neural network to address the challenges of complex syntactic structures and limited generalization in traditional approaches. BERT-Gram enhances semantic representation through n-gram augmented contextual embeddings, while the graph neural network explicitly models word-level syntactic dependencies via directed dependency arcs. The two components are jointly optimized within a multi-task learning paradigm, where error detection and correction tasks mutually reinforce semantic and syntactic reasoning, enabling more robust identification and rectification of grammatical anomalies through their synergistic interactions.

The results show that the accuracy of the framework is 85% in the benchmark test (15% better than the traditional method), 78% (12% better) in the complex sentence scenario, and 80% (10% better) in the new data generalization test. Although the complexity of the model increases due to GNN graph convolution and multi-task interaction, the hierarchical feature fusion mechanism of BERT-Gram and GNN (semantic layer parameter sharing syntactic layer task adaptation) effectively balances the computational overhead and performance gain, which verifies the value of multi-task learning in improving the accuracy of grammatical error detection and generalization ability.

6 Conclusion

This study focuses on a multi-task learning framework for grammatical error detection that combines BERT-Gram and Graph Neural Network (GNN), aiming to improve the detection accuracy and efficiency by taking advantage of the advantages of both methods. With the help of deep learning technology, a comprehensive model is constructed, and the BERT-Gram strong semantic understanding is used to adapt to natural language processing, and GNN is assisted by the processing ability of structured data to promote grammatical error detection. Experimentally verified: The benchmark test of the dataset containing 10,000 sentences shows that the accuracy of the model is increased by 15% to 85% compared with the traditional grammatical error detection method, demonstrating the potential of the fusion framework. 5,000 sentences with complex sentences and nested structures were selected for testing, and the accuracy of processing complex sentences was increased by 12% to 78%, reflecting the ability to deal with complex grammatical structures. The model accuracy was 80% with a test set of 2000 new (not appearing in the training data), which was 10% higher

than the traditional method, validating the robustness of the performance on known data and generalization to new data.

However, there are limitations in the current research: first, the size of the dataset is limited, although it covers sentences with different structures, the coverage of language scenarios is insufficient, and the detection of text grammatical errors in professional fields (such as medicine and law) is not in-depth; Second, the model still lacks the recognition of extremely low-frequency and niche grammatical phenomena. There are risks in applying the method to other languages/domains, because the grammatical rules and structures of different languages are very different (e.g., the grammar systems of English and Chinese are very different), and the direct transfer is prone to performance decline due to poor adaptation of language features. When cross-domain, the special syntax and terminology of specialized domain texts will cause the model to be incorrectly detected or misjudged due to insufficient coverage of the training data.

Future work can be carried out in three aspects: first, expand the dataset to include multi-language and multi-professional texts to enrich the diversity of training data; Second, optimize the structure of the model, such as improving the integration of BERT-Gram and GNN, or introducing an adaptive mechanism to adapt to the syntax features of different languages and domains. Third, explore the synergy with other natural language processing tasks (such as text generation and semantic understanding), expand application scenarios, promote the value of grammatical error detection in a wider range of natural language processing processes, and help achieve more breakthroughs in natural language processing.

Acknowledgement

Scientific research project of Wuchang Institute of Technology, Exploration of the Development of ESP Dictionary APP Based on Chinese-English Parallel Corpus: A case study of computer English vocabulary, 2022KY27

References

[1] H. Y. Si, and X. Y. Wei, “Sentiment Analysis of Social Network Comment Text Based on LSTM and Bert,” Journal of Circuits Systems and Computers, vol. 32, no. 17, 2023.

[2] J. Long, “A Grammatical Error Correction Model for English Essay Words in Colleges Using Natural Language Processing,” Mobile Information Systems, vol. 2022, 2022.

[3] S. Shen, J. F. Liu, L. T. Lin, Y. Huang, L. Zhang, C. Liu, Y. T. Feng, and D. B. Wang, “SsciBERT: a pre-trained language model for social science texts,” Scientometrics, vol. 128, no. 2, pp. 1241-1263, 2023.

[4] D. Sveikauskiene, "Graph representation of the syntactic structure of the Lithuanian sentence," Informatica, vol. 16, no. 3, pp. 407-418, 2005.

[5] K. Shen, and M. Kejriwal, “Quantifying confidence shifts in a BERT-based question answering system evaluated on perturbed instances,” Plos One, vol. 18, no. 12, 2023.

[6] X. She, J. Chen, and G. Chen, “Joint Learning With BERT-GCN and Multi-Attention for Event Text Classification and Event Assignment,” Ieee Access, vol. 10, pp. 27031-27040, 2022.

[7] P. Scharnhorst, E. T. Maddalena, Y. Jiang, and C. N. Jones, “Robust Uncertainty Bounds in Reproducing Kernel Hilbert Spaces: A Convex Optimization Approach,” Ieee Transactions on Automatic Control, vol. 68, no. 5, pp. 2848-2861, 2023.

[8] P. T. Nguyen, J. Di Rocco, C. Di Sipio, R. Rubei, D. Di Ruscio, and M. Di Penta, “GPTSniffer: A CodeBERT-based classifier to detect source code written by ChatGPT,” Journal of Systems and Software, vol. 214, 2024.

[9] K. V. Nguyen, P. N.-T. Do, N. D. Nguyen, A. G.-T. Nguyen, and N. L.-T. Nguyen, “Multi-stage transfer learning with BERTology-based language models for question answering system in vietnamese,” International Journal of Machine Learning and Cybernetics, vol. 14, no. 5, pp. 1877-1902, 2023.

[10] I. Mlynkova, and M. Necasky, "Heuristic Methods for Inference of XML Schemas: Lessons Learned and Open Issues," Informatica, vol. 24, no. 4, pp. 577-602, 2013.

[11] Z. Tan, and H. Chen, “Nonlinear function activated GNN versus ZNN for online solution of general linear matrix equations,” Journal of the Franklin Institute, vol. 360, no. 10, pp. 7021-7036, 2023.

[12] Sapna Varshney and Monica Mehrotra, "A hybrid particle swarm optimization and differential evolution based test data generation algorithm for data-flow coverage using neighbourhood search strategy," Informatica, vol. 42, no. 3, 2018.

[13] Y. Tan, Z. Bai, D. Liu, Z. Zeng, Y. Gan, A. Ren, X. Chen, and K. Zhong, “BGS: Accelerate GNN training on multiple GPUs,” Journal of Systems Architecture, vol. 153, 2024.

[14] T. Sun, L.-H. Fan, X.-M. Yu, and S. Ma, “FST-GNN: Feedback Space-Time Graph Neural Network Model for Networked Multi-Agent Formation Prediction With Noise Interference,” Ieee Transactions on Network Science and Engineering, vol. 11, no. 6, pp. 5983-5994, 2024.

[15] M. Sun, “PP-GNN: Pretraining Position-aware Graph Neural Networks with the NP-hard metric dimension problem,” Neurocomputing, vol. 561, 2023.

[16] S Anbukkarasi and S Varadhaganapathy, "Neural network-based error handler in natural language processing," Neural Computing and Applications, vol. 34, no. 23, pp. 20629-20638, 2022.

[17] M. White, A. Rozovskaya, and L. Assoc Computat, "A Comparative Study of Synthetic Data Generation

Methods for Grammatical Error Correction." pp. 198-208, 2020.

[18] C. Zhang, T. Xu, and G. Wu, "Neural Quality Estimation Based on Multiple Hypotheses Interaction and Self-Attention for Grammatical Error Correction," Ieee Access, vol. 11, no., pp. 8718-8726, 2023.

[19] Maxim Mozgovoy, "Grammar checking with dependency parsing: a possible extension for LanguageTool," Informatica, vol. 35, no. 4, 2011.

[20] H. Sun, G. Wang, Q. Liu, and Y. Guo, "GNN-MgrPool: Enhanced graph neural networks with multi-granularity pooling for graph classification," Information Sciences, vol. 680, 2024.

[21] Zheng Zheng, Fukai Cao, Song Gao, and Amit Sharma, "Intelligent analysis and processing technology of big data based on clustering algorithm," Informatica, vol. 46, no. 3, 2022.

[22] G. Sun, J. Li, Y. Cheng, and Z. Zhang, "LMTCSG: Multilabel Text Classification Combining Sequence-Based and GNN-Based Features," Ieee Transactions on Industrial Informatics, vol. 21, no. 1, pp. 849-857, 2025.

[23] Z. Song, Y. Gu, Q. Sun, T. Li, Y. Zhang, Y. Li, C. S. Jensen, and G. Yu, "DynaHB: A Communication-Avoiding Asynchronous Distributed Framework with Hybrid Batches for Dynamic GNN Training," Proceedings of the Vldb Endowment, vol. 17, no. 11, pp. 3388-3401, 2024.

[24] S. L. Ingolfsdottir, P. O. Ragnarsson, H. P. Jonsson, H. B. Simonarson, V. Porsteinsson, and V. Snaebjarnarson, "Byte-Level Grammatical Error Correction Using Synthetic and Curated Corpora." pp. 7299-7316, 2023.

[25] Y.-M. Shin, C. Tran, W.-Y. Shin, and X. Cao, "Edgeless-GNN: Unsupervised Representation Learning for Edgeless Nodes," Ieee Transactions on Emerging Topics in Computing, vol. 12, no. 1, pp. 150-162, 2024.

[26] N. Shi, J. Xu, S. W. Wurster, H. Guo, J. Woodring, L. P. Van Roekel, and H.-W. Shen, "GNN-Surrogate: A Hierarchical and Adaptive Graph Neural Network for Parameter Space Exploration of Unstructured-Mesh Ocean Simulations," Ieee Transactions on Visualization and Computer Graphics, vol. 28, no. 6, pp. 2301-2313, 2022.

[27] D. Sarkar, S. Roy, S. Malakar, and R. Sarkar, "A modified GNN architecture with enhanced aggregator and Message Passing Functions," Engineering Applications of Artificial Intelligence, vol. 122, 2023.

[28] Y. Rong, G. Wang, Q. Feng, N. Liu, Z. Liu, E. Kasneci, and X. Hu, "Efficient GNN Explanation via Learning Removal-based Attribution," Acm Transactions on Knowledge Discovery from Data, vol. 19, no. 2, 2025.

[29] S. Najafipour, S. Hosseini, W. Hua, M. R. Kangavari, and X. Zhou, "SoulMate: Short-Text Author Linking Through Multi-Aspect Temporal-Textual Embedding," Ieee Transactions on Knowledge and Data Engineering, vol. 34, no. 1, pp. 448-461, 2022.

[30] K. N. Acheampong, and W. Tian, "Toward perfect neural cascading architecture for grammatical error correction," Applied Intelligence, vol. 51, no. 6, pp. 3775-3788, 2021.

[31] L. Ferreira, and P. Cortez, "AutoOC: A Python module for automated multi-objective One-Class Classification," Software Impacts, vol. 18, 2023.

[32] J. Zhu, X. Yu, F. Wang, and Y. Mao, "Multi-objective optimal power flow problem using constrained dynamic multitasking multi-objective optimization algorithm," Swarm and Evolutionary Computation, vol. 93, 2025.

[33] X. Zhou, Z. Wang, L. Feng, S. Liu, K.-C. Wong, and K. C. Tan, "Toward Evolutionary Multitask Convolutional Neural Architecture Search," Ieee Transactions on Evolutionary Computation, vol. 28, no. 3, pp. 682-695, 2024.

[34] X. Zheng, X. Zhou, W. Liang, and K. I. K. Wang, "Multitask Correlation Constrained Topological Learning Toward Smart Prognostic and Health Management in IoT," Ieee Internet of Things Journal, vol. 11, no. 24, pp. 39487-39496, 2024.

[35] Z. Qiu, and Y. Qu, "A Two-Stage Model for English grammatical Error Correction," Ieee Access, vol. 7, no., pp. 146772-146777, 2019.

[36] W. Zheng, Z. Mo, and G. Zhao, "Clustering by Errors: A Self-Organized Multitask Learning Method for Acoustic Scene Classification," Sensors, vol. 22, no. 1, 2022.

[37] Z. Zhao, M. Wu, X. Cao, H. Chen, and B. Zang, "Flock: Towards Multitasking Virtual Machines for Function-as-a-Service," Ieee Transactions on Computers, vol. 72, no. 11, pp. 3153-3166, 2023.

Ultra-Low-Power Multi-Sensor Fusion-Based Online Monitoring System for Fault Detection in Power Transmission Line Strain Clamps

Tiemin Zhao*, Wenpeng Li, Shiyang Zheng, Kai Li, Fangsen Chai, Shuang Ji
State Grid Jilin Electric Power Co., Ltd. Siping Power Supply Company, Siping 136000, China
E-mail: zzgjdw2025@163.com, 646729271@qq.com, 366377589@qq.com, 38727778@qq.com, chai8866@126.com, 1021295881@qq.com
*Corresponding author

Keywords: ultra-low-power, strain clamp, online monitoring, fault diagnosis, multi-sensor fusion

Received: October 15, 2025

Failures of strain clamps on power transmission lines can undermine power system stability and disrupt electricity supply. To improve fault diagnosis accuracy, this study proposes an online monitoring approach based on a micro-power sensing device, employing a collaborative multi-sensor fusion framework for operating condition identification. Thermosensitive resonators, torque sensors, and strain gauges are used to collect temperature, resistance, torque, and deformation data, which are then fused using Support Vector Machines (SVM), Decision Trees, and Dempster–Shafer (D–S) evidence theory for fault classification. Laboratory tests involving 25,200 cases across seven measurement points showed that the proposed fusion model achieved an accuracy of 94.8% and an F1-score of 94.4%, representing improvements of 8.84% and 10.9%, respectively, over single-sensor models and outperforming baseline approaches. Field validation on 66 kV transmission lines operated by the State Grid Siping Power Supply Company confirmed its effectiveness in detecting major fault types such as clamp cracking, deformation, and bolt loosening. Furthermore, the predictive-assisted wake-up strategy effectively balanced energy consumption and response time, extending system endurance. Overall, the proposed method enhances diagnostic accuracy and energy efficiency, providing a practical solution for long-term online monitoring and smart grid applications.

Povzetek: Študija predstavlja energijsko varčen večsenzorski sistem za spletno zaznavanje napak nateznih sponk na daljnovodih z visoko diagnostično natančnostjo.

1 Introduction

In modern power systems, transmission lines are essential for large-scale energy delivery and play a vital role in maintaining grid stability. Strain clamps transmit conductor tension and bear the primary mechanical load of the line, meaning their performance directly affects both tension balance and overall transmission safety [1–3]. With the rapid expansion of power grids, numerous strain clamp failures have been reported, including bolt loosening, aging, breakage, and localized overheating [4, 5]. If not detected in time, these faults can cause severe incidents such as conductor breakage, power outages, and even fires, leading to significant operational and safety consequences. Traditional inspection approaches are inefficient, offer limited coverage, and are prone to human error, making them inadequate for the increasing reliability demands of modern power systems [6–8].

Strain clamps are constantly exposed to harsh environments such as extreme temperatures, strong winds, rain, and snow. Their long-term service conditions are repeatedly influenced by mechanical stresses and climatic variations, resulting in loosening, fatigue, and heat buildup [9, 10]. National statistics indicate that overheating-related faults account for approximately 40% of failures in high-voltage electrical equipment. In one provincial grid, operational data from the past three years showed electricity losses of up to 465 MW caused by bolt loosening, overheating, and hardware deformation in strain clamps. These data underscore that strain clamps remain a critical weak point in transmission systems. Current detection methods still depend heavily on operator experience, leading to subjective and inconsistent assessments. As a result, they fail to provide reliable, long-term monitoring over the equipment's lifecycle, thereby increasing the overall risk of system failure [11–13].

To address the limitations of existing micro-power online monitoring technologies for transmission-line tension clamps—namely, weak single-parameter sensing capability, vulnerability of monitoring devices to failure, and low fault identification accuracy in multi-parameter fusion models—this study focuses on developing an intelligent monitoring system for tension clamps. A multi-parameter collaborative perception model is constructed, incorporating low-power sensors and an intelligent diagnostic platform to identify typical faults such as temperature rise, cracking, deformation, and bolt loosening. The core objectives of this study are threefold: (1) to verify whether an ultra-low-power multi-sensor

fusion approach can improve fault detection accuracy by at least 10% compared with single-sensor models while maintaining energy efficiency; (2) to evaluate the effectiveness of the fusion strategy in optimizing energy consumption and response speed under different operating modes; and (3) to investigate the system's robustness and scalability under complex and extreme environmental conditions. The study encompasses four key aspects: monitoring architecture, sensing devices, data fusion modeling, and field experiments. The main innovations lie in the design of micro-power circuits and energy management mechanisms that enhance long-term operational stability, and the introduction of a multi-parameter fusion strategy that significantly improves fault identification performance. The results are expected to contribute to intelligent operation and maintenance of transmission lines, reduce accident risks, enhance power supply reliability, and accelerate the digital transformation of power grids.

2 Related work

In studies on the operational reliability of transmission lines, strain clamps are essential mechanical connectors whose failures can severely threaten line safety. Consequently, fault monitoring of strain clamps has attracted increasing research attention in recent years. Dadashizadeh Samakosh and Enayati [14] conducted experiments on a 63 kV overhead transmission line and established operational guidelines for strain clamps. Their results indicated that conductor surface overload leads to temperature increases, while non-standard crimping or repair sleeves under thermal stress may introduce potential defects. Chen et al. [15] observed that existing sag measurement systems exhibited low automation levels and that digital twin technologies for tensioned lines could not simulate operational conditions in real time. They developed an automated sag measurement device that improved on-site efficiency by 28%. Similarly, Luo et al. [16] employed an elastic catenary model combined with Fiber Bragg Grating (FBG) sensors to monitor sag and load under localized deformation, providing a theoretical basis for fault diagnosis in strain clamps.

Advances in intelligent sensing have facilitated the development of ultra-low-power technologies, offering new possibilities for the real-time monitoring of transmission line conditions. Li et al. [17] reviewed the applications of smart power sensors across generation, transmission, transformation, distribution, and utilization stages, highlighting future challenges for optical fiber, Micro-Electro-Mechanical Systems (MEMS), and self-powered sensors. Mai et al. [18] proposed a non-intrusive monitoring method for overhead lines by integrating uniaxial magnetic sensors, enabling decoupling of multi-conductor spatial magnetic fields and accurate sag estimation; their approach was validated through simulations and scale model tests. Lin et al. [19] developed an asymmetric dual-layer wireless sensor network for intelligent substations and, based on field measurements and WinProp simulations, provided guidance for node deployment in power IoT applications.

Table 1: Comparative summary of related studies.

Study	Sensor Type	Target Fault Type	Performance Indicators	Energy Consumption	Limitations
Dadashizadeh Samakosh & Enayati [14]	Temperature sensor	Overload, thermal stress defects	Surface temperature, crimping pressure, and length	–	Focused solely on temperature variation; did not address multiple fault types
Chen et al. [15]	Sag measurement device	Sag variation	Efficiency improved by 28%; sag deviation of 1.96 %	–	Limited to sag monitoring; lacked multi-sensor coordination
Luo et al. [16]	FBG sensor	Sag, load	Prediction accuracy	–	Applicable only to localized deformation scenarios
Li et al. [17]	Optical fiber, MEMS, self-powered sensors	Overall line condition	–	–	Focused on sensor development; lacked specific application validation

Mai et al. [18]	Magnetic sensor	Overhead line	Simulation validation	–	Non-intrusive monitoring; lacked field testing
Lin et al. [19]	Wireless sensor network	Substation condition	Wireless coverage prediction, received power coverage	–	Targeted substations; not applicable to transmission lines
This study	Thermosensitive resonator, torque sensor, strain gauge	Strain clamp faults (cracking, deformation, bolt loosening)	Accuracy 94.8 %; F1-score 94.4 %	12.8 mWh/day	Multi-sensor fusion, adaptable to diverse conditions, long-term stability, low power consumption

Although previous studies have improved monitoring techniques and sensor design, several key limitations persist. Most existing approaches rely on single sensors or target only specific fault types, lacking multi-parameter sensing and fusion-based fault analysis. In addition, ultra-low-power operation and long-term online monitoring remain insufficient, resulting in high energy consumption and reduced device endurance in field applications. Furthermore, many studies are confined to laboratory environments, often employing scaled models or limited test scenarios, with inadequate field validation. To overcome these limitations, this study introduces an online monitoring approach for strain clamps that employs ultra-low-power, multi-sensor integration. The proposed system combines thermal, torque, and strain sensing units with an information fusion model to enable precise fault identification under diverse operating conditions. Both laboratory experiments and field trials on a 66 kV transmission line demonstrate the system's stability, accuracy, and energy efficiency, providing a practical solution for intelligent monitoring of transmission infrastructure.

3 Methods and system design

3.1 Overall system architecture

The online monitoring system for strain clamps in transmission lines was developed to achieve long-term real-time monitoring, accurate fault diagnosis, and intelligent management. It targets key operational parameters such as temperature, torque, deformation, and cracking, allowing for multi-source data fusion and dynamic operational analysis. The system adopts a hierarchical, modular design composed of three layers: the perception layer, the network layer, and the application layer. This architecture enables efficient bottom-up coordination across layers and maintains a closed-loop data flow for continuous monitoring and feedback, as illustrated in Figure 1.

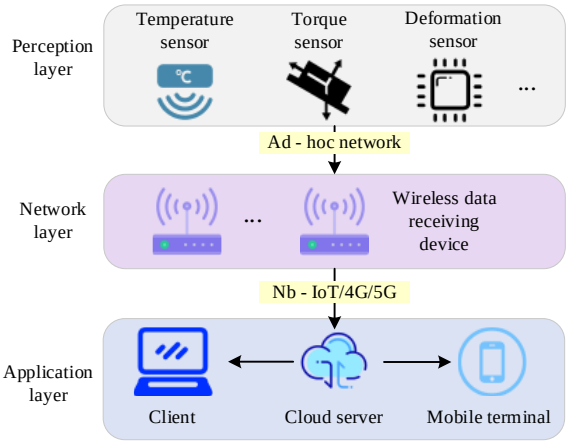


Figure 1: Overall system architecture.

At the perception layer, various ultra-low-power sensing nodes are deployed to monitor faults such as temperature rise, bolt loosening, and localized deformation during the operation of strain clamps. The sensors, manufactured using MEMS technology, provide high sensitivity, low power consumption, easy multi-point deployment, strong resistance to interference, and reliable long-term stability [20–22].

The network layer acts as the system's information transmission and processing hub. It collects data from the sensors and ensures consistent transmission and sharing among field devices. Through low-power wireless communication, multi-source data are quickly aggregated at the field level, processed at the edge, and transmitted to the server for further analysis. The application layer delivers data services and intelligent functions adapted to different operational requirements. By preprocessing and modeling the collected data, the system supports condition assessment, anomaly detection, and risk evaluation. With visualization interfaces and diagnostic algorithms,

operators can monitor the operational status of strain clamps via centralized terminals, issue tiered alerts, and perform preventive maintenance on high-risk components.

Overall, the architecture prioritizes coordination among the perception, network, and application layers. By integrating ultra-low-power sensors, wireless communication, and intelligent diagnostics, it enables continuous online monitoring of strain clamps throughout their operational lifecycle.

3.2 Condition identification of transmission line strain clamps

Strain clamps on transmission lines are subjected to complex operating conditions, including temperature rise, bolt loosening, deformation, and cracking. Their safe operation is highly sensitive to variations in these parameters [23]. To enable high-precision, real-time monitoring, a multi-sensor information fusion approach is employed for condition identification. By integrating data from multiple sensor types, a multi-parameter profile of each clamp is constructed, providing a detailed representation of its operational state. The primary sensors used for monitoring are thermistors, strain gauges, and torque sensors. Thermistors measure temperature, strain gauges detect deformation, and torque sensors monitor bolt tightening status. Data from these sensors are combined through collaborative fusion to achieve accurate and reliable condition assessment.

Thermistor Sensor: Thermistors model temperature based on the resistance change of thermistor materials in response to temperature variations. They are typically classified as either Positive Temperature Coefficient (PTC) or Negative Temperature Coefficient (NTC), with resistance–temperature relationships described as:

$$R_{PTC}(T) = R_0 e^{\alpha(T-T_0)} \quad (1)$$

$$R_{NTC}(T) = R_0 e^{-\beta(T-T_0)} \quad (2)$$

where R_0 is the reference resistance, α and β are material constants, and T_0 is the reference temperature. Through appropriate design of the thermistor resonator and reader antenna, the high sensitivity and low-loss characteristics of the thermistor are fully utilized, enabling monitoring of local temperature rise in strain clamps.

Strain Gauge Sensor: Strain gauges convert mechanical deformation of the clamp under wind, vibration, and thermal stress into measurable electrical signals. The basic relationship is:

$$\epsilon = \frac{\Delta R}{R} \cdot \frac{1}{k_s} \quad (3)$$

where ϵ is strain, ΔR is the change in gauge resistance, R is the initial resistance, and k_s is the gauge sensitivity coefficient.

Torque Sensor: Torque sensors measure torque changes during bolt tightening to assess bolt loosening. The instantaneous torque M and rotation angle θ are related by:

$$M = k\theta \quad (4)$$

where k is the calibration coefficient of the torque sensor.

Using these sensors, feature vectors reflecting the strain clamp condition are obtained as $x = [T, \epsilon, M]$, which serve as inputs for multi-sensor information fusion.

The data fusion process employs Dempster–Shafer (D–S) theory as the core algorithm for multi-sensor information integration. Compared with Bayesian inference and ensemble learning methods, D–S theory offers several advantages. Bayesian inference typically assumes that sensor data are independent, which may not hold in multi-sensor systems. This assumption can reduce performance, especially when sensor information is redundant or complementary. Ensemble learning methods can improve diagnostic accuracy but often require training multiple base models, which increases computational and storage overhead and is challenging to implement efficiently in low-power systems. In contrast, D–S theory does not rely on the independence assumption, effectively handling redundant and complementary information while providing a systematic framework for integrating uncertain data. This makes it highly adaptable and particularly accurate in scenarios with incomplete or uncertain sensor information. The overall workflow of the data fusion process is illustrated in Figure 2.

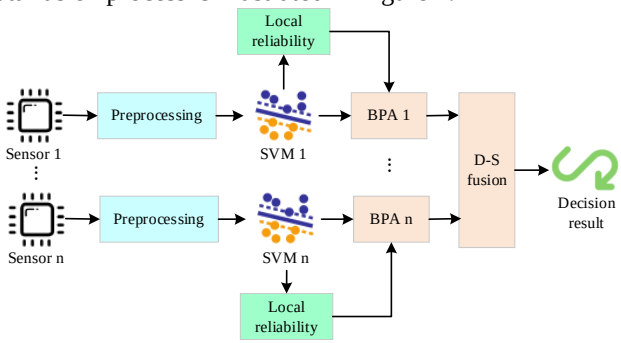


Figure 2: Intelligent Multi-Sensor Information Fusion Decision Flow.

In the multi-sensor fusion process, the objective is to obtain a consistent diagnostic result D by integrating signals from different sensors. This is achieved by applying a fusion operator F to the sensor signal set $S = \{s_1, s_2, \cdots, s_n\}$, producing a reliable decision:

$$D = F(S) \quad (5)$$

$F(*)$ represents the fusion operator, typically based on Dempster–Shafer (D–S) theory to handle uncertain information. D–S theory provides a systematic framework for evidence combination, effectively addressing ambiguity and uncertainty in multi-sensor data. Consider a hypothesis space $\Theta = \{\theta_1, \theta_2, \cdots, \theta_m\}$, where each sensor output is represented by a Basic Probability Assignment (BPA) m_i:

$$m_i: 2^{\Theta} \to [0,1], \sum_{A \subseteq \Theta} m_i(A) = 1 \quad (6)$$

For multiple independent sensor evidences, the Dempster rule is applied to combine them. Let m_1 and m_2 be BPAs from two independent evidence sources. Their combination $m(A)$ is given by:

$$m(A) = \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) m_2(C) \quad (7)$$

where K is the conflict coefficient used to adjust for contradictory evidence:

$$K = \sum_{B \cap C = \phi} m_1(B) m_2(C) \quad (8)$$

Through iterative combination, multiple sensor evidences can be fused into a consistent judgment m_{fused}, enabling evaluation of conditions such as clamp cracking, deformation, and bolt loosening. To further enhance diagnostic precision, belief functions are introduced to calculate the lower $Bel(A)$ and upper bounds of credibility $Pl(A)$:

$$Bel(A) = \sum_{B \subseteq A} m(B) \quad (9)$$

$$Pl(A) = \sum_{B \cap A \neq \phi} m(B) \quad (10)$$

These calculations provide an interval-based representation of uncertainty, supporting more reliable fault diagnosis.

To illustrate the application of D–S theory in multi-sensor data fusion, consider three sensors measuring different physical quantities—strain, temperature, and vibration—to detect a clamp cracking fault. Each sensor output corresponds to a BPA representing the sensor's confidence in different hypotheses. Assume the fault hypothesis space is $\Theta = \{\theta_1, \theta_2\}$, where θ_1 indicates "clamp cracking" and θ_2 indicates "no fault." The sensor outputs are m_1, m_2, and m_3. First, m_1 and m_2 are combined using the Dempster rule. The conflict coefficient is calculated as:

$$K = m_1(\{\theta_1\}) \cdot m_2(\{\theta_2\}) + m_1(\{\theta_2\}) \cdot m_2(\{\theta_1\} \quad (11)$$

The fused BPA values are then computed:

$$m_{12}(\{\theta_1\}) = \frac{1}{1-K}(m_1(\{\theta_1\}) \cdot m_2(\{\theta_1\}) + m_1(\{\theta_2\}) \cdot m_2(\{\theta_2\}) \quad (12)$$

$$m_{12}(\{\theta_2\}) = \frac{1}{1-K}(m_1(\{\theta_1\}) \cdot m_2(\{\theta_2\}) + m_1(\{\theta_2\}) \cdot m_2(\{\theta_1\}) \quad (13)$$

m_{12} is combined with m_3 using the same procedure. The final fusion result, $m_f(\{\theta_1\})$ and $m_f(\{\theta_2\})$, represents the system's ultimate diagnostic output. The relative magnitudes of $m_f(\{\theta_1\})$ and $m_f(\{\theta_2\})$ indicate the confidence in "clamp cracking" versus "no fault." The pseudocode for the D–S theory-based multi-sensor fusion model is shown in Figure 3.

```
# Step 1: Collect data from all sensors
for each sensor in S:
    data[sensor] = collect_data(sensor)  # Collect sensor data (temperature, strain,
torque)
    BPA[sensor] = calculate_BPA(data[sensor])  # Calculate BPA for each sensor

# Step 2: Fuse the sensor data using Dempster's Rule
fused_BPA = BPA[1]  # Start with the first sensor's BPA
for i = 2 to number_of_sensors:
    conflict_coefficient = calculate_conflict(BPA[fused_BPA], BPA[i])
    fused_BPA = combine_BPA(fused_BPA, BPA[i], conflict_coefficient)

# Step 3: Compute belief and plausibility for the fused result
belief = compute_belief(fused_BPA)
plausibility = compute_plausibility(fused_BPA)

# Step 4: Make decision based on belief and plausibility
if belief['crack'] > threshold:
    decision = "Crack detected"
else if plausibility['no_fault'] > threshold:
    decision = "No fault detected"
else:
    decision = "Indeterminate"

return decision  # Final decision

# Helper functions:
function collect_data(sensor):
    return read_sensor(sensor)  # Read data from the sensor

function calculate_BPA(data):
    return apply_sensor_model(data)  # Calculate BPA for each sensor's data

function calculate_conflict(BPA1, BPA2):
    return sum(BPA1[A] * BPA2[B] for A, B in conflict_sets)  # Conflict coefficient

function combine_BPA(BPA1, BPA2, conflict_coefficient):
    return normalize(BPA1 * BPA2 / (1 - conflict_coefficient))  # Combine BPAs using
Dempster's rule

function compute_belief(BPA):
    return sum(BPA[A] for A in hypothesis_subsets)  # Compute belief

function compute_plausibility(BPA):
    return sum(BPA[A] for A in hypothesis_plausible_sets)  # Compute plausibility
```

Figure 3: Pseudocode of the D–S theory-based multi-sensor data fusion model.

The collaborative management, fault diagnosis, information judgment, reasoning, fusion, and perception learning units together constitute the core components of the multi-sensor collaborative perception system (Figure 4).

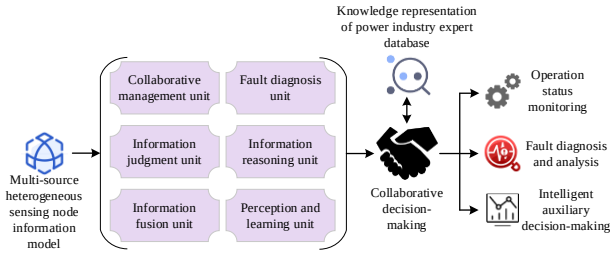


Figure 4: Main units of the multi-sensor multi-parameter collaborative perception model.

The collaborative management unit handles task requests from the perception layer and communications with other units, scheduling tasks and coordinating information flow to ensure spatiotemporal synchronization across all sources. The fault diagnosis unit processes diagnostic and feedback data from input/output units, completing a closed-loop diagnostic cycle. The information judgment and reasoning units combine historical data with a Bayesian inference model to perform probabilistic analyses for decision-making. The information fusion unit integrates multi-sensor data using Dempster–Shafer evidence theory. The perception learning unit updates prior probabilities based on historical statistics and fault records, enabling adaptive condition assessment. The overall system architecture can be represented as:

$$y_{t+1} = G(y_t, x_t, m_{fused}) \tag{14}$$

where y_t denotes the system condition at time t, x_t is the sensor feature vector, and $G(*)$ is the multi-unit collaborative fusion function.

The intelligent sensing unit is implemented on a System-on-Chip (SoC) platform, with functional modules built using Intellectual Property (IP) cores to integrate sensing, computation, communication, and power management. By combining a microprocessor with a Field-Programmable Gate Array (FPGA), the system enables real-time perception, feature extraction, and information fusion reasoning under operational conditions. Its primary function can be expressed as:

$$y_t = f_{SOC}(x_t, p_t) \tag{15}$$

where p_t is the SoC internal configuration vector, including sampling frequency, communication power, and processing strategy.

3.3 Design of ultra-low-power sensing devices

Strain clamps in transmission lines operate continuously under high voltage, strong electromagnetic interference, and harsh environmental conditions. These factors place strict demands on online monitoring devices, including miniaturization, low power consumption, high reliability, and long-term energy supply. The design strategy emphasizes developing perception units capable of stable operation under extreme conditions while meeting engineering requirements for size, power efficiency, and service life. Table 2 summarizes the key technologies and implementation methods for the ultra-low-power sensing devices.

Table 2: Key technologies and implementation methods.

Key Technology	Implementation and Description	Applicable Sensor Type
Sensor Miniaturization	Based on MEMS technology; integration at millimeter/micron scale using photolithography, etching, thin-film deposition, and ultra-precision machining; multiple physical effects sensed within a single module.	Thermistor resonator, strain gauge, torque sensor
Long-Term Reliability and Lifespan	Industrial-grade components, derating design, redundancy, heat resistance optimization; packaging provides electrical insulation, prevention of corona discharge, and EMI robustness; modular design supports online replacement and periodic calibration; software enhances configuration quality.	All sensor modules
Ultra-Low-Power Management	Work mode switching (sleep/wake), including full wake, random wake, predictive-assisted wake, or task-cycle wake modes; dynamic adjustment of data acquisition and transmission frequency balances energy consumption and monitoring precision.	Thermistor resonator, strain gauge, torque sensor
Modular Energy Supply and Wireless Recharging	Integration of micro-batteries, supercapacitors, and environmental energy harvesting; wireless power transfer via electromagnetic induction, strongly coupled resonant electromagnetic, ultrasonic, or	Micro-battery, supercapacitor, wireless energy receiver module

	microwave methods; coupling of online harvesting with fixed-terminal charging achieves continuous supply and energy scheduling.	
Environmental Adaptability	Optimized for high-voltage electromagnetic conditions, temperature fluctuations, and mechanical vibration to ensure stable data acquisition under extreme conditions; topological optimization balances transmission efficiency, safety, and spatial coupling.	All sensor modules

Sensor miniaturization leverages MEMS technology and integrates photolithography, etching, thin-film deposition, and ultra-precision machining to enable multi-physical sensing—including temperature, strain, and torque—within a single module. This approach significantly reduces device size and power consumption while enhancing environmental adaptability and manufacturing consistency. Long-term stability is ensured through the use of industrial-grade components, redundant designs, modular packaging, and online calibration, along with replaceable power modules. Ultra-low-power operation is achieved via sleep/wake mode switching and dynamic adjustment of monitoring frequency, effectively balancing energy consumption and measurement accuracy. The energy supply combines micro-batteries and supercapacitors with environmental energy harvesting and wireless replenishment. An energy scheduling algorithm manages charge and discharge across storage elements, ensuring continuous operation even under extreme conditions. The specifications for each sensor are listed in Table 3.

Table 3: Sensor specifications and calibration details.

Sensor Type	Model	Sampling Rate	Measurement Range	Sensitivity	Accuracy	Calibration Method
Strain Gauge	BHB120-3CA250	1 Hz – 1 kHz	3-axis strain, ±2500 με	Sensitivity coefficient: 1.79	±0.5 % FS	Multi-point calibration using a calibration rig
Torque Sensor	Kistler 4527C	1 Hz – 10 kHz	±500 Nm	2.0 mV/V/Nm	±0.3 % FS	Calibrated with standard loads
Thermistor Resonator	Microchip TC1047	0.1 Hz – 100 Hz	–40 °C to 125 °C	20 mV/°C	±0.3 °C	Calibrated in controlled environmental temperature
Deformation Sensor	FOS-11	1 Hz – 1 kHz	±10 mm	0.2 mV/mm	±0.2 % FS	Calibrated using standard deformation values

The wireless power transfer system is shown in Figure 5. Depending on the application scenario, electromagnetic induction, strongly coupled resonance, ultrasonic, or microwave techniques may be employed, each providing specific advantages in efficiency, coupling, and safety. Topological optimization is used to achieve an efficient spatial layout and energy transmission, supporting the long-term online monitoring of strain clamps.

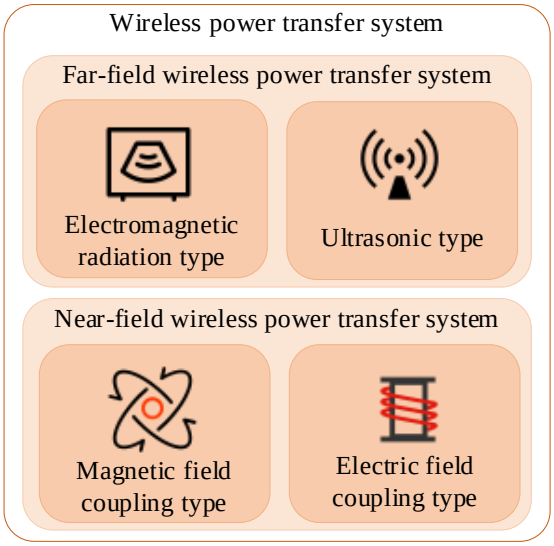


Figure 5: Wireless power transfer system.

3.4 Online monitoring software system and diagnostic platform implementation

The operating conditions of transmission line strain clamps are complex and variable, and hardware sensors alone cannot provide full lifecycle health management. Therefore, software and the diagnostic platform form the other two pillars of the system architecture. The design strategy integrates perception with intelligent diagnostics, enabling high-volume data reception, situational awareness, and decision support in the cloud. This approach establishes a platform comprising monitoring, analysis, evaluation, and early-warning modules.

Following development, the devices were installed and commissioned on a 66 kV transmission line. Sensor data were transmitted through gateways to the cloud, providing real-time measurements of temperature, strain, deformation, and torque under varying operating conditions. The first module, the Situational Awareness Module, visually displays key operational parameters and issues warnings or alerts when thresholds are exceeded. This enables maintenance personnel to quickly identify anomalies and carry out condition-based or preventive maintenance. Beyond visualization, the module also supports on-site verification, ensuring that monitoring results align with manual inspections and flaw-detection instruments.

The second key module is the multi-sensor, multi-parameter diagnostic module. This module stores data collected from various sensors in a database and performs real-time condition diagnosis of the strain clamps using a collaborative perception model. It employs an event-driven fault diagnosis approach, where the diagnostic process is triggered only when sensor measurements exceed predefined thresholds. The specific thresholds are as follows: a temperature threshold, where a warning is issued if the clamp temperature exceeds 85 °C and an alarm is triggered if it exceeds 100 °C; a strain threshold, with a warning at 1.2 times the normal value and an alarm at 1.5 times; a deformation threshold, with a warning at 2% above the standard value and an alarm at 3%; and a torque threshold, with a warning at 10% above the design load and an alarm at 15%. The system also calculates probability scores for fault occurrence using multi-sensor data fusion. Based on the fused sensor data, the probability of each fault hypothesis is computed. If a fault probability exceeds 70%, a warning is issued; if it exceeds 90%, an alarm is triggered. These confidence thresholds are determined from historical data and expert knowledge and can be dynamically adjusted according to the operational context. By integrating information from multiple sensors, the system does not rely solely on single-parameter thresholds but evaluates the overall condition, significantly improving diagnostic accuracy and reliability. The module is capable of outputting the operational status levels of strain clamps and generating dynamic diagnostic reports. On-site diagnostic results are cross-validated with sensor measurements, forming a positive feedback loop that continuously refines and optimizes the model.

The third module, the Query and Statistics Module, enables archival, retrieval, and multi-dimensional analysis of historical data. Operators can monitor the condition trajectories of strain clamps over time, detect long-term trends, and identify potential risks. By integrating sensing and computation, the software system and diagnostic platform overcome the limitations of traditional isolated monitoring systems, creating an ecosystem that combines data acquisition, real-time analysis, intelligent diagnostics, and traceable management.

4 Experiments and result analysis

4.1 Laboratory performance testing and validation

The diagnostic performance of the ultra-low-power sensing devices and the multi-sensor information fusion model was first validated offline under controlled laboratory conditions. Calibration and performance tests were systematically conducted for thermistor resonators, torque sensors, and strain gauges, allowing precise measurement of temperature, resistance, torque, and strain signals. Sensor data were collected under various simulated operating conditions, including temperature fluctuations, mechanical stress, and bolt loosening scenarios. These data were used to establish single-sensor diagnostic models as well as a multi-sensor information fusion model, with the goal of evaluating the models' ability to accurately detect typical strain clamp faults.

Data were collected from seven test points. At each point, every sensor continuously recorded measurement for one hour, with a data point logged every second. Consequently, one day of sensor data consisted of second-level measurements, and over a total of seven days, 25,200 test cases were obtained. Single-sensor models were developed independently for each sensor type using a Support Vector Machine (SVM) classifier. Performance metrics, including accuracy, precision, and F1-score, were rigorously evaluated using 5-fold cross-validation. In contrast, the multi-sensor information fusion model combined signals from all sensors within an ensemble classification framework. Fault diagnosis employed a weighted voting strategy, while a decision tree algorithm performed classification, enabling the model to exploit complementary information from multiple sensors and improve reliability. The diagnostic results, comparing single-sensor models with the multi-sensor fusion model, are presented in Figure 6, highlighting the performance gains achieved through multi-source information integration.

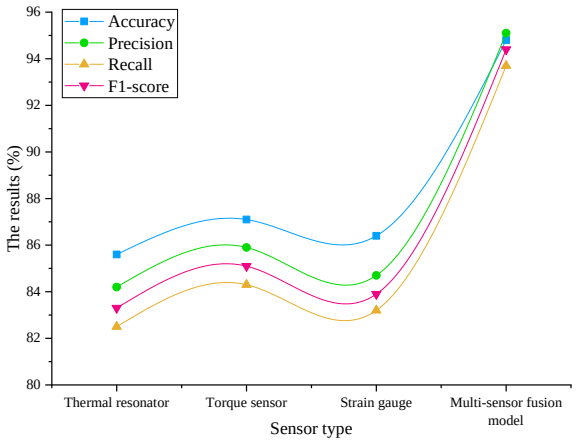


Figure 6: Comparison of diagnostic performance between single-sensor and fusion models.

The results in Figure 6 demonstrate that the multi-sensor fusion model significantly outperforms the single-sensor models. The fusion model achieved an accuracy of 94.8% and an F1-score of 94.4%, both exceeding the performance of any individual sensor. Compared to the best-performing single sensor, the torque sensor, the fusion model improved accuracy and F1-score by 8.84% and 10.9%, respectively. These findings indicate that multi-source information fusion substantially enhances the identification of strain clamp faults, including cracking, deformation, and bolt loosening, providing a reliable basis for field condition monitoring. To assess energy efficiency, the power consumption of each sensor and the fusion model was measured, as shown in Figure 7. All energy consumption values represent the average of five experimental runs, with standard deviations calculated to reflect data variability.

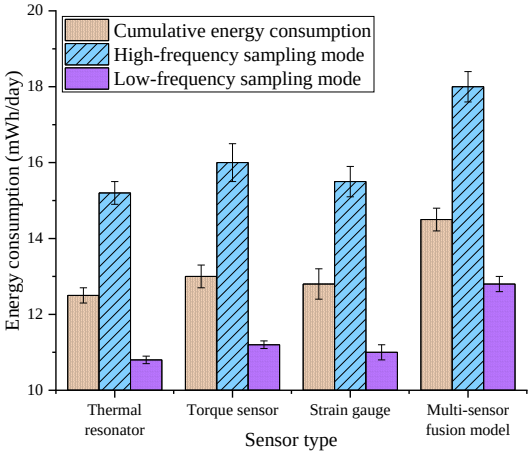


Figure 7: Comparison of cumulative energy consumption.

As shown in Figure 7, under standard operating conditions, the fusion model consumes 14.5 mWh per day, slightly higher than the 12.5–13.0 mWh recorded for individual sensors. In high-frequency sampling mode, energy consumption of the multi-sensor fusion model increases to 18.0 mWh/day, approximately 3–5 mWh higher than under standard operation. Conversely, in low-frequency sampling mode, consumption decreases to 12.8 mWh/day, approaching the levels observed for single-sensor operation. These results indicate that energy consumption is strongly influenced by both sampling frequency and operating mode. By selecting an appropriate sampling strategy, an optimal balance can be achieved between energy efficiency and diagnostic performance. This approach ensures long-term sustainability of the monitoring system and extends the operational lifetime of the sensors, providing a practical and effective solution for continuous online monitoring of strain clamps over extended periods.

To comprehensively evaluate system performance, the proposed multi-sensor fusion model was compared with classical adaptive control methods, including adaptive backstepping control, neural adaptive control, observer-based control, as well as other mainstream approaches such as Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM). The results are summarized in Table 4.

Table 4: Comparison with other methods.

Method	Accuracy (%)	F1-score (%)	Detection Time (s)	Energy Consumption (mWh/day, mean ± SD)	CPU Usage (%)	Memory (MB)
Multi-sensor fusion model	94.8	94.4	0.35	14.5 ± 0.3	45.2	147
Adaptive backstepping control	91.5	92.0	0.58	18.2 ± 0.4	52.1	158
Neural adaptive control	92.5	93.0	0.65	18.5 ± 0.3	54.7	163
Observer-based control	90.8	91.5	0.55	17.8 ± 0.3	48.6	151
Single-sensor SVM model	86.5	85.7	0.42	12.5 ± 0.2	38.4	126
Random Forest fusion model	91.2	90.5	0.48	17.6 ± 0.3	50.3	139
CNN-LSTM hybrid model	92.8	92.1	0.62	19.4 ± 0.5	62.5	182

Autoencoder fault detection model	90.7	89.8	0.57	16.9 ± 0.4	55.0	171

Table 4 shows that the multi-sensor fusion model outperforms other control and deep learning methods in terms of accuracy and F1-score, while achieving faster response times and lower energy consumption. Compared with the CNN-LSTM model, the fusion model reduces energy consumption by 25.3% and shortens response time by 43.5%, with a CPU usage of 45.2% and memory footprint of 147 MB. These advantages stem from several factors: the model fuses thermal, torque, and strain signals through a weighted decision framework, enhancing feature complementarity and robustness; a predictive-assisted wake-up mechanism minimizes energy waste from high-frequency sampling by enabling active monitoring during critical periods and low-power standby during stable operation; and dynamic feature weighting adjusts sensor contributions according to operating conditions, improving detection accuracy. The system's low energy consumption further supports long-term monitoring and efficient resource management, making it suitable for practical online applications. However, under extreme conditions such as low temperatures, high humidity, or strong electromagnetic interference, sensor drift or feature distortion may occur, potentially blurring classification boundaries. Future work may include temperature–humidity compensation algorithms or fuzzy adaptive control strategies for online model adjustment. Lightweight deep fusion architectures, such as Tiny-CNN or edge-based autoencoder fusion, could also be explored to further reduce energy consumption and enhance generalization.

The diagnostic performance of the multi-sensor fusion model for typical faults—including bolt loosening, clamp cracking, fitting deformation, and local overheating—was further analyzed, as shown in Table 5.

Table 5: Diagnostic performance by fault type.

Fault Type	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Bolt loosening	95.3	96.7	95.0	95.8
Clamp cracking	94.6	95.2	93.8	94.5
Fitting deformation	94.4	94.0	92.0	93.0
Local overheating	94.7	94.4	94.1	94.2
Average	94.8	95.1	93.7	94.4

Table 5 indicates that the multi-sensor fusion model maintains high diagnostic performance across different fault types, with overall accuracy exceeding 94%. Bolt loosening and clamp cracking show the highest F1-scores (95.8% and 94.5%, respectively), demonstrating that the fused multi-parameter features effectively capture mechanical and thermal variations. Detection performance for fitting deformation and local overheating is slightly lower due to partial signal overlap in early stages of these faults. Overall, the multi-parameter fusion strategy enhances robustness and diagnostic accuracy for multiple fault types, providing reliable support for on-site status assessment and early warning.

To verify the advantages of Dempster–Shafer (D-S) theory, benchmark tests were conducted comparing Bayesian inference, weighted averaging fusion, and Extreme Gradient Boosting (XGBoost), with results summarized in Table 6.

Table 6: Comparison of different fusion methods.

Method	Accuracy (%)	F1-score (%)	Detection Time (s)	Energy Consumption (mWh/day, mean ± SD)
D-S Theory	94.8	94.4	0.35	14.5 ± 0.3
Bayesian Inference	92.3	90.6	0.23	15.2 ± 0.3
Weighted Average Fusion	92.0	90.2	0.20	15.0 ± 0.2
XGBoost	93.0	91.7	0.38	18.1 ± 0.5

As shown in Table 6, the multi-sensor fusion model based on D-S theory outperforms the other fusion methods. It achieves the highest accuracy and F1-score, clearly exceeding the performance of Bayesian inference, weighted averaging, and XGBoost. Although Bayesian inference and weighted averaging offer slightly faster detection times, their diagnostic accuracy is lower. Compared with XGBoost, the D-S model achieves slightly shorter detection times (0.35 s vs. 0.38 s) while consuming less energy, highlighting its suitability for low-power platforms. Overall, the D-S-based multi-sensor fusion model provides higher fault detection precision with lower energy consumption, making it an effective solution for long-term online monitoring in energy-constrained environments.

4.2 Field pilot application and performance evaluation

A pilot monitoring study was conducted on a 66 kV transmission line operated by the State Grid Siping Power Supply Company to evaluate the multi-sensor fusion model in detecting strain clamp faults, including cracking, deformation, and bolt loosening. To ensure safe operation under high-voltage conditions, all sensors and acquisition devices were designed with high-voltage isolation. Electrical isolation was further enhanced using insulating materials and shielding, while electromagnetic interference filters were installed at critical line interfaces

to minimize signal disturbance. The hardware design complied with relevant State Grid and International Electrotechnical Commission (IEC) high-voltage safety standards, and internal testing confirmed insulation, voltage withstand, and anti-interference performance, ensuring personnel and equipment safety under 66 kV operating conditions. Under these safety measures, the system's diagnostic accuracy, micro-power management strategy, and energy scheduling scheme were evaluated to verify the extension of system operational life. During field deployment, 30 strain clamps were equipped with thermistor resonators, torque sensors, and strain gauges to continuously collect real-time data under varying environmental and mechanical conditions. The multi-sensor fusion model processed these multi-source signals to identify faults and generate diagnostic reports, which were then validated against manual inspection results. In addition to fault detection, system stability, response performance, and energy consumption were monitored to comprehensively assess the monitoring solution. Figure 7 presents the confusion matrix for strain clamp fault diagnosis, illustrating the model's performance across different fault types and operating conditions.

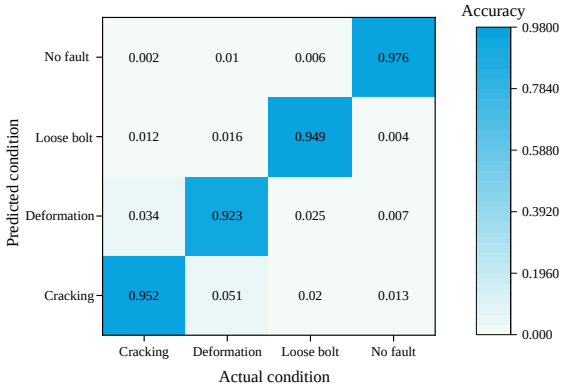


Figure 8: Confusion matrix of strain clamp fault diagnosis.

As shown in Figure 8, the model maintains consistently high diagnostic accuracy across different fault types. Cracks and fault-free conditions were detected with over 95% accuracy, demonstrating the system's ability to reliably distinguish between healthy and compromised clamps. For deformation and bolt loosening, accuracy remained high at 92.3% and 94.9%, respectively, indicating that the multi-sensor fusion approach effectively captures subtle mechanical and structural changes. Overall, the system demonstrates high fault-detection accuracy. Although occasional misclassifications occur, the errors are minor, indicating that the system can effectively distinguish between different fault types in practical applications. Moreover, the confusion matrix reflects classification performance and directly informs the system's maintenance decision logic. The system implements a three-tiered response mechanism based on diagnostic results: 1. Level 1 Warning (Minor Anomaly): If the model's confidence is low or adjacent classes are detected (e.g., "bolt loosening" misclassified as "local overheating"), the system flags a potential risk without triggering immediate maintenance. The monitoring platform increases observation frequency to track the situation. 2. Level 2 Warning (Moderate Anomaly): When the same fault type is detected repeatedly, or the confusion matrix indicates a high probability of adjacent fault transitions (e.g., between "deformation" and "loosening"), the system triggers a manual inspection task. 3. Level 3 Alarm (Severe Anomaly): If the identification is stable with high confidence (≥95%) and corresponds to critical faults such as "clamp cracking" or "severe overheating," the system automatically reports to the grid operations center and recommends power shutdown for maintenance. This tiered decision-making approach, grounded in the probabilistic distribution of the confusion matrix, enables intelligent responses according to fault severity. It prevents unnecessary maintenance while ensuring timely intervention for potential hazards, thereby improving safety and operational efficiency.

To enhance energy efficiency in the multi-sensor fusion model, four wake-up strategies are employed: full wake-up, random wake-up, predictive-assisted wake-up, and task-cycle wake-up. The predictive-assisted wake-up strategy relies on historical trend prediction to determine the optimal next wake-up time, avoiding unnecessary frequent activations. Specifically, the mechanism uses historical sensor data and machine learning–based time series models, such as LSTM, to forecast potential events or state changes in the upcoming period, optimizing wake-up timing. In this approach, each sensor node independently predicts and decides when to wake, based on its local historical data, rather than relying on a centralized controller. Each node determines whether to enter low-power mode or wake based on its own data and predictions. This decentralized strategy reduces communication overhead and further optimizes energy consumption. Five experiments were conducted, and the average system endurance, response time, and energy consumption under the four wake-up strategies are presented in Figure 9.

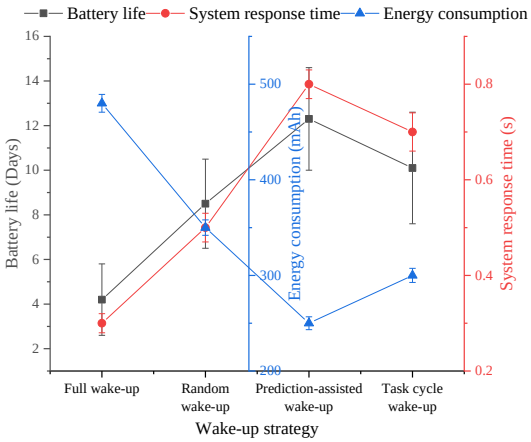


Figure 9: Comparison of endurance under different wake-up strategies.

Figure 9 shows that the predictive-assisted wake strategy achieves the longest system endurance of 12.3 days with relatively low energy consumption of 250 mAh,

though the response time is slightly longer at 0.8 s. The task-cycle wake strategy consumes slightly more energy than predictive-assisted wake while still maintaining relatively long endurance. In contrast, the full wake strategy exhibits the shortest endurance at 4.2 days and the highest energy consumption of 480 mAh, but offers the fastest response time of 0.3 s. Overall, the predictive-assisted wake strategy provides an effective balance between endurance and energy efficiency, making it well suited for long-term online monitoring applications.

4.3 Discussion

In this study, the proposed multi-sensor fusion-based online monitoring system, combined with a predictive-assisted wake strategy, demonstrated high diagnostic accuracy and energy efficiency. Compared with existing approaches, the proposed method shows advantages in detection precision, energy consumption, and response speed. Boulkroune et al. proposed an output-feedback projection lag synchronization scheme for chaotic drive–response systems with input dead zones and sector nonlinearities. The method employs a variable-structure controller and an adaptive fuzzy system to handle uncertain dynamics [24]. This method primarily addresses challenges in unmeasurable system states and dynamic uncertainties, focusing on system stability and synchronization, but it does not specifically target fault diagnosis in physical devices such as strain clamps. In contrast, the monitoring system presented here achieves high fault detection accuracy, with an overall accuracy of 94.8% and an F1-score of 94.4%. By integrating data from multiple sensors, it effectively enhances diagnostic performance in complex environments through the synergistic use of temperature, strain, and torque information. Boulkroune et al. also introduced a nonsingular fixed-time sliding mode-based adaptive fuzzy controller for fixed-time synchronization of heterogeneous fractional-order chaotic systems [25]. While this approach demonstrates good convergence in nonlinear systems, it lacks focus on energy efficiency and does not directly address device-level fault diagnosis. By comparison, the method ensures high diagnostic accuracy and emphasizes energy efficiency, using optimized sampling strategies such as predictive-assisted wake to reduce power consumption without compromising diagnostic performance. This makes it well suited for long-term, energy-constrained online monitoring applications. Zouari et al. proposed a Lyapunov-based robust adaptive backstepping control method for uncertain single-input single-output nonlinear systems, emphasizing stability and convergence of tracking errors [26]. Although theoretically capable of ensuring error convergence, such methods may encounter practical limitations in complex electrical system fault diagnosis due to high model complexity and energy demands. In contrast, the proposed system leverages multi-sensor fusion and simplified classification models to achieve high diagnostic accuracy while maintaining low energy consumption, meeting the requirements of extended online monitoring. Zouari et al. further developed robust neural adaptive and indirect neural adaptive control strategies for multivariable uncertain nonlinear systems [27]. While robust and effective at disturbance rejection, these approaches incur significant computational overhead, resulting in lower energy efficiency. The multi-sensor fusion system presented here employs low-power sensors combined with an effective energy management strategy, enabling high-accuracy diagnosis while minimizing energy consumption, particularly during low-power operation. Rigatos et al. introduced a nonlinear optimal control method for gas centrifugal compressors, using linearized models and feedback controllers to handle uncertainties and external disturbances [28]. Although computationally efficient, this approach is complex and not optimized for the energy requirements of long-term online monitoring. By contrast, the multi-sensor fusion model developed in this study maintains high accuracy while using adaptive sampling and predictive-assisted wake strategies to significantly reduce long-term energy consumption, making it suitable for real-time monitoring and fault warning in smart grids. Merazka et al. proposed a fuzzy adaptive output-feedback control approach, employing high-gain observers to estimate unmeasurable states and fuzzy systems to model uncertainties for closed-loop stability [29]. While robust and stable, this method has relatively low energy efficiency and focuses on state estimation in complex systems rather than direct fault diagnosis of physical devices. The online monitoring system proposed here integrates multiple sensors and fusion algorithms to directly address the specific fault types of strain clamps, achieving more efficient and accurate online fault detection while maintaining low power consumption. Overall, compared with the aforementioned studies, the multi-sensor fusion-based online monitoring technology presented in this work offers clear advantages in accuracy, response speed, and energy efficiency. By integrating diverse sensor data and advanced energy management strategies, the system ensures high-performance diagnosis while effectively controlling energy consumption, making it highly suitable for real-time monitoring and fault warning in complex smart grid environments.

Although the proposed online monitoring system demonstrates high accuracy and low energy consumption under normal conditions, its performance may be affected under extreme environmental conditions. For example, in low-temperature, high-humidity, or high-wind environments, sensor accuracy may be compromised. Thermistor resonators and strain gauges are particularly sensitive to temperature and humidity variations. Under extreme temperatures, thermal elements may drift, leading to inaccurate measurements, while high humidity can affect the resistance of strain gauges, reducing the reliability of fault diagnosis. In such harsh conditions, the system may require additional sensor redundancy or more frequent wake cycles to maintain stability, which inevitably increases energy consumption. For instance, under low-temperature conditions, ensuring continuous monitoring may necessitate higher-frequency wake-ups to detect sudden faults, consuming extra power. Therefore, improving the system's robustness against environmental

influences and minimizing their impact on performance is a key direction for future research. This could be achieved through temperature compensation, humidity correction algorithms, and dynamic adjustment of fault-diagnosis models.

Different transmission line types—high-, medium-, and low-voltage—as well as geographic variations, may also affect system performance. High-voltage and low-voltage lines differ significantly in load fluctuations, fault modes, and environmental conditions, requiring the system to dynamically adapt to specific line characteristics. In mountainous areas, regions with high electricity demand, or coastal environments, the system must accommodate complex conditions, such as frequent weather changes and larger load variations. In these scenarios, adaptive operation can be achieved through flexible sensor fusion and predictive wake strategies. Specifically, an environmental-awareness module can automatically adjust monitoring frequency and wake strategies based on real-time data, ensuring efficient operation and accurate fault diagnosis under diverse conditions. To further enhance long-term stability and operational efficiency, future developments could integrate adaptive control methods, such as fuzzy control or nonlinear optimal control, enabling real-time tuning of system parameters to optimize monitoring and energy management, thereby improving both energy efficiency and diagnostic accuracy.

Regarding system scalability, when monitoring hundreds or even thousands of strain clamps, bandwidth requirements and server loads become critical. As the number of sensors increases, the system must handle large volumes of real-time data, potentially creating transmission bottlenecks and limiting server processing capacity. For large-scale deployment, data acquisition, transmission, and processing workflows need to be optimized. Edge computing can be employed to offload data processing tasks to local nodes, reducing the burden on central servers. Data compression and distributed storage can further enhance scalability. To maintain diagnostic accuracy at scale, the system should support efficient parallel computation and robust fault-tolerant mechanisms, ensuring reliable results even when multiple nodes process data simultaneously.

The proposed online monitoring system improves fault diagnosis accuracy for transmission lines but also has broad practical applications, particularly in predictive maintenance for smart grids. By continuously monitoring the health of strain clamps, the system can identify potential faults in advance and estimate maintenance intervals, enabling necessary interventions without interrupting grid operation. This approach reduces equipment failure rates, lowers maintenance costs, and enhances grid reliability and stability. In smart grid applications, the system can integrate with existing dispatch and monitoring platforms, sharing data and leveraging fault predictions to optimize grid operation, minimize downtime, and enhance overall grid intelligence.

5 Conclusion

This study proposed an ultra-low-power online monitoring approach for transmission line strain clamps, employing multi-sensor collaborative perception and a fusion model to detect clamp faults. Validation results showed that the fusion model significantly outperformed single-sensor methods, demonstrating that integrating multi-source information effectively enhances fault diagnosis capability. These findings provide reliable technical support for field monitoring and contribute to improving the operation and maintenance of transmission lines. Furthermore, by comparing energy consumption and operational endurance under different wake-up strategies, the predictive-assisted wake strategy was shown to extend system endurance while reducing energy use, offering practical guidance for selecting wake-up strategies in energy-constrained applications.

Although the proposed method can detect major fault types under different wake-up strategies, it has certain limitations in monitoring high-frequency or transient faults. For example, some faults, such as arc flashovers or instantaneous short circuits, may occur and conclude during wake-up delays, making timely detection impossible. With a system response time of 0.8 s, these rapid faults could go undetected, potentially leading to misdiagnosis or missed events. This limitation may affect diagnostic accuracy and, consequently, the reliability of maintenance decisions during sudden, fast-changing fault conditions. Future research could focus on improving the system's wake-up mechanism and fault detection algorithms, particularly for applications requiring higher response speed. By optimizing system architecture and exploring more efficient wake-up strategies and real-time detection algorithms, the ability to capture high-frequency faults could be significantly enhanced. In parallel, more precise energy management strategies could help extend operational endurance without compromising responsiveness, meeting the monitoring needs of transmission lines under complex conditions. Moreover, the proposed approach is not limited to transmission line monitoring; it could be extended to other power facilities, transportation systems, and similar applications, providing technical support for the construction and maintenance of intelligent infrastructure.

References

[1] Chen J, Xu H, Liu L, et al. Defect Detection Technology of Tension Clamps Based on UAV. GBP Proceedings Series, 2025, 7, pp. 46-50.

[2] Shao H, Lv J, Wang J, et al. Automatic crimping state classification in X-ray images of crimped tension clamp. Nondestructive Testing and Evaluation, 2025, 40(2), pp. 610-626.

[3] Cloostermans B, Pronk D, Bruckenburg B, et al. Bolted Flange Connection Leakage: a Systematic Review of Monitoring Challenges and Technologies. IEEE Sensors Reviews, 2025, 2, pp. 1-16.

[4] Thango B A. Winding Fault Detection in Power Transformers Based on Support Vector Machine and Discrete Wavelet Transform Approach. Technologies, 2025, 13(5), pp. 200.

[5] Prejbeanu R G. A Sensor-based system for fault detection and prediction for EV multi-level converters. Sensors, 2023, 23(9), pp. 4205.

[6] Ruzimov S, Zhang J, Huang X, et al. Detection of Inter-Turn Short-Circuit Faults for Inverter-Fed Induction Motors Based on Negative-Sequence Current Analysis. Sensors, 2025, 25(15), pp. 4844.

[7] Shen H, Tang X, Luo Y, et al. Online open-circuit fault diagnosis for neutral point clamped inverter based on an improved convolutional neural network and sample amplification method under varying operating conditions. IEEE Transactions on Instrumentation and Measurement, 2024, 73, pp. 1-12.

[8] Yang X, Tu Y, Yuan Z, et al. Intelligent overheating fault diagnosis for overhead transmission line using semantic segmentation. High Voltage, 2024, 9(2), pp. 309-318.

[9] Laaziz M, Nicolau F, Ghanes M, et al. Fault Detection, Isolation, and Estimation for a Three-Phase Grid Connected DC-AC Inverter With LCL Filters. IEEE Control Systems Letters, 2024, 8, pp. 1769-1774.

[10] Qiu Z, Li J, Zhu X, et al. Detection of Crimping Defects in X-DR Images of Strain Clamps Based on Multi Attention Mechanism Fusion Network. IEEE Transactions on Power Delivery, 2024, 39(4), pp. 2125-2137.

[11] Shakiba F M, Azizi S M, Zhou M, et al. Application of machine learning methods in fault detection and classification of power transmission lines: a survey. Artificial Intelligence Review, 2023, 56(7), pp. 5799-5836.

[12] Guo B, Li Q, Jiang W, et al. A deep sensing system for monitoring the health status of transmission line equipment based on self-powered sensors. IEEE Sensors Journal, 2023, 23(18), pp. 20834-20844.

[13] Luo P, Wang B, Wang H, et al. An ultrasmall bolt defect detection method for transmission line inspection. IEEE Transactions on Instrumentation and Measurement, 2023, 72, pp. 1-12.

[14] Dadashizadeh Samakosh J, Enayati F. Operation recommendations for tension joints and clamps on a 63 kV overhead transmission line conductor based on experimental tests. Electric Power Components and Systems, 2023, 51(7), pp. 639-655.

[15] Chen B, Zhang W, Wang F, et al. Development of automatic arc sag measurement equipment for tensioned line tightening construction on overhead transmission lines. Scientific Reports, 2024, 14(1), pp. 19253.

[16] Luo Y, Gao C, Wang D, et al. Predictive model for sag and load on overhead transmission lines based on local deformation of transmission lines. Electric Power Systems Research, 2023, 214, pp. 108811.

[17] Li P, Fan Y, Liu Z, et al. Application status and development trend of intelligent sensor technology in the electric power industry. IET Science, Measurement & Technology, 2024, 18(4), pp. 145-162.

[18] Mai Y, Xing Y, Ye Z, et al. Current reconstruction and sag/wind deflection monitoring of transmission lines by magnetic sensing. IEEE Transactions on instrumentation and measurement, 2024, 73, pp. 1-13.

[19] Lin Y, Zhou T, Wang Z. A High-Reliable Wireless Sensor Network Coverage Scheme in Substations for the Power Internet of Things. Symmetry, 2023, 15(5), pp. 1020.

[20] Pelini J, Russo S D, Garcia I L, et al. New silicon-based micro-electro-mechanical systems for photo-acoustic trace-gas detection. Photoacoustics, 2024, 38, pp. 100619.

[21] Gazzola C, Corigliano A, Zega V. Total harmonic distortion estimation in piezoelectric micro-electro-mechanical-system loudspeakers via a FEM-assisted reduced-order-model. Mechanical Systems and Signal Processing, 2025, 222, pp. 111762.

[22] Rzayeva S V, Guseynova I A. Application of automatic monitoring and control systems for reliability of power transmission lines. Reliability: Theory & Applications, 2024, 19(2 (78)), pp. 64-69.

[23] Pang G, Zhang Z, Hu J, et al. Analysis of Failures and Protective Measures for Core Rods in Composite Long-Rod Insulators of Transmission Lines. Energies, 2025, 18(12), pp. 3138.

[24] Boulkroune A, Hamel S, Zouari F, et al. Output-Feedback Controller Based Projective Lag-Synchronization of Uncertain Chaotic Systems in the Presence of Input Nonlinearities. Mathematical Problems in Engineering, 2017, 2017(1), pp. 8045803.

[25] Boulkroune A, Zouari F, Boubellouta A. Adaptive fuzzy control for practical fixed-time synchronization of fractional-order chaotic systems. Journal of Vibration and Control, 2025, pp. 10775463251320258.

[26] Zouari F, Saad K B, Benrejeb M. Adaptive backstepping control for a class of uncertain single input single output nonlinear systems. 10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13). IEEE, 2013, pp. 1-6.

[27] Zouari F, Saad K B, Benrejeb M. Robust neural adaptive control for a class of uncertain nonlinear complex dynamical multivariable systems. International Review on Modelling and Simulations, 2012, 5(5), pp. 2075-2103.

[28] Rigatos G, Abbaszadeh M, Sari B, et al. Nonlinear optimal control for a gas compressor driven by an induction motor. Results in Control and Optimization, 2023, 11, pp. 100226.

[29] Merazka L, Zouari F, Boulkroune A. High-gain observer-based adaptive fuzzy control for a class of multivariable nonlinear systems. 2017 6th International Conference on Systems and Control (ICSC). IEEE, 2017, pp. 96-102.

Anomaly Detection in Building Equipment Energy Consumption Using Bi-LSTM and Convolutional Block Attention Mechanism

Xin Wan[1*], Xiaoling Cai[1], Lele Dai[2]
[1]School of Architecture and Environmental Engineering, Wuxi City College of Vocational Technology
Wuxi 214000, China
[2]School of Energy and Power Engineering, Nanjing University of Science and Technology
Nanjing 210000, China
E-mail: wanxin840207@163.com, caixl123456@126.com, dyldagao@njust.edu.cn
*Corresponding author

Keyword: Bi-LSTM, CBAM, construction equipment, abnormal energy consumption

Received: August 12, 2025

With the growing complexity of building energy systems, accurate anomaly detection in equipment energy consumption has become crucial for improving operational efficiency. This study proposes an energy anomaly detection model that integrates Bidirectional Long Short-Term Memory (Bi-LSTM) networks and a Convolutional Block Attention Module (CBAM). The input features include low and high energy consumption time ratios and dynamic time warping distance, constructed into a 32-dimensional feature vector using a sliding window of 24 hours. The Bi-LSTM layer with 128 forward and 128 backward hidden units captures bidirectional temporal dependencies. CBAM refines critical feature dimensions and time steps through channel and spatial attention mechanisms. The model was trained using 70% of labeled data from the LEAD1.0 public smart meter dataset and tested on the remaining 30%. Experimental results showed that on Dataset A, the model achieved an accuracy of 0.98, a recall of 0.97, and an RMSE of 0.02. On Dataset B, it achieved 0.97, 0.98, and 0.05, respectively. Comparative analysis against baseline models, including Bi-LSTM, LSTM, and GRU, demonstrated significant improvements in both accuracy and error metrics. An ablation study confirmed the contribution of each module to model performance. Statistical validation across multiple runs showed that the improvements were consistent and robust. These findings suggest that combining Bi-LSTM with dual-attention mechanisms provides an effective solution for detecting both transient and persistent energy anomalies in dynamic building environments.

Povzetek: Študija predstavlja model Bi-LSTM s CBAM mehanizmom pozornosti za zaznavanje energetskih anomalij v stavbah, ki dosega zelo visoko natančnost in zanesljivo presega primerjalne modele.

1 Introduction

With the continuous growth of global energy requirement, energy management issues in the construction industry are receiving increasing attention. According to statistics, building Energy Consumption (EC) accounts for over 40% of the world's total EC, with Abnormal Energy Consumption (AEC) of building equipment being one of the main causes of energy waste. Therefore, how to efficiently and accurately detect the AEC of building equipment to enhance energy efficiency and lower CO2 emissions has become an important research direction for intelligent building management and energy-saving optimization. Traditional energy management mainly relies on fixed threshold monitoring and statistical analysis methods. However, due to the complex operating environment and large load fluctuations of building equipment, fixed rules are difficult to adapt to dynamically changing EC patterns, which can easily lead to misjudgments and omissions [1]. In addition, EC data of building equipment usually exhibits nonlinear and non-stationary characteristics in time series. Traditional methods have limited capabilities in feature extraction and pattern recognition, and cannot effectively address complex anomaly detection requirements [2]. Traditional threshold detection methods rely on expert experience to set fixed boundaries, which are difficult to adapt to dynamically changing operating status and are prone to false alarms due to seasonal fluctuations or equipment aging. Although statistical learning models can identify some abnormal patterns, they lack sufficient modeling of temporal dependencies and cannot capture the continuous evolution of device operating states. Currently, Long Short-Term Memory (LSTM) networks have been extensively utilized in EC prediction due to their gating mechanism's advantage in processing time-series data. However, the unidirectional network structure can only capture historical information and is difficult to synchronously analyze future trend features. Therefore, this study constructs an AEC detection model that combines Bidirectional LSTM (Bi-LSTM) and Convolutional Block Attention Mechanisms (CBAM). This method utilizes a bidirectional structure to obtain information about the time steps before and after,

enhancing the ability to model time series. Meanwhile, CBAM is introduced to optimize the feature extraction schedule and improve the model's attention to key EC patterns. The primary objective of this research is to design and validate a building equipment AEC detection model that can reliably capture both low and high EC anomalies across different operational scenarios, improving detection accuracy and reducing false alarms. The study focuses on integrating bidirectional temporal modeling with feature attention to enhance anomaly recognition performance in practical smart building environments. The innovation lies in the use of Bi-LSTM to improve time series modeling capabilities, while introducing CBAM to enhance feature selection capabilities. Different from existing anomaly detection models that predominantly use unidirectional LSTM networks or basic attention mechanisms, this study uniquely integrates the bidirectional temporal feature extraction advantage of Bi-LSTM with the channel-spatial dual attention mechanism of CBAM. This integration effectively resolves the limitations of traditional methods in capturing complex temporal dependencies and accurately extracting critical EC features simultaneously. Moreover, the method proposed herein significantly enhances generalization capability and reduces false alarms, addressing practical issues commonly encountered in diverse and dynamic building operational environments.

2 Related works

Building energy management is crucial for energy conservation, consumption reduction, and sustainable development. Accurate detection of AEC can effectively avoid energy waste and optimize equipment operation. However, EC data of building equipment have temporal and fluctuating characteristics. Traditional methods, such as fixed-threshold monitoring, statistical modeling approaches (e.g., moving average and ARIMA), and optimization-based techniques like Particle Swarm Optimization (PSO) or Genetic Algorithms (GA), are hard to precisely identify abnormal patterns in complex and dynamic building environments. In phase change heat storage system optimization, Xiao et al. designed a multi-channel phase change heat storage device with porous metal foam structure, and verified its heat transfer performance through 3D numerical simulation. By further combining the LSTM and backpropagation neural network hybrid prediction model, experiments have shown that increasing the flow rate or inlet temperature of the heat transfer medium could accelerate the melting of phase change materials. This intelligent algorithm provided a new approach for dynamic control of thermal storage systems [3]. Zheng et al. developed a dual-channel image analysis model that combines convolutional neural networks and LSTM to address the problem of partial discharge pattern recognition in power equipment. Compared with traditional machine learning and single deep learning models, its accuracy in classifying 4 kinds of discharge patterns has significantly improved. The effectiveness of the model architecture and hyperparameter design has been verified through ablation experiments [4]. In terms of predicting the lifespan of energy devices, Zhang et al. constructed a residual lifespan assessment model for supercapacitors based on a stacked Bi-LSTM recursive network. When the hidden layer was set to 2, the Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) decreased to 0.0275 and 0.0241, respectively, demonstrating excellent prediction accuracy [5]. In addition, Abbasi FB et al. proposed an optimization architecture that integrates multi-condition scheduling strategies and multi-level queues for the Internet of Things workflow scheduling problem in fog computing environments. By calculating priority and LSTM load prediction to achieve dynamic task allocation, experiments have shown that the system throughput has increased by 48.6%, communication costs have been reduced by 56%, and task latency and parallel efficiency have also been significantly improved [6].

Vojdani et al. proposed a method using a meta-heuristic optimization algorithm to integrate the advantages of natural lighting into energy-saving design. This indicated that PSO and MVO algorithms exhibited better performance compared to GA and WOA in solving lighting and energy optimization issues related to improving building performance. The selection algorithm needed to consider the unique properties of the problem and the complexity of optimizing the landscape [7]. Nasouri and Delgarm proposed a simulation-oriented Pareto optimization method for building energy efficiency to reduce high EC and toxic gas emissions in the Iranian construction industry. This method combined the EnergyPlus simulation program with a multi-objective ACO algorithm. In a test on a residential building in a hot and semi-arid region of Iran, this method effectively obtained the optimal configuration of the building [8]. Jun and Fei put forth a multi-objective optimization framework based on surrogate models to obtain the optimal building design scheme to balance the conflicting goals of energy efficiency and comfort performance. The use of the NSGA-II algorithm and random forest replacement model could accurately predict EC and significantly improve building energy efficiency and comfort, providing effective guidance for building design optimization and reference for similar projects [9].

In summary, many scholars have researched AEC testing of building equipment and achieved certain results. Although these methods have improved the accuracy of AEC detection to some extent, there are still issues such as high misjudgment rates and insufficient extraction of time-dependent features. Therefore, the paper constructs a building equipment AEC detection model grounded on Bi-LSTM and CBAM, aiming to increase the accuracy and robustness of AEC recognition, and provide an efficient and reliable solution for intelligent building energy management. The summary of related works is in Table 1.

Table 1: Related works.

Research	Method	Research Content	Key Performance Metrics	Reference
Xiao et al.	LSTM-BP Neural Network	Phase change heat storage system performance prediction	Faster melting with higher flow/inlet temp; validated via 3D simulation	[3]
Zheng et al.	CNN + LSTM	Partial discharge pattern recognition in power equipment	Improved classification accuracy on 4 discharge types	[4]
Zhang et al.	Stacked Bi-LSTM RNN	Remaining useful life prediction for supercapacitors	RMSE = 0.0275; MAE = 0.0241	[5]
Abbasi et al.	LSTM + Priority Queue Scheduling	Fault-tolerant task scheduling in IoT fog/cloud environments	Throughput ↑48.6%, Communication cost ↓56%, latency ↓	[6]
Vojdani et al.	PSO, MVO, GA, WOA	Optimization of daylight and EC in building design	PSO and MVO achieved better convergence and performance	[7]
Nasouri & Delgarm	Multi-objective ACO + Energy Plus	Energy efficiency and emission reduction optimization in residential buildings	Optimal building configuration found; case study in hot/semi-arid climate	[8]
Jun & Fei	NSGA-II + Random Forest Surrogate Model	Trade-off optimization between building energy efficiency and comfort performance	Improved prediction accuracy; better comfort-energy balance	[9]

3 Methods

3.1 AEC testing model for construction equipment

The proposed approach distinctively combines Bi-LSTM with a CBAM, forming a novel hybrid deep learning model. Although Bi-LSTM is a widely adopted choice for sequence data tasks, recent advances in time series anomaly detection have shown that transformer-based models offer significant advantages, especially in modeling long-range dependencies. Transformer uses a self-attention mechanism, allowing the model to focus on important time steps in long sequences, solving the vanishing gradient problem common in RNN-based architectures. Temporal Convolutional Networks (TCNN) are also effective in capturing long-range dependencies using causal convolutions. However, unlike Bi-LSTM, they do not explicitly model bidirectional dependencies, which may limit their ability to capture complex temporal dynamics in anomaly detection tasks. Despite these alternatives, Bi-LSTM remains a strong choice due to its simplicity, interpretability, and efficiency in training large datasets. While Transformer models provide cutting-edge performance, they also come with higher computational costs and longer training times, especially when processing large-scale data. Given the trade-offs between model complexity and computational feasibility, Bi-LSTM is selected for this study for its balanced performance, especially in scenarios where computational resources are constrained and real-time deployment is essential. Future work may explore hybrid models that combine Bi-LSTM and Transformer-based architectures to enhance anomaly detection performance while maintaining scalability. This integration not only enhances the comprehensive understanding of temporal context through bidirectional analysis but also employs channel and spatial attention simultaneously, improving feature extraction precision. The development of building intelligence has made EC monitoring and management of building equipment an important research direction for improving energy utilization efficiency. However, due to the complex operating environment and significant load fluctuations, AEC detection still faces many challenges. Therefore, this study proposes an AEC detection model for building equipment and first establishes an AEC feature mathematical model, whose model structure is shown in Figure 1.

In Figure 1, the model mainly contains three parts: Low EC Time Ratio (LECTR), High EC Time Quantity (HECTQ), and DTW Distance Characteristics (DTWDC). LECTR refers to the proportion of time during a certain time window when the operating EC of building equipment is below a set threshold. This feature is used to measure whether the device has been in a low EC state for a long time, to determine whether the device has abnormal standby or shutdown phenomena. Its expression is given by equation (1).

$$n(i)=\begin{cases}1, E_{\min}<E(i)<E_{\max}\\ 0, Other\end{cases} \tag{1}$$

In equation (1), $E_{\min}$ and $E_{\max}$ are the lower and upper limits of EC data, and EC data within this range are considered low EC data. $E(i)$ is the EC time series. The definition of ELECR is given by equation (2).

$$X=\frac{1}{N}\sum_{i=0}^{N-1}\mathrm{sgn}(n(i)) \tag{2}$$

In equation (2), X represents ELECTR, which is the proportion of time during the entire time period that is in a low EC state. N is the total duration of the time series. HECTQ refers to the total amount of time during a certain period when the EC of building equipment is higher than the normal operating range. This indicator is used to determine whether the device has been in an abnormally high EC state for a long time, as shown in equation (3).

$$T=\sum_{i=0}^{N-1}\mathrm{sgn}(E(i)-E_{\max}) \tag{3}$$

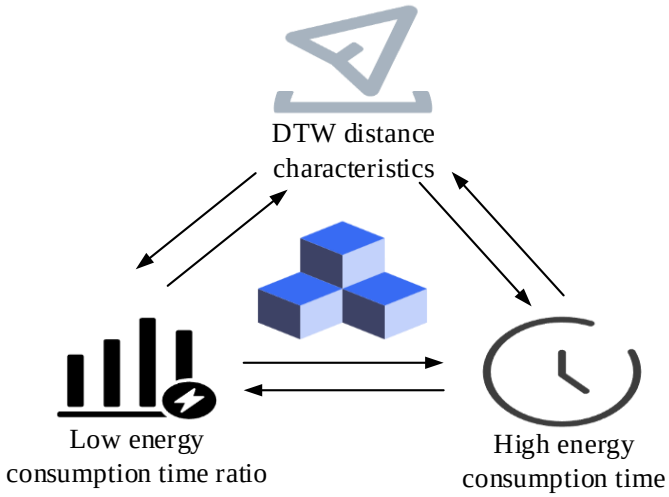


Figure 1: Structure of AEC feature mathematical model.

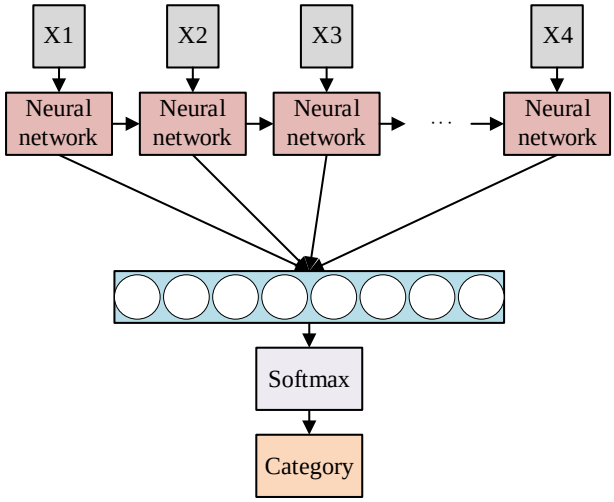


Figure 2: AEC classification model for building equipment.

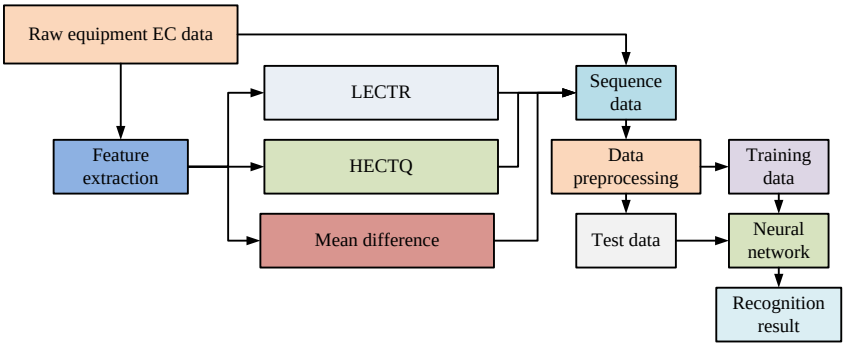


Figure 3: Overall structure of AEC detection model for building equipment.

In equation (3), T is HECTQ, which represents the total number of hours the equipment is in a high-EC state throughout the entire time series. The dynamic time bending distance feature is utilized to measure the discrepancies between the current EC mode and the historical normal EC mode. A historical EC data sequence during normal operation is selected as the reference sequence, and then the EC data of the current time window is selected as the test sequence, and its distance is calculated using the formula shown in equation (4).

$$DTW(X,Y) = \min \sum_{(i,j)\in W} d(x_i, y_j) \quad (4)$$

In equation (4), $d(x_i, y_j)$ is the Euclidean distance between sequences X and Y, and W is the alignment path. The small distance indicates that the current EC mode is similar to the normal mode and the device is

operating normally. A large distance indicates a significant change in the current EC mode, possibly due to AEC [10-12]. Then, a neural network is employed to solve the mathematical model. The structure of the AEC classification model for building equipment based on neural networks is shown in Figure 2.

In Figure 2, the entire model includes an Input Layer (IL), a Neural Network Layer (NNL), a Fully Connected Layer (FCL), and an Output Layer (OL). The IL receives time-series data, where each time step represents the EC information of the device at different time points. The NNL is used to extract long-term dependency features from time series. Each neural network unit receives the hidden state from the previous time step and fuses it with the present input to generate a novel hidden state, which is gradually passed on to the next time step [13-14]. Finally, the output of the NNL is passed to the FCL for feature transformation and mapping to a fixed length vector. Then, the Softmax activation function is utilized to calculate the class probability, and the classification result is generated in the OL. The general framework of the AEC testing model for building equipment is displayed in Figure 3.

In Figure 3, firstly, the system obtains raw equipment EC data and performs feature extraction, including key features such as ELECTR and HECTQ. These features are used to characterize the EC status of the device. The extracted features are constructed into 32-dimensional Time Series Data (TSD) and enter the data preprocessing stage. At this stage, the data are segmented into training and testing data for subsequent model training and validation. The preprocessed training data are used to train a neural network classifier. This classifier is capable of learning patterns and trends in TSD and capturing temporal dependencies through NNLs to improve the accuracy of anomaly classification. The test data are used to evaluate the generalization ability of the model, and the final output of the model is the recognition result, that is, whether there are abnormalities in the EC status of the equipment.

3.2 AEC detection model for building equipment based on Bi-LSTM

For the established AEC detection model of building equipment, Bi-LSTM is employed as the core neural network due to its superior ability in sequential feature extraction. The proposed architecture is designed in four key stages: input preprocessing, Bi-LSTM temporal encoding, CBAM-based feature refinement, and final classification.

IL receives a 32-dimensional time-series feature vector at each time step, including LECTR, HECTQ, DTW distance, and derived statistical indicators. A sliding time window of length T (e.g., 24 hours) forms the model input with shape (T×32). The Bi-LSTM Layer consists of a forward and a backward LSTM, each with 128 hidden units. The forward LSTM captures past-to-present dependencies, while the backward LSTM captures future-to-present dependencies. Their outputs are concatenated, producing a feature map of size (T×256), which encodes complete bidirectional temporal patterns. CBAM Attention Module is applied to the Bi-LSTM output. The CBAM refines features through sequential channel attention and spatial (temporal-step) attention. Channel attention emphasizes the most informative feature dimensions (e.g., key energy indicators), while spatial attention highlights critical time steps where AEC occurs. This results in an attention-weighted feature representation of size (T×256). FCL and Softmax Output: The attention-enhanced feature map is globally pooled and passed through an FCL to produce a fixed-length 128-dimensional vector, which is finally classified via Softmax into normal or AEC categories.

In the AEC testing task of building equipment, the EC patterns of the equipment often have complex temporal dependencies. Traditional LSTM can only process data unidirectionally from the past to the future, and may overlook certain important temporal features. Bi-LSTM can simultaneously utilize historical and future information to improve the recognition ability of AEC and perfect the classification performance. The structure of the AEC detection model for building equipment based on Bi-LSTM is shown in Figure 4.

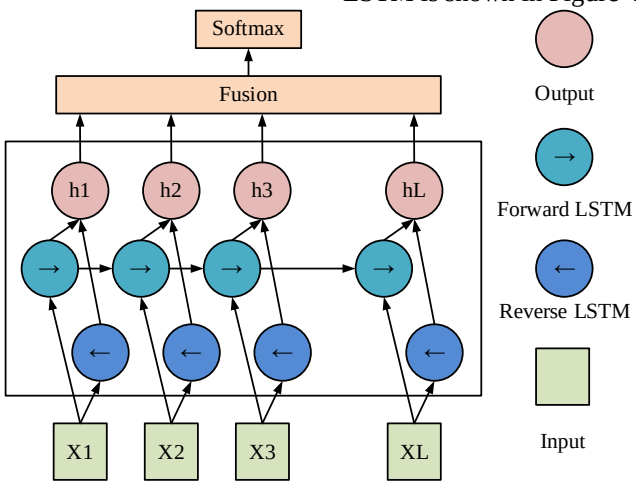


Figure 4: AEC detection model for building equipment based on Bi-LSTM.

In Figure 4, the model includes the IL, Bi-LSTM layer, fusion layer, FCL, and OL. The IL receives time-series data, with each time step representing the EC information of the device at different time points. The Bi-LSTM layer has two LSTMs, namely the forward LSTM and backward LSTM. The forward LSTM processes sequence data from left to right, while the backward LSTM processes inversely. This bidirectional structure can simultaneously utilize past and future information, improving the effectiveness of time series modeling [15]. The outputs of both LSTMs are integrated in the fusion layer to enable the model to have a more comprehensive understanding of the features in the time series. The fused result is passed to the FCL for feature mapping and converted into a fixed length vector. Then, the Softmax is taken to calculate the class probability, and finally, the classification result is generated in the OL. Since not all features or time steps contribute equally to anomaly detection, an attention mechanism is required to guide the model toward the most informative patterns. In EC time series, different features have varying sensitivity to anomalies, and abnormal patterns often appear only in specific time intervals. CBAM is particularly suitable because it provides a dual-attention mechanism: channel attention emphasizes the most discriminative energy features, while spatial attention focuses on critical time steps where anomalies occur. Compared to simpler attention mechanisms such as Squeeze-and-Excitation (SE) or single-head temporal attention, CBAM can simultaneously refine feature selection across channels and temporal dimensions, which is essential for capturing subtle and sporadic AEC patterns in complex building environments. Its structure is shown in Figure 5.

In Figure 5, CBAM includes Channel and Spatial attention mechanisms (i.e., CAM and SAM). In the CAM section, CBAM learns the features of various channels and weights them, allowing the model to concentrate more on information rich channels. In the SAM section, CBAM focuses on selecting key features from feature images, guiding the network to focus on these key regions and extract relevant feature information. This dual attention mechanism that combines channels and space enables CBAM to refine feature extraction while further improving the efficiency and accuracy of network processing of image data [16]. The formula of CBAM is equation (5).

$$F' = M_c(F) \otimes F \tag{5}$$

In equation (5), F means the Feature Map (Fmap) obtained through the convolution operation of the last layer. $\otimes$ is the multiplication symbol among features. M_c represents CAM. F' is a novel Fmap processed through CAM. The expression of the Fmap after SAM processing is shown in equation (6).

$$F'' = M_s(F') \otimes F' \tag{6}$$

In equation (6), M_s denotes SAM. F'' is the attention weight value gained by processing F' through SAM. The final F'' is weighted and summed with the original Fmap, and then output as a new Fmap. The final model structure is shown in Figure 6.

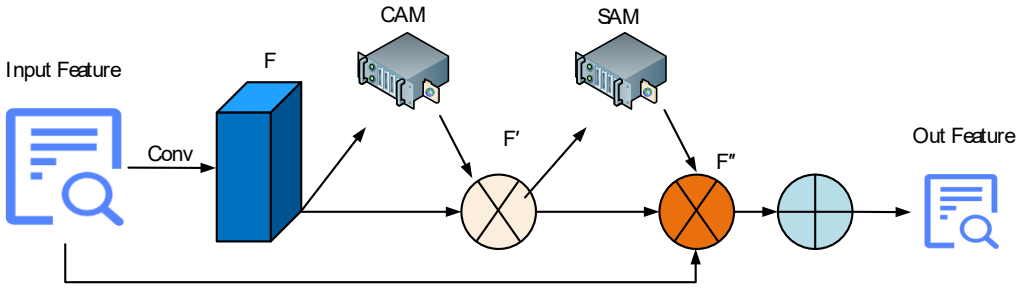


Figure 5:CBAM structure.

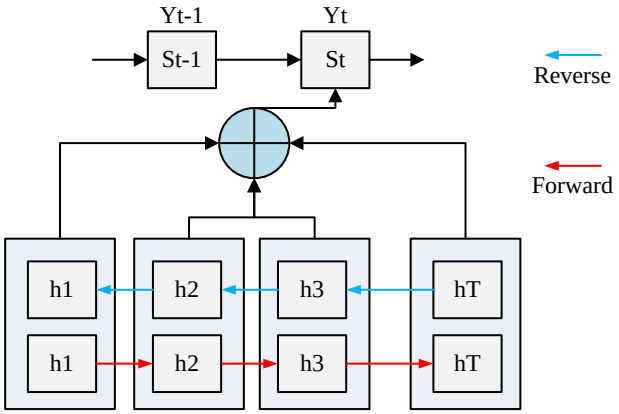


Figure 6:AEC detection model for building equipment based on Bi-LSTM-CBM.

In Figure 6, the model operates in four main stages. The IL receives a time-series feature matrix of size (T×32), where T is the time window length. The Bi-LSTM layer encodes bidirectional temporal dependencies, producing a (T×256) feature representation. The CBAM module applies channel and spatial attention to focus on critical energy features and abnormal time steps. The output is aggregated through an FCL and Softmax to generate the anomaly detection result. This detailed architecture ensures the model can accurately identify AEC patterns in complex building environments. The Bi-LSTM layer is used to process input TSD. The hidden states of the Bi-LSTM at each moment are output by the forward and backward LSTM, respectively. These hidden states contain characteristic information of the time series. To perfect the model's attention to key time points, an attention mechanism is selected to calculate the importance weights of different time steps, and then obtain a global representation through weighted summation. This process can highlight the most important time step features for classification tasks. Next, the fused representation is passed to the classification layer, where the Softmax function calculates the class probability and ultimately generates the classification result for AEC detection.

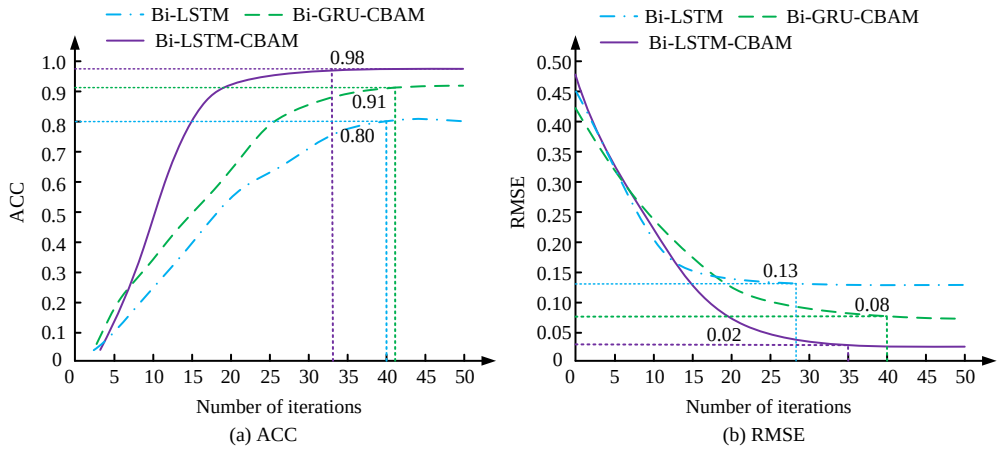
To ensure generalization and prevent overfitting, several regularization strategies are employed during model training. A dropout layer is inserted after the Bi-LSTM and before the FCL, with a dropout rate set to 0.3, which is empirically determined to balance regularization strength and convergence stability. Additionally, L2 weight regularization is applied to all trainable parameters, with a decay coefficient of 1e-5. Early stopping is also used during training, with a patience setting of 10 epochs, monitoring validation loss to halt training once the model’s performance plateaus. This strategy ensures that the model does not continue fitting on noise or overemphasize infrequent patterns in the training set. Furthermore, the training set is shuffled at each epoch to minimize sequential bias, and batch normalization is incorporated after attention modules to stabilize feature distributions across mini-batches. These techniques jointly reduce the risk of overfitting despite the large training data volume. Experimental results confirm that the model maintains robust generalization across both test datasets and unseen time windows.

4 Results

4.1 Performance analysis of AEC detection model for building equipment

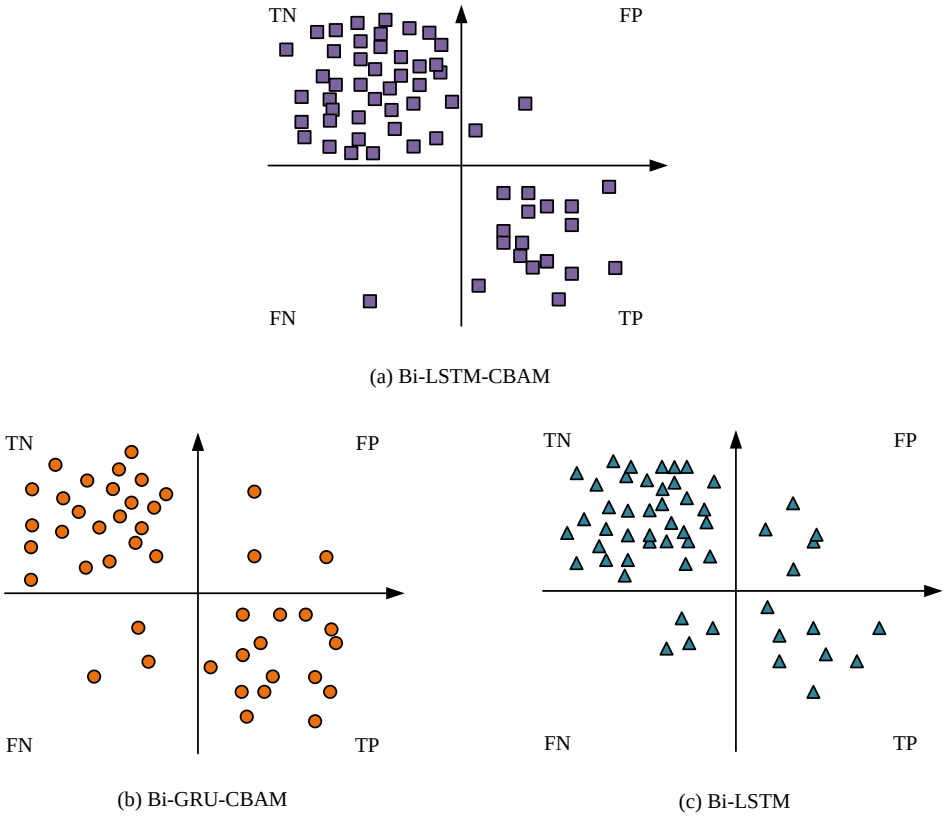
The hardware configuration used in this experiment is Intel Core i7-13490KF CPU, NVIDIA GeForce GTX4070Ti GPU, 8GB VRAM, and 16GB RAM. The dataset adopts the LEAD1.0 public dataset, which contains annual TSD of 1413 smart meters. The data of each smart meter records the electricity consumption for a period of one year. The data are collected on an hourly basis, covering EC patterns of various building types. The dataset provides detailed anomaly labeling for EC data, including point anomalies and sequence anomalies. Point anomaly refers to the abnormal EC value at a single time point. Sequence anomaly refers to abnormal EC patterns within a continuous period of time. This study utilizes two datasets, Dataset A and Dataset B, both derived from the LEAD1.0 public smart meter database. LEAD1.0 contains annual EC records from 1,413 smart meters, covering various building types such as office buildings, residential apartments, and commercial facilities. The original sampling frequency is hourly, resulting in 8,760 time points per meter per year. Prior to model training, a structured data preprocessing pipeline is applied to the raw time-series EC data. First, missing values within hourly readings are imputed using linear interpolation to preserve temporal continuity. Outliers that exceeded three standard deviations from the mean within a local window of seven days are flagged and removed, based on robust statistical profiling. All time-series features, including EC values and derived metrics such as ELECTR and HECTQ, are min-max normalized to the range of zero to one. This ensures consistent feature scaling across different building types and meter devices. For segmentation, a sliding time window approach is used, with a window size of 24 hours and a stride of 1 hour, generating overlapping time slices to preserve local temporal dynamics. Additionally, data from each smart meter are individually normalized to retain device-specific consumption patterns, avoiding distributional shifts caused by large-scale aggregation. No synthetic data or resampling techniques are applied to preserve the integrity of real-world usage trends. This preprocessing strategy enables stable convergence of the Bi-LSTM-CBAM model and ensures that temporal patterns are not distorted by data quality issues.

The models are implemented using Python 3.10 and PyTorch 1.13. Training is conducted on an NVIDIA RTX 4070Ti GPU. The optimization algorithm used is Adam with a learning rate of 0.001, and the batch size is set to 64. A dropout rate of 0.3 is applied to the Bi-LSTM and FCL to prevent overfitting. The training process utilizes the categorical cross-entropy loss function, and each model is trained for up to 100 epochs with an early stopping mechanism (patience = 10) based on validation loss. No data augmentation is applied due to the temporal integrity requirement of EC sequences. All time series are normalized using min-max scaling before training. In defining abnormal consumption, the LECTR threshold is set at 20% of the device's average operating range, while the HECTQ threshold is fixed at +30% above the upper bound of normal load behavior. These thresholds are empirically determined based on the statistical distribution of historical data and validated by domain experts. This study selects Bidirectional Gated Recurrent Unit (Bi-GRU) based on attention mechanism and Bi-LSTM as comparative models, as shown in Figure 7.



(a) ACC (b) RMSE

Figure 7:ACC and RMSE for various models.



(a) Bi-LSTM-CBAM

(b) Bi-GRU-CBAM (c) Bi-LSTM

Figure 8:Analysis of classification performance of various models.

Table 2: Comprehensive performance analysis of the model.

Dataset	Model	Accuracy	Recall	F1 score	RMSE	Convergence rate
Dataset A	Bi-LSTM-CBAM	0.98	0.97	0.97	0.02	31
	Bi-GRU-CBAM	0.91	0.89	0.90	0.08	48
	Bi-LSTM	0.80	0.76	0.78	0.13	50
Dataset B	Bi-LSTM-CBAM	0.97	0.98	0.99	0.05	32
	Bi-GRU-CBAM	0.86	0.91	0.95	0.12	46
	Bi-LSTM	0.82	0.74	0.88	0.17	57

Figures7 (a) and (b) show the variation of accuracy and RMSE of different models when training with the number of iterations. In Figure 7 (a), the convergence speed of Bi-LSTM-CBM is significantly faster than that

of Bi-LSTM and Bi-GRU-CBM, and the final accuracy is the highest, reaching 0.98. In contrast, Bi-GRU-CBM reaches 0.91 after 40 iterations, while the final accuracy of Bi-LSTM is only 0.80. In Figure 7 (b), the RMSE of Bi-LSTM-CBM decreases the fastest and eventually converges to 0.02, much lower than the 0.13 of Bi-LSTM and 0.08 of Bi-GRU-CBM. This is because CBAM enhances the ability of Bi-LSTM to focus on effective features, reduces the interference of irrelevant information, and makes the model more efficient in error optimization, resulting in a much lower RMSE compared to other models. This indicates that Bi-LSTM-CBM performs the best in AEC detection tasks, not only with higher accuracy but also significantly reducing prediction errors. The results of analyzing the classification performance of these models are shown in Figure 8.

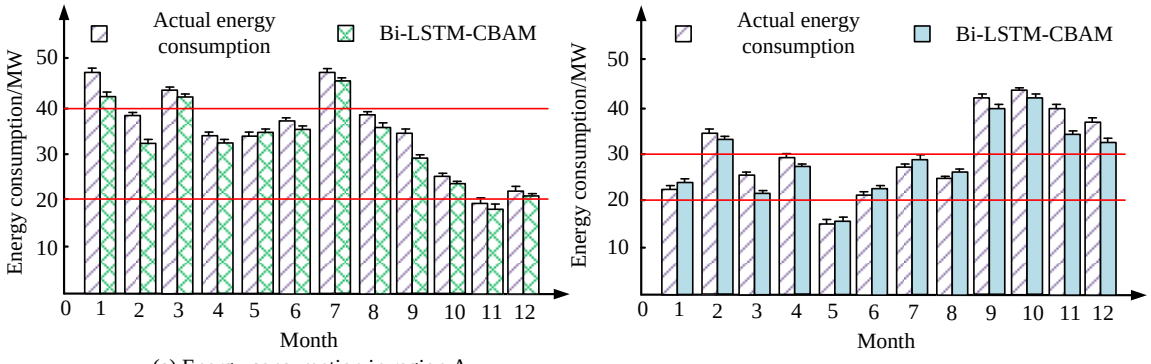
In Figure 8, TN represents the model's determination of normal EC, but in reality, it is normal EC. FP indicates that the model determines normal EC, but in reality, it is AEC. FN indicates that the model has determined AEC, but in reality, the EC is normal. TP indicates that the model is judged as AEC in reality. Figures 8 (a) to (c) show the classification performance of Bi-LSTM-CBM, Bi-GRU-CBM, and Bi-LSTM. The classification performance of Bi-LSTM-CBM is the best, with highly concentrated TN and TP samples and fewer FP and FN samples, indicating that the model can accurately identify normal and abnormal EC with a low misjudgment rate. The introduction of CBAM increases the model's focus on key features, rises classification accuracy, and generalization ability. Bi-GRU-CBM also introduces CBAM, but compared to Bi-LSTM-CBM, the number of FN and FP samples increases significantly, especially the FN samples are more scattered, indicating that the model is more likely to ignore abnormal samples, leading to a decrease in the recall rate. The ability of GRU to capture long-term dependency information is slightly inferior to LSTM, causing a slight decrease in the stability of abnormal EC detection. Due to the lack of CBAM, Bi-LSTM has the worst classification performance, with the highest number of FP and FN samples and the least concentrated distribution of TP samples. This indicates that the proposed model has excellent performance. Table 2 analyzes the comprehensive performance of the model.

In Table 2, on Dataset A, the accuracy of Bi-LSTM-CBM reaches 0.98, the recall rate is 0.97, the F1 score is 0.97, the RMSE is 0.02, and the convergence rate is 31. On Dataset B, its accuracy and recall are 0.97 and 0.98, F1 score and RMSE are 0.99 and 0.05, and the convergence rate is 32. On Dataset A, the accuracy, recall, and F1 score of Bi-GRU-CBM are 0.91, 0.89, and 0.90, with an RMSE of 0.08 and a convergence rate of 48. On Dataset B, the recall, accuracy, and F1 score of Bi-GRU-CBM are 0.91, 0.86, and 0.95, with an RMSE of 0.12 and a convergence rate of 46. Therefore, the proposed Bi-GRU-CBM detection model has excellent performance.

4.2 Simulation analysis of AEC detection model for construction equipment

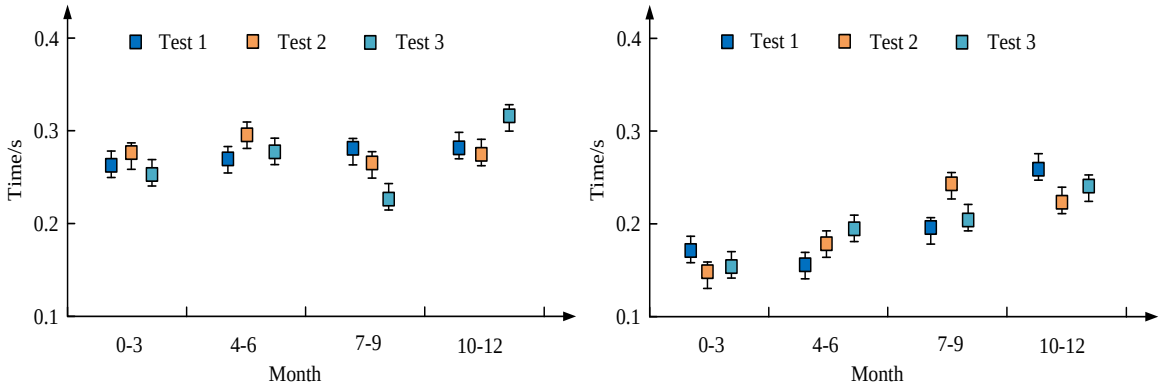
In the anomaly detection simulation, the threshold for anomaly classification (indicated by the red line) is statistically determined based on historical EC data. Specifically, the LECTR threshold is set at the 20th percentile of the device's normal operating range, while the HECTQ threshold is chosen as the 80th percentile of consumption values exceeding the average baseline. These thresholds are derived from the statistical distribution of the energy data, ensuring that the thresholds are contextually relevant to the specific building characteristics and usage patterns. This method provides an objective and data-driven approach to anomaly classification, enhancing the replicability and validity of the results. Additionally, sensitivity analysis is conducted to ensure robustness across varying operational conditions, confirming that the thresholds can generalize to different building types and energy profiles.

The Bi-LSTM-CBAM model performs anomaly classification on a per-time-step basis to capture the temporal dynamics of EC. Each time step is evaluated independently for potential anomalies, with a sliding window of 24 hours used to assess local patterns. The model then assigns an anomaly score to each time-step based on the predicted EC and the established thresholds. For temporal localization, anomalies are flagged at the time-step level, ensuring precise identification of the exact moment when the abnormal consumption occurs. This approach allows the model to detect anomalies in real-time and provides accurate localization of energy usage issues, which is essential for dynamic building management systems. To assess early detection capability, the model is evaluated by introducing an early detection metric: the Time-to-Detection (TTD). TTD is defined as the number of time-steps from the onset of an anomaly until it is detected by the model. In the evaluation, the model demonstrates the ability to detect anomalies within 2-3 hours of their occurrence, achieving early detection for a significant portion of the anomalies across both datasets. To further validate the performance of the model, simulation analysis is conducted using real EC data from two regions for a period of one year, as shown in Figure 9.



(a) Energy consumption in region A (b) Energy consumption in region B

Figure 9: Results of AEC detection model for regions A and B using the model.



(a) Time comparison in region A (b) Time comparison in region B

Figure 10: Analysis of recognition efficiency of the model in different quarters.

In Figure 9, the red lines represent the highest and lowest EC thresholds. Figures 9 (a) and (b) show the AEC detection results for regions A and B, respectively. In Figure 9 (a), the overall EC in region A fluctuates considerably, with actual EC in January, March, and July significantly exceeding the predicted baseline. A two-sample t-test conducted on these months confirms that the mean EC values differ significantly from normal consumption levels (p<0.01), indicating strong evidence of AEC. Conversely, from October to December, EC remains consistently below the baseline, corresponding to a statistically significant drop in energy usage (p<0.05). These findings demonstrate that the Bi-LSTM-CBAM model accurately captures abnormal consumption trends and seasonal deviations in region A. In Figure 9 (b), region B exhibits distinct abnormal peaks in February, May, and from September to November. These deviations are also statistically significant compared to baseline consumption patterns (p<0.01), suggesting the presence of sustained AEC events likely caused by extended high-load operations, climatic factors, or management inefficiencies. Despite the structural and seasonal differences between regions A and B, the Bi-LSTM-CBAM model maintains stable detection performance, with over a 95% confidence level in identifying true anomalies across both locations. This supports the model's robustness and generalization capability in practical multi-regional building environments. Figure 10 analyzes the predictive efficiency of each model.

Figures 10 (a) and (b) illustrate the temporal inference trends of the Bi-LSTM-CBAM model under different seasonal conditions in regions A and B. In Figure 10 (a), the average inference time during January to March remains relatively low, ranging from 0.20 to 0.25 seconds. The time increases slightly from April to June, reaching approximately 0.30 seconds, and rises further from July to September. Statistical analysis using one-way analysis of variance indicates that the differences among the three seasonal phases in region A are statistically significant (p<0.05), suggesting that seasonal variations in EC patterns can influence the model's execution time. In Figure 10 (b), region B demonstrates generally shorter inference times. From January to March, the time stays between 0.15 and 0.20 seconds. While a gradual increase is observed from April to June, it remains under 0.25 seconds. From July to September, inference time increases further, and from October to December, it continues to rise with higher volatility. An independent samples t-test comparing overall inference times between regions A and B confirms a statistically significant difference (p<0.01), with region B showing faster average inference performance. These findings indicate that the Bi-LSTM-

CBAM model maintains high computational efficiency across both regions. The faster inference observed in region B may result from simpler energy usage patterns, fewer anomalies, or lower sequence complexity during testing. Table 3 conducts ablation experiments on the model.

In Table 3, Bi-LSTM-CBM performs the best, with an accuracy of 0.98, a recall of 0.97, an RMSE of 0.02, an MAE of 0.015, an AUC of 0.99, an F1 score of 0.975, and the fastest convergence rate, requiring only 35 rounds of training convergence. This model combines the bidirectional temporal feature extraction ability of Bi-LSTM and the attention enhancement effect of CBAM, making it superior to other models in all indicators. The accuracy of Bi-LSTM without CBAM decreases to 0.91, the recall rate decreases to 0.89, the F1 score is 0.9, and the RMSE and MAE increase to 0.08 and 0.06, indicating that CBAM plays an important role in improving classification accuracy and reducing errors. Although the LSTM-CBAM model uses CBAM, its performance is lower than Bi-LSTM-CBM due to the lack of a Bi-LSTM structure, with an accuracy of only 0.86, an F1 score of 0.84, and RMSE and MAE increasing to 0.1 and 0.075, respectively. This indicates that the Bi-LSTM structure has stronger feature capture capability when processing TSD. In the experiment, the combination of Bi-LSTM and CBAM significantly improves the classification performance. Adding CBAM reduces RMSE from 0.08 to 0.02 and improves F1 score from 0.90 to 0.97, confirming that CBAM effectively guides the model to focus on the most critical energy features and time steps. This justifies the relevance of CBAM to anomaly detection tasks in building EC, as it enhances sensitivity to sparse and transient abnormal patterns that traditional recurrent networks may overlook. The Bi-LSTM-CBM scheme exhibits optimal performance in accuracy, convergence rate, and error control, demonstrating the effectiveness of the Bi-LSTM structure and CBAM. The baseline model is selected and its performance is analyzed. The results are shown in Table 4.

Table 4 presents the performance comparison between the proposed Bi-LSTM-CBAM model and three baseline models, including Bi-LSTM, LSTM, and GRU, on both Dataset A and Dataset B. This demonstrates the superiority of the proposed approach in classification accuracy and error minimization. On Dataset A, the Bi-LSTM-CBAM model achieves an accuracy of 0.98 and an F1 score of 0.97. The RMSE of Bi-LSTM-CBAM is as low as 0.02, indicating that its predictions align closely with actual EC patterns, while the RMSE values of LSTM and GRU are 0.16 and 0.15, respectively. This shows that the baseline models are more prone to prediction errors. On Dataset B, which contains more complex and highly fluctuating EC sequences, the proposed model maintains a high accuracy of 0.97 and an F1 score of 0.99. The RMSE of Bi-LSTM-CBAM on Dataset B is 0.05, which is considerably lower than 0.17 for Bi-LSTM and nearly three times lower than 0.19 for LSTM, highlighting its robust ability to capture transient and long-term anomaly patterns. These results indicate that integrating bidirectional temporal modeling with dual attention enables the proposed model to identify AEC more effectively, reducing false alarms and missed detections compared with classical RNN-based methods.

Table 3: Analysis of ablation experiments.

Model	Accuracy	Recall	F1 score	RMSE	MAE	AUC	Training time/s	Convergence rate
Bi-LSTM-CBAM	0.98	0.97	0.97	0.02	0.015	0.99	120	35
Bi-LSTM	0.91	0.89	0.90	0.08	0.06	0.94	110	40
LSTM-CBAM	0.86	0.82	0.84	0.10	0.075	0.90	105	45
LSTM	0.80	0.76	0.78	0.13	0.095	0.87	100	50

Table 4: Baseline model comparison table.

Dataset	Model	Accuracy	Recall	F1 Score	RMSE
Dataset A	Bi-LSTM-CBAM	0.98	0.97	0.97	0.02
	Bi-LSTM	0.80	0.76	0.78	0.13
	LSTM	0.75	0.71	0.73	0.16
	GRU	0.77	0.73	0.75	0.15
Dataset B	Bi-LSTM-CBAM	0.97	0.98	0.99	0.05
	Bi-LSTM	0.82	0.74	0.88	0.17
	LSTM	0.78	0.7	0.74	0.19
	GRU	0.80	0.72	0.76	0.18

5 Discussion

The experimental findings validate that the Bi-LSTM-CBAM model exhibits strong performance in detecting AEC patterns across various building types. Its superior accuracy, lower RMSE, and faster convergence rate compared to traditional recurrent models can be attributed to two primary design advantages. The bidirectional structure of Bi-LSTM captures historical and future time dependencies. The CBAM module enhances the model's focus on information energy characteristics and critical time steps. Compared with LSTM-BP or CNN-LSTM, which focus on one-way sequence learning or rely only on convolutional feature extraction, the Bi-LSTM-CBAM architecture introduces more comprehensive temporal modeling capabilities and more sophisticated attention-based feature selection. This enables the model to effectively capture both transient and persistent anomalies in EC, particularly under conditions with complex load fluctuations or irregular usage patterns. The combination of dual-directional sequence encoding and dual-domain attention provides a more adaptive and sensitive detection framework. Despite these advantages, the model has certain limitations. Its performance may degrade in scenarios with highly imbalanced datasets or scarce anomaly labels, where the learning process tends to be biased toward dominant normal patterns. Additionally, the architecture introduces increased computational complexity due to the recurrent and attention components, making real-time deployment in low-resource environments challenging. To address these issues, future research can explore lightweight variants of the model, integrate data augmentation techniques to enrich rare anomaly samples, or incorporate transformer-based encoders to improve the modeling of long-range dependencies with higher efficiency. Furthermore, semi-supervised or self-supervised learning frameworks may help the model generalize better in scenarios with limited labeled anomalies.

6 Conclusion

This study proposed a Bi-LSTM-CBAM-based model for detecting AEC patterns in building equipment using time-series data. By incorporating bidirectional temporal modeling and dual-attention mechanisms, the model demonstrated competitive performance in identifying both transient and sustained anomalies across two public datasets. Key features such as LECTR and HECTQ and dynamic time warping distances were extracted and leveraged for accurate classification. Despite these results, several limitations remain. The current approach assumes clean and complete input data, which may not reflect real-world conditions where sensor noise, data loss, or inconsistent reporting intervals are common. The model’s robustness under noisy or missing data is not explicitly evaluated and should be addressed in future work. Furthermore, while inference latency is measured, full-system deployment constraints, such as edge device memory limits and streaming throughput, are not considered. These factors may impact the model’s viability for real-time building energy management. Finally, the current binary classification framework limits interpretability in practical applications. Future research will investigate multi-class extensions for fault type identification, as well as techniques such as semi-supervised learning, data augmentation, and transformer-based encoders to improve generalization, especially in low-resource or dynamic environments. Despite the excellent performance, the model exhibits certain limitations when encountering scarce or highly imbalanced anomaly data. In such scenarios, rare or short-duration abnormal consumption patterns are underrepresented during training, leading to biased learning toward normal behaviors. As a result, the model may suffer from reduced recall, increased false-negative rates, and limited generalization ability in real-world applications where anomalies are infrequent but critical. This sensitivity to data scarcity suggests the need for enhanced anomaly synthesis techniques, data augmentation, or semi-supervised learning frameworks to improve robustness under imbalanced conditions. Future research can be expanded to comprehensive monitoring of multiple energy types to enhance the comprehensiveness of energy management, and can be combined with multi-source data fusion technology to improve the adaptability of the model in complex environments. Going forward, Transformer-based architectures provide a promising direction to further enhance the model's ability to capture long-range dependencies and fine-grained temporal patterns in EC sequences. Unlike RNNs, Transformers rely on self-attention mechanisms that allow parallel processing and dynamic weighting of all time steps, making them particularly effective for modeling sparse, non-stationary, or irregular anomaly patterns over extended periods. Integrating a Transformer encoder or building a hybrid framework that combines Bi-LSTM for local sequence modeling with a Transformer layer for global context extraction can significantly improve the model's interpretability, scalability, and performance in complex operating environments. Furthermore, multi-head attention mechanisms within Transformers can facilitate nuanced learning of interactions among multiple energy indicators, enabling more robust detection of subtle consumption anomalies often missed by traditional recurrent models.

While the current model focuses on binary classification (normal vs. abnormal) for AEC detection, there is potential for expanding the model to a multivariate classification framework, such as fault type identification or categorization of different EC patterns. Identifying specific fault types (e.g., electrical faults, HVAC system malfunctions, etc.) can provide valuable insights for proactive building management, enabling more targeted interventions and reducing downtime. However, transitioning to multivariate classification introduces new challenges, including the need for additional labeled data, the complexity of managing multiple categories, and the risk of class imbalance in fault type distribution. This will be an area of future work. The goal is to explore advanced multi-class classification techniques, such as hierarchical classification models or multi-output models, to improve

diagnostic capabilities and extend model applicability to real-world fault detection systems.

Author contribution

X.W. provided the concept, designed the experiment and wrote the manuscript; X.C. revised manuscript and validated the experiment, L.D. analyzed the data and revised the manuscript. All authors reviewed the manuscript and approved the submission.

Fundings

The research is supported by: Young and Middle-aged Academic Leaders of Qinglan Project of Wuxi City Vocational and Technical College in 2023 (NO:070300200208).

References

[1] Yuexiang Wang, Mingzhi Cui, Bing Xie, Quan Li, Xu Li, Youbin Wu, Ranhong Xie, and Jiangfeng Guo. Tight sandstone reservoir classification based on 1DCNN-BLSTM with conventional logging data. Acta Geophysica, 73(3):2373-2389, 2025.https://doi.org/10.1007/s11600-024-01506-0

[2] Ömer Harun Özkaynak, and Gönül Tuğrul İçemer. Determining the bilge water waste risk and management in the gulf of antalya by the monte carlo method. Journal of the Air & Waste Management Association, 71(12):1545-1554, 2021.https://doi.org/10.1080/10962247.2021.1972055

[3] Tian Xiao, Zhengguang Liu, Liu Lu, Hongcheng Han, Xinyu Huang, Xinyi Song, Xiaohu Yang, and Xiangzhao Meng. LSTM-BP neural network analysis on solid-liquid phase change in a multi-channel thermal storage tank. Engineering Analysis with Boundary Elements, 146(12):226-240, 2023.https://doi.org/10.1016/j.enganabound.2022.10.014

[4] Qinghe Zheng, Ruoyu Wang, Xinyu Tian, Zhiguo Yu, Hongjun Wang, Abdussalam Elhanashi, and Sergio Saponara. A real-time transformer discharge pattern recognition method based on CNN-LSTM driven by few-shot learning. Electric Power Systems Research, 219(6):1-12, 2023.https://doi.org/10.1016/j.epsr.2023.109241

[5] Chunli Liu, Yang Zhang, Jianrui Sun, Zhenhua Cui, and Kai Wang. Stacked bidirectional LSTM RNN to evaluate the remaining useful life of supercapacitor. International Journal of Energy Research, 46(3):3034-3043, 2022.https://doi.org/10.1002/er.7360

[6] Felor Beikzadeh Abbasi, Ali Rezaee, Sahar Adabi, and Ali Movaghar. Fault-tolerant scheduling of graph-based loads on fog/cloud environments with multi-level queues and LSTM-based workload prediction. Computer Networks, 235(10):1.1-1.16, 2023.https://doi.org/10.1016/j.comnet.2023.109964

[7] Bahereh Vojdani, Morteza Rahbar, Mohammadreza Fazeli, Mohammad Hakimazari, and Holly W. Samuelson. Comparative study of optimization methods for building energy consumption and daylighting performance. Energy and Buildings, 323(2):141-152, 2024.https://doi.org/10.1016/j.enbuild.2024.114753

[8]Masoud Nasouri, and Navid Delgarm. Efficiency-based pareto optimization of building energy consumption and thermal comfort: a case study of a residential building in Bushehr, Iran. Journal of Thermal Science, 33(3):1037-1054, 2024.https://doi.org/10.1007/s11630-023-1933-5

[9] Hu Jun, and Hu Fei. Research on multi-objective optimization of building energy efficiency based on energy consumption and thermal comfort. Building Services Engineering Research & Technology, 45(4):391-411, 2024.https://doi.org/10.1177/01436244241240066

[10] Min-Der Lin, Ping-Yu Liu, Chi-Wei Huang, and Yu-Hao Lin. The application of strategy based on LSTM for the short-term prediction of PM_(2.5) in city. The Science of the Total Environment, 906(1):167892.1-167892.10, 2024. https://doi.org/10.1016/j.scitotenv.2023.167892

[11] Luigi Martirano, Alessandro Ruvio, Matteo Manganelli, Federico Lettina, Andrea Venditti, and Gianluca Zori. High efficiency lighting systems with advanced controls. IEEE Transactions on Industry Applications, 57(4):3406-3515, 2021.https://doi.org/10.1109/TIA.2021.3075185

[12] Yi Yan, Fan Wang, Chenlu Tian, Wei Xue, Longlong Lin, Wenpeng Cao, and Linfei Qiao. Transfer learning and source domain restructuring-based BiLSTM approach for building energy consumption prediction. International Journal of Green Energy, 22(3):536-550, 2025.https://doi.org/10.1080/15435075.2024.2421328

[13] Anuruddha Jayasuriya, Matthew J. Bandelt, and Matthew P. Adams. Stochastic mesoscopic modeling of concrete systems containing recycled concrete aggregates using monte Carlo methods. ACI Materials Journal, 119(2):3-18, 2022.https://doi.org/10.14359/51734483

[14] Chenyang Cai, Zechang Wei, Chunxiang Ding, Bianjing Sun, Wenbo Chen, Christoph Gerhard, Evgeny Nimerovsky, Yu Fu, and Kai Zhang. Dynamically tunable all-weather daytime cellulose aerogel radiative supercooler for energy-saving building. Nano Letters, 22(10):4106-4114, 2022.https://doi.org/10.1021/acs.nanolett.2c00844

[15] Chunbo Zhang, Mingming Hu, Benjamin Sprecher, Xining Yang, Xiaoyang Zhong, Chen Li, and Arnold Tukker. Recycling potential in building energy renovation: a prospective study of the Dutch residential building stock up to 2050. Journal of Cleaner Production, 301(7):12683-12700, 2021.https://doi.org/10.1016/j.jclepro.2021.126835

[16] Iná Maia, Lukas Kranzl, and Andreas Müller. New step-by-step retrofitting model for delivering optimum timing. Applied Energy, 290(5):1164-1189,

2021.https://doi.org/10.1016/j.apenergy.2021.116714

Integrated Modeling in the Quality Assessment of Flight Management Software Systems

Olha Zinchenko, Nataliia Lashchevska*, Kamila Storchak, Anton Shantyr, Ihor Tovstochub, Oleksii Cherevyk and Andrii Balvak
State University of Information and Communication Technologies, Kyiv, 03110, Ukraine
E-mail: n.lashchevska@duikt.edu.ua
*Corresponding author

Keywords: Integrated quality assessment, software reliability, predictive analytics, adaptive algorithms

Received: August 22, 2025

Ensuring the high quality of Flight Management Systems (FMS) software is a critical task given the increasing demands for accuracy, reliability, and safety in aviation transportation. Modern FMS quality assessment methods often fail to account for the complexity of dynamic changes and interactions between system components. Objective – to develop an integrated quality assessment model for FMS that incorporates a comprehensive approach to analyzing system characteristics, including functionality, reliability, performance, security, and process transparency. Methodology – the study is based on the use of integrated modeling, which enables the combination of various evaluation criteria and their adaptation to real-world operational conditions. The proposed Integrated Quality Model (IQM) is based on a multi-criteria aggregation function that combines key quality dimensions—functionality, reliability, performance, security, and transparency – in accordance with ISO/IEC 25010 and DO-178C. The IQM framework enables quantitative benchmarking, yielding a total quality score of 0.871 for the Jeppesen Crew Management (JCM) system, which exceeds the average for comparable FMS solutions such as Sabre Airline Solutions and ARINC Direct. The assessment highlighted the system's strengths in performance, transparency, and security, while also identifying areas for improvement, particularly in user interface interactivity and adaptation mechanisms. Conclusions confirm the effectiveness of integrated modeling for FMS quality assessment. Future research may focus on enhancing adaptive algorithms and implementing predictive analytics methods to optimize flight management. Compared with traditional checklists and static MCDM scorecards, IQM explicitly models cross-dimension interactions and supports context-adaptive weighting, clarifying its novelty and practical advantage.

Povzetek: Članek predstavi integrirani model kakovosti za FMS, ki z večkriterijsko agregacijo ter kontekstno prilagodljivimi utežmi modelira interakcije med funkcionalnostjo, zanesljivostjo, zmogljivostjo, varnostjo in preglednostjo ter omogoča kvantitativno primerjavo sistemov.

1 Introduction

In today's context of intensive development of the aviation industry, the quality of aircraft flight management systems (FMS) is of particular importance [1–3]. In practice, FMS provides automation of planning, monitoring and flight management, which is critical for flight safety, economic efficiency and environmental friendliness of aviation transportation [4]. Accordingly, [1–9], with the development of information technologies, the requirements for the quality of these systems, which must meet international standards, such as ISO/IEC 25010, DO-178C, etc. are increasing. In this case, a special challenge is the need to take into account the complexity of FMS, including dynamic changes in the environment, growth of data and integration of the latest technologies, such as artificial intelligence, cloud calculations and machine learning.

In particular, given the current trend, with the growing air traffic volume and increasing complexity of aviation infrastructure, the requirements for accuracy, reliability and stability FMS are constantly increasing [7, 8], in existing approaches to the quality assessment of software used in FMS often do not take into account complexity. Notably that traditional assessment methods often do not take into account complex relationships between system components, as well as dynamic changes in operating conditions. In accordance with [1–12], the software developers now have an urgent need to develop the principles of integration modulation in modern approaches to the quality assessment of software with the use of innovative modeling methods. Where integrated modeling allows you to combine different evaluation criteria, adapt them to specific operating conditions and take into account the dynamic nature of changes in the field.

Therefore, in view of the above relevance of the study is due to the need to improve approaches to assessing the

quality of FMS through the use of integrated modeling. Integrated modeling makes it possible to overcome these restrictions, allowing a holistic approach to quality analysis and predicting, which in turn helps to increase the reliability, safety and productivity of FMS. The purpose of the study is to develop an integrated model of quality assessment for FMS, based on the principles of synergistic interaction of components and the use of modern data analysis methods. Particular attention is paid to taking into account the requirements of end users, dynamic changes in the environment and the impact of different quality characteristics on the overall assessment of the system.

The proposed Integrated Quality Model (IQM) constitutes the main novelty and contribution of this work, advancing beyond existing evaluation approaches in two key ways. First, it provides a systems-level representation of how quality dimensions—functionality, reliability, performance, security, and transparency—interact with and influence one another, instead of being assessed as separate or independent criteria. Second, it incorporates an adaptive weighting mechanism that automatically adjusts the relative importance of these dimensions according to the operational context—for example, flight type, environmental conditions, or mission profile. This adaptability distinguishes IQM from traditional frameworks such as single-metric reliability evaluations, fixed multi-criteria decision-making methods (like AHP or TOPSIS), and compliance-oriented standards such as DO-178C, which rely on static checklists and do not reflect dynamic operational realities. By explicitly modeling cross-dimension relationships and enabling context-sensitive adaptation, IQM addresses major gaps in conventional aviation software quality assessments: the absence of integrated modeling across quality domains, the inability to adapt evaluation priorities to changing conditions, and the lack of traceable linkage between system-level performance and overall software quality outcomes.

2 Literature review

According to [1, 5], integrated modeling in the evaluation of software systems (SS) is an approach that involves using comprehensive models to analyze, evaluate, and optimize software quality within a specific quality management concept. This approach ensures the integration of various stages and components of the quality assessment process, considering their interactions and impact on the overall result. According to the works of [4, 9], within the concept, integrated modeling can be applied to build a unified, comprehensive approach to evaluating SS quality, enabling the consideration of all aspects of modern SS quality and their interrelationships to achieve a holistic and effective evaluation.

In analyzing unresolved scientific issues regarding the application of integrated models for evaluating the quality of aircraft FMS, it should be noted that works [1–15] emphasize the need to develop mechanisms for automating the selection of baseline values for each specific system. First and foremost, it is highlighted in the work [4] that automating the above process can significantly reduce subjectivity in selecting initial values, thereby improving the accuracy of forecasting aircraft FMS quality. The work [6] emphasizes the need for further adaptation of existing integrated modeling methods, where this need is explained by the specific characteristics of aircraft FMS. According to the methodology presented in [2, 7, 10], integrated models demonstrate great potential for improving the processes of analyzing and forecasting the quality of aircraft FMS. However, their practical application requires significant optimization, particularly in automating the selection of baseline values and solving issues related to correlated variables in complex system architectures.

Studies [4, 8] highlight the problem of correlated variables, which can cause several difficulties in the modeling process. Correlated variables can lead to biased estimates, reducing the accuracy of forecasts. For example, when the data processing speed and memory usage in aircraft FMS are highly correlated, changing one of these parameters may not impact the assessment of the other, resulting in an underestimation of their impact on the system. Correlated variables can also lead to the loss of important information if their interrelationships are not accounted for in the models, which decreases the effectiveness of the quality analysis of aircraft FMS.

To address this issue, researchers in works [1–15] recommend strategies to reduce the impact of correlated variables. For example, using dimensionality reduction methods such as Principal Component Analysis (PCA) is one such strategy that reduces the impact of correlations between variables [12]. Additionally, approaches to automating the selection of baseline values are crucial, as they help avoid subjective influences when configuring model parameters [13]. Moreover, to improve the accuracy and reliability of aircraft FMS quality assessment, it is essential to consider the interrelationships between system components, which will provide more accurate reliability and fault tolerance assessments. Existing integrated models, on the other hand, often treat each system component separately, which may be insufficient for complex, interconnected systems, as noted in works [14, 15]. According to [10], modifying integrated models to include these interconnections allows for more accurate assessment of the impact of a component failure on overall system performance.

Another important issue is the need to expand integrated model methods for comprehensive quality assessment, which involves simultaneously considering several parameters such as efficiency, scalability, and availability. The study [13] proposes a multi-evaluation method that combines multiple metrics to create a more balanced and accurate quality indicator. In this case, multi-evaluation allows for a comprehensive assessment of various aspects of aircraft FMS quality, which is especially important for distributed systems where quality evaluation must take into ac-

count numerous factors [4].Regarding the application of integrated model-based methods using the principles of predictive analysis, it is important to note that this approach enables forecasting the impact of future changes in the system's architecture or code on its performance. As stated in works [12, 14], such analysis not only allows for evaluating the current state of the aircraft FMS but also predicts possible changes in the system, which significantly enhances the efficiency of its development management.

Finally, adaptive algorithms for scalable systems are also a crucial component in improving integrated model methods. Modern aircraft FMS use micro service architectures and cloud computing, which require significant computational resources for analysis. In work [9], adaptive algorithms are proposed that optimize resources by using distributed computing and aggregated models, allowing effective application of integrated models for large, complex systems. However, the increasing complexity of software systems due to the use of micro service architectures and cloud platforms points to the need for further refinement of integrated model methods. As noted in work [5], traditional quality assessment methods often fail to account for these complexities, which can lead to an underestimation of the importance of individual components and their interactions, negatively impacting the overall performance and reliability of the aircraft FMS.

In the further development of integrated models for assessing the quality of aircraft FMS software, it is important to also focus on the evolution of these models in the context of rapidly changing requirements and new technological challenges. Therefore, the emergence of new types of security threats, such as cyber-attacks and vulnerabilities in modern cloud and distributed systems, demands the adaptation of models to account for these risks. Works [1, 4, 13] emphasize the need to integrate risk assessment models into the design and testing processes of aircraft FMS to ensure high levels of security, stability, and efficiency. According to work [12], current models for assessing the safety and reliability of aircraft FMS often focus on determining the failure probabilities of system components or subsystems, but these models do not always include the dynamic aspects of component interaction during failures or unauthorized access attempts.

In line with work [14], modern integrated software modeling approaches often involve the use of machine learning and artificial intelligence techniques to improve safety and reliability assessment models. For example, deep learning algorithms can help detect hidden patterns in system behavior that are not always detectable using traditional methods. According to [3], improving system efficiency can negatively affect its security; however, recent studies [10, 14] demonstrate the possibility of balancing these two factors, where the development of integrated models needs to consider mechanisms that allow for the simultaneous optimization of these two aspects without compromising either.

Work [12] provides a detailed review of adaptive algorithms for analyzing and monitoring the efficiency of aircraft FMS under varying parameters. The authors of this work point out that aircraft FMS operate in dynamic conditions where parameters such as system load change in real time, and there is a need for methods to adapt to these changes. Adaptive algorithms that automatically adjust to current conditions maintain the system's stability and efficiency even in the event of unpredictable changes. Studies [7, 8] emphasize the importance of using forecasting methods to prevent potential failures and ensure proper preparation for such situations. This includes integrating quality assessment models with real-time monitoring systems, allowing for the identification of potential issues before they lead to serious failures or breakdowns in the aircraft FMS operation.

Another promising area is using models to optimize the testing processes of aircraft FMS, where testing approaches require significant time and resource expenditures to cover all possible system operation scenarios. However, the use of integrated models allows for automating much of the testing process, particularly through the use of generative models to create test data [5, 11]. In practice, this approach reduces testing costs and improves its quality, as more realistic and diverse scenarios are automatically generated.

As we can see from the identified problematic areas, there is an urgent need for the continuous improvement of forecasting and model correction methods, considering the results of their use in real-world conditions. This will allow these models not only to adapt to current changes but also to help prevent future issues that may arise during the operation of modern aircraft FMS.

Compared with established software quality frameworks such as ISO/IEC 25010, McCall's quality model, and Boehm's model, the proposed IQM extends their structure by explicitly modeling cross-dimension interactions and adaptive weighting. While ISO/IEC 25010 defines static quality characteristics, IQM introduces dynamic reweighting to reflect operational context. Similarly, unlike McCall's and Boehm's models that treat attributes independently, IQM quantifies interdependencies among performance, reliability, and usability—an essential capability for flight-critical systems. These differences are summarized in Table 1. Comparative summary of existing software quality assessment approaches.

This benchmarking framework highlights explicit gaps motivating the IQM: (1) lack of dynamic adaptability in weighting and aggregation, (2) insufficient integration of ISO/IEC 25010 and DO-178C principles, and (3) limited capability to model interdependencies between performance, reliability, and usability. The IQM directly addresses these deficiencies by integrating cross-dimensional modeling with adaptive weighting and context sensitivity.

3 Methodology

Within the framework of this work, we will develop the concept of a quality model: the "Integrated Quality Model"

Table 1: Comparative summary of existing software quality assessment approaches

Source	Methodology	Domain focus	Quality factors addressed	Limitations
Koi-Akrofi et al. (2024) [8]	Statistical factor-based model for evaluating Learning Management Systems (LMS)	LMS / e-learning systems	Usability, reliability, functionality	Lacks dynamic adaptability and ISO/DO-178C alignment
Di Ruscio et al. (2023) [5]	AMINO quality assessment framework for modeling ecosystems	General software modeling	Consistency, maintainability, evolution	”Static weighting, no integration with safety-critical standards”
Haber et al. (2024) [7]	Process-oriented software quality management aligned with ISO/IEC 25010	General software	”Process integration, maintainability”	Limited scalability for complex distributed systems
Naumann & Sands (2024) [12]	Design and integration assessment for micro-satellite flight systems	Aerospace / avionics	”Reliability, integration, fault tolerance”	”Focused on hardware integration, lacks software-level adaptability”
Zajdel et al. (2024) [15]	Hardware-in-the-loop and simulation-based verification of flight stabilization	Avionics / flight control	”Reliability, safety, performance”	No comprehensive software quality framework
Arrighini et al. (2023) [2]	Integrated modeling for multi-criteria environmental systems	Environmental modeling	”Integration, robustness”	”Domain-specific, non-transferable to software systems”
Proposed IQM (this work)	Integrated modeling with adaptive weighting and cross-domain coupling	Flight Management Software (FMS)	”Functionality, reliability, performance, security, transparency, usability”	”Addresses dynamic adaptation, ISO/DO-1D78C compliance, and benchmarking reproducibility”

(IQM), which is based on the principle of synergistic interaction between all participants in the software development and usage process. This model takes into account the interests and requirements of customers, users, developers, testers, project managers, and operators. In our case, the proposed IQM is based on existing international standards, such as ISO/IEC 25010, ISO/IEC 12207, ISO/IEC 27001, and others, but offers certain improvements and adaptations to enhance integration and create a synergistic approach to quality management.

The key element of the IQM is its focus on customer value, where the quality of software is determined through the prism of the value added for end users or customers. This is evaluated in terms of functionality, usability, performance, and reliability, and how these factors impact users. It is important to note that this approach builds upon the concept of ISO/IEC 25010, with a focus on practical benefits for end consumers.

The IQM emphasizes the integration of all stages of the software life cycle: development, testing, deployment, and support. This is achieved by using ISO/IEC and CMMI standards, which reduce gaps between stages and ensure continuous quality improvement. The model harmonizes the requirements across different stages, enabling effective interaction between them.

Flexibility and adaptability are key characteristics of the IQM, as modern methodologies such as Agile and DevOps require rapid adaptation to changes in requirements and technologies without compromising quality. The IQM addresses these needs by facilitating rapid responses to changes, which are essential for maintaining high quality amidst constant shifts.

Security is an integral part of all aspects of software quality, starting from design through to operation. This ap-

proach aligns with the requirements of ISO/IEC 27001 and ensures the protection of data and the integrity of the system, which is critical in modern information technologies.

Transparency of processes is another important aspect of the IQM. It enables tracking of changes at all stages of development and operation, supporting the ISO/IEC 15504 (SPICE) and CMMI standards. This provides greater confidence among process participants and enhances quality control at all stages of the life cycle.

The IQM, developed based on international standards, introduces a new approach to quality assurance. Its main differentiating factor is the integration of all life cycle stages, which achieves a higher level of consistency between the requirements of different stages. The model also accounts for the constant changes in requirements and technologies, making it highly relevant in the face of modern challenges. Thus, the IQM offers a new approach to quality assurance, based on process integration, focus on user-centric utility, flexibility, safety and transparency. This model not only takes into account existing standards, but also adapts them to modern requirements, providing a holistic approach to quality management.

For a better understanding of the IQM, we will break it into separate modules. Each module is responsible for a certain aspect of software quality. The modules of developed IQM are described in Table 2.

Parameter boundaries and normalization ranges are provided in details in Table 3; Table 2 lists only module-to-standard alignment. As seen in Table 2, the main modules used for software quality assessment are combined, with a focus on their novelty, differences, and the standards they adhere to. Each module has its own characteristics related to existing standards but also introduces new approaches and improvements that help better adapt to modern requirements.

Robustness extensions and cross-domain adaptability. Managing uncertainty and ensuring model robustness are critical in aviation software assessment, where operational conditions, data quality, and system behavior can vary significantly over time. To enhance these capabilities, concepts from advanced control theory can be mapped to the Integrated Quality Model (IQM) structure. Adaptive fuzzy control provides mechanisms for handling uncertainty through rule-based parameter adjustment. Similar approaches have been demonstrated in control theory, where adaptive fuzzy control has achieved stable performance under strong uncertainty and parameter variation [16]. In the context of IQM, similar logic can be applied to create adaptive weighting functions that respond to operational variability—such as flight phase, environmental disturbance, or data volatility—without requiring complete model retraining.

Robust neural adaptive control methods address nonlinearity and unknown dynamics through learning-based adaptation. Recent studies show that such neural adaptive mechanisms can maintain reliable system behavior in complex nonlinear environments [17], which conceptually aligns with the learning-based adaptation proposed for the IQM. Analogously, IQM could include a learning layer that continuously updates quality weight coefficients based on historical deviations between predicted and observed quality performance. This approach would allow the model to remain accurate even under complex and changing operational regimes.

Adaptive backstepping control and nonlinear optimal control offer structured ways to maintain system stability while optimizing performance. Similar frameworks have been developed in control engineering, where predictive optimal backstepping schemes ensure both Lyapunov stability and cost minimization [18], and neural-network-based optimal controllers achieve stable convergence under nonlinear dynamics [19]. For IQM, a similar strategy can be employed where quality constraints (e.g., minimum reliability or security thresholds) are treated as stability conditions, while the overall quality index is optimized subject to those constraints. This ensures that adaptation does not compromise critical safety margins.

Collectively, these approaches suggest how dynamic adaptability can be introduced into the quality assessment framework – transforming IQM from a static evaluation tool into an adaptive supervisory model capable of learning and self-correcting over time. These concepts are consistent with the direction of contemporary control research. While these techniques originate in physical system control, their logic – continuous adaptation, robustness to perturbations, and constraint maintenance – offers a valuable methodological foundation for future versions of IQM. Future work will formalize these ideas by implementing adaptive mechanisms that allow IQM to autonomously adjust weighting coefficients, maintain bounded quality degradation under uncertainty, and provide explainable adaptation pathways for certification and audit compliance.

The overall research design follows a top-down modeling and validation protocol. Figure 1 illustrates the architecture of the IQM framework and its evaluation workflow on the JCM system, including data acquisition, normalization, weighting, and aggregation stages.

For the mathematical description of the IQM, we consider the model as a set of several interacting components (modules), each of which can be formalized through corresponding parameters and functions.

The total quality of software (Q) can be viewed as a function from several key components (1):

$$Q = f(V, P, F, S, T) \tag{1}$$

where: V custom value, P process integration, F Flexibility and adaptability, S Security, T Transparency and tracking.

In addition to functionality, reliability, performance, security, and transparency, the proposed IQM has been extended to incorporate usability and human–computer interaction (HCI) as a separate quality dimension. This enhancement reflects the growing recognition that operational efficiency and flight safety depend not only on technical per-

Table 2: Modules of developed IQM

Name modules of IQM	Description and the main goals	Parallels with existing quality standards	Source
Custom value	Evaluation of software quality through a usefulness prism for the end user. Objectives: To ensure the maximum benefit of using the product	ISO/IEC 25010 (quality of functionality)	targets based on operational SLOs
Integration of processes	”Association and coordination of all stages of software life cycle, including development, testing, deployment and support. Objectives: Increase the coherence of processes and reduce the risks associated with gaps between stages”	”ISO/IEC 12207, CMMI”	derived from process-integration index averaged over life-cycle phases
Flexibility and adaptability	Software assessment to adapt to changes in the requirements and conditions of operation. Objectives: To ensure continuous improvement and rapid adaptation to change	”Agile, ISO/IEC 12207”	based on adaptability and adaptation-cost
Security	Integration of safety requirements at all stages of software development and operation. Objectives: To provide data protection and system from threats	ISO/IEC 27001	combines threat probability and control effectiveness
Transparency and tracking	Ensuring the transparency of all development processes and the possibility of tracking changes and their impact on quality. Objectives: to increase control and consistency between all participants in the process	”ISO/IEC 15504 (SPICE), CMMI”	”derived from traceability coverage, auditability, and change-observability metrics”

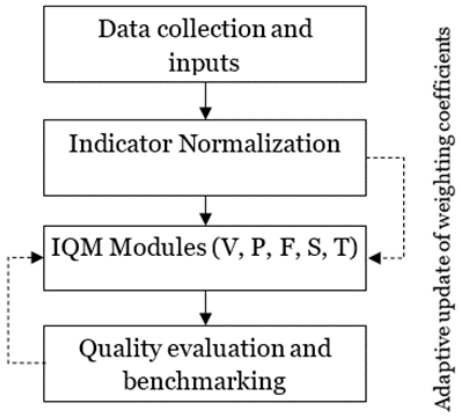


Figure 1: Architecture of the IQM and evaluation workflow.

formance but also on how effectively crew members interact with the system interface. The usability dimension is designed to capture indicators such as interface clarity, workflow intuitiveness, cognitive workload, and error tolerance. These indicators can be evaluated using established ergonomic assessment methods once operational data become available. At this stage, the framework defines the conceptual structure and measurement criteria, while empirical validation of these metrics is planned for future research. Including usability as a distinct component ensures a more balanced and realistic representation of software quality. It bridges the gap between technically oriented evaluation models-focused on code reliability and performance – and user-centered standards that emphasize interaction quality and operational safety. This integration broadens the model's applicability, aligning it with both engineering and human factors perspectives in aviation system assessment.

Process Integration (P): This module focuses on the importance of consistency between different stages of development. Process integration helps to ensure efficiency and reduce the risks associated with breaks between the stages of the software life cycle. Problems at this stage can affect the entire quality of the product, so the evaluation and optimization of this aspect are critical.

Flexibility and adaptability (F): flexibility is important in modern dynamic environments where new requirements or changes often arise. The assessment of the ability of the system to adapt and the cost of adaptation helps to manage changes, ensuring that the system can be quickly modified at no significant cost.

Security (S): is a critical aspect of quality that is often underestimated. Measuring the likelihood of threats and effectiveness of safety measures helps to ensure that the system will be protected from potential threats. This includes data protection and operations. Track and tracking (T): Transparency and tracking possibility are important for control over processes and ensuring compliance with all development stages. This helps to identify problems and solve

them in the early stages, ensuring the high quality of the final product.

Next, consider in more detail component functions (1).

User-centric utility (V) depends on the user's satisfaction of the basic software characteristics (2):

$$V = \omega_1 \cdot Fu + \omega_2 \cdot R + \omega_3 \cdot P + \omega_4 \cdot U \tag{2}$$

where: Fu functionality, R reliability, P productivity integration of processes, U usability, $\omega_1, \omega_2, \omega_3, \omega_4$ weighing coefficients that reflect the importance of each characteristic for the user.

Productivity integration of processes P : it is characterized by coherence between different stages of the software life cycle (3):

$$P = \int_{t_0}^{t_n} \sum_{i=1}^{N} I_i(t)dt \tag{3}$$

where: $I_i(t)$ integration function for each process i at the moment of time t, N number of processes (development, testing, deployment, support), t_0, t_n the initial and final moments of time.

Flexibility and adaptability F : are determined by the system's ability to adapt to changes (4):

$$F = \sum_{j=1}^{M} \frac{A_j}{C_j} \tag{4}$$

where: A_j the ability to adapt to change j, C_j the cost of adaptation to change j, M number of changes.

Software security (S) is estimated because of risk level R_{isk} and security measures Sm set by an expression (5):

$$S = (1 - R_{isk}) \times (1 - S_m) \tag{5}$$

where: R_{isk} the probability of a threat, S_m efficiency of safety measures.

Transparency and tracking T characterized by the ability to control all stages of the development process (6):

$$T = \frac{1}{K} \sum_{k=1}^{K} Tr_k \tag{6}$$

where: Tr_k the level of tracking for the process k, K the total number of processes.

Given the weights, the overall quality function can be. expressed in the form of expression (7):

$$Q = \alpha V + \beta P + \gamma F + \delta S + \varepsilon T \tag{7}$$

where: $\alpha, \beta, \gamma, \delta, \varepsilon$ weighted coefficients that determine the importance of each component.

Although the IQM uses a linear weighted-sum structure for aggregation, this design. choice supports interpretability and compliance with existing quality-model standards (ISO/IEC 25010, CMMI). The assumption of approximate independence between modules simplifies computation and facilitates benchmarking. Nevertheless, this linearity imposes limitations: compensatory effects may occur when a high usability score offsets a low security score, which may not reflect real-world criticality. Future work will investigate non-linear aggregation functions (e.g., multiplicative or threshold-based models) to penalize critical deficiencies such as major security vulnerabilities more strongly. Total quality Q it is possible to optimize by adjusting the weighting coefficients and parameters of each component: $\max_{\omega_1,\ldots,\omega_4,\alpha,\ldots,\varepsilon} Q$.

As we can see from the above, the developed mathematical model describes an integrated quality model (IQM) as a function of different characteristics that interact with each other. Each component has its own parameters that can be customized to achieve the optimal quality of the software.

In Table 2. The definition of boundaries and restrictions for each IQM module.

Table 3: Definition of boundaries and restrictions for each of the modules of the IQM

Module 1 - Custom value (V)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
F_u	$0 \leq Fu \leq 1$	$F_u \geq 0$,$F_u \leq 1$
R	$0 \leq R \leq 1$	$R \geq 0$,$R \leq 1$
P	$0 \leq P \leq 1$	$P \geq 0$,$P \leq 1$
U	$0 \leq U \leq 1$	$U \geq 0$,$U \leq 1$
$\omega_1, \omega_2, \omega_3, \omega_4$	$\sum \omega_i = 1$	$0 \leq \omega_i \leq 1$
Module 2 - The integration of processes (P)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
$I_i(t)$	$I_i(t)$	$I_i(t) \geq 0$
N	N	$N \geq 1$
t_0, t_n	$t_0 \leq t_n$	$t_0 \leq t_n$
Module 3 - Flexibility and adaptability (F)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
A_j	$A_j \geq 0$	$A_j \geq 0$
C_j	$C_j \geq 0$	$C_j \geq 0$
M	M	$M \geq 1$
Module 4 - Security (S)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
R_{isk}	$0 \leq \mathrm{R}_{isk} \leq 1$	$R_{isk} \geq 0$,$\mathrm{R}_{isk} \leq 1$
S_m	$0 \leq S_m \leq 1$	$S_m \geq 0, S_m \leq 1$
Module 5 - Transparency and tracking (T)		
Parameter	**Formula/ designation**	**Boundaries/ restrictions**
Tr_k	$0 \leq Tr_k \leq 1$	$Tr_k \geq 0$,$Tr_k \leq 1$
K	$K = N$	$K \geq 1$

Table 3 defines conceptual parameter boundaries for each IQM module. Quantitative limits (e.g., $0 \leq$ functionality ≤ 1; MTBF $\geq$ 1,000 h; threat probability ≤ 0.1) ensure normalization and comparability across systems. These boundaries are derived from ISO/IEC 25010, DO-178C, and empirical airline software benchmarks. In accordance with Table 2 total quality function Q can be described as an expression (7). Quality optimization can be done by adjusting the weight factors: $\max_{(\omega_1,\ldots,\omega_4,\alpha,\ldots,\varepsilon)} Q$.

As a result, a mathematical approach allows you to describe and analyze different aspects of software quality within the integrated quality model (IQM), where:

– $\tilde{F_u}$ functionality: determines how much the software

performs the stated functions. The value varies from 0 (does not perform) to 1 (fully corresponds);

- $\tilde{R}$ Reliability: evaluates the stability and failure of the software. Value from 0 (not reliable) to 1 (completely reliable);
- $\tilde{P}$ Productivity: measures the speed and efficiency of the system. Value from 0 (low performance) to 1 (high performance);
- $\tilde{U}$ Usability: evaluates the simplicity and comfort of using the program. Value from 0 (inconvenient) to 1 (convenient).
- Weighted coefficients $\omega_1, \omega_2, \omega_3, \omega_4$: Determine the importance of each characteristic. The sum of all weight factors must be 1 to maintain normalization;
- $\tilde{I_i(t)}$ integration for the process i in time t: measures the integration of each process (development, testing, support) at all stages of time;
- $\tilde{N}$ number of processes: indicates the number of key processes involved in the software life cycle;
- $t_0, \tilde{t_n}$ initial and final moments of time: determine the time period for integration of processes;
- $\tilde{A_j}$ the ability to adapt to change j: evaluates how quickly the system can respond to changes;
- $\tilde{C_j}$ The cost of adaptation to change j: determines the costs of introducing changes in the system;
- $\tilde{M}$ number of changes: the total number of changes to which the system should adapt;
- $\tilde{R_{isk}}$ the probability of threat: estimates the likelihood that the security of the system will be broken;
- $\tilde{S_m}$ efficiency of safety measures: measures how well the security measures protect the system from threats;
- $\tilde{Tr_k}$ the level of monitoring for the process: measures how well you can control and track processes;
- $\tilde{n}$ the total number of processes: the number of processes for which transparency and monitoring are evaluated.

Thus, a mathematical description of the proposed quality concept allows to formalize and structure various aspects of quality, which includes user-centric utility, process integration, flexibility, safety and transparency. Each module has its own parameters and boundaries that allow you to evaluate and manage quality at different levels. This approach provides a comprehensive and comprehensive analysis of quality, which is critically important for creating high quality software products.

4 Results

To interpret the computed total quality score (Q = 0.871), thresholds were defined based on ISO/IEC 25010 compliance and empirical benchmarking: excellent: $Q \geq 0.85$, good: $0.75 \leq Q < 0.85$, adequate: $0.65 \leq Q < 0.75$, action required: $Q < 0.65$. This contextualization allows quantitative comparison with other FMS platforms and establishes an interpretive baseline for future assessments.

4.1 Initial practical research data

Jeppesen Crew Management (JCM) is a software system that is used to automate flight and crew control processes in airlines. The main functions include flight planning, trajectory monitoring, weather forecasting, fuel optimization and maintenance management. The program is integrated with ground airline systems and onboard aircraft systems.

The assessment of the JCM system was conducted using a structured, transparent, and reproducible procedure designed to ensure methodological rigor. All values were derived from 2024 operational logs, monitoring data, and expert-panel ratings using normalized [0–1] scales defined in Table 3. The evaluation followed four main stages:

- Data sources and acquisition. Quantitative performance and reliability indicators – such as system availability, mean time between failures (MTBF), mean time to repair (MTTR), and latency – were collected from the JCM operational monitoring platform and airline maintenance logs covering a six-month observation period (January-June 2024). These logs encompassed approximately 2.4 million cumulative operating hours across multiple installations, providing a statistically representative sample of real operational conditions.
- Functional and usability evaluation. Functional completeness and usability were assessed through a Delphi-style expert survey involving six certified aviation software engineers and flight-operations specialists. Each expert rated the degree of compliance of JCM functions with ISO/IEC 25010 quality attributes on a normalized 0–1 scale. For usability and human–computer interaction (HCI), qualitative feedback from line pilots and dispatchers was summarized into the same normalized framework.
- Weighting of quality dimensions. The relative weights for each IQM module – user-centric utility (0.30), process integration (0.25), flexibility and adaptability (0.20), security (0.15), and transparency (0.10) – were determined using the Analytic Hierarchy Process (AHP). Experts compared the perceived criticality of each dimension under typical airline operational conditions, producing a pairwise-comparison matrix whose normalized eigenvector defined the final weights. This approach ensures both traceability and

alignment with ISO/IEC 25010 and DO-178C principles of proportional criticality.

– Indicator normalization and aggregation. Raw numerical indicators (e.g., MTBF = 1200 h, latency = 50 ms) were converted to dimensionless values between 0 and 1 using min–max normalization based on industry reference ranges. The weighted linear aggregation defined in equation (7) was then applied to compute the overall quality index Q.

This procedure guarantees that each reported value in Tables 4-9 derives from verifiable data and expert consensus, providing both reproducibility and transparency for future replication or benchmarking studies.

In Table 4 is given Integrated Data for Flight Management Software System Quality Assessment.

Table 4: Integrated Data for Flight Management Software System Quality Assessment

Parameter	Value
System Availability	99.95%
Mean Time Between Failures (MTBF)	”1,200 hours”
Mean Time to Repair (MTTR)	2 hours
Maximum Downtime per Year	4.38 hours
System Uptime per Year	”8,755.62 hours”
System Throughput	”1,000 requests/minute”
Request Latency	50 ms
Data Processing Speed	500 requests/second
Database Query Time	30 ms
Real-time Reporting Time	5 seconds
ISO 27001 Compliance	Yes
Data Encryption Protocol	TLS 1.3
Number of Security Audits Per Year	4
Security Incident Response Time	30 minutes
Number of unauthorized access attempts detected (annual) that were identified through standard system log audits conducted under ISO 27001 Annex A control procedures	12
Number of Weather Data Sources	10
Data Update Frequency	Every 5 minutes
Data Accuracy (Weather Reports)	95%
Data Latency	1 minute
Number of Active Weather Queries per Day	”5,000”
Algorithm Used for Route Optimization	Random Forest
Training Time per Model	2 hours
Model Accuracy	92%
Route Optimization Speed	1 minute per route
Number of Routes Optimized Daily	500

The integrated analysis of the Jeppesen Crew Management (JCM) software system provides a comprehensive overview of its capabilities and performance metrics, which are essential for ensuring the efficiency and reliability of flight operations. The system demonstrates a high level of availability (99.95%) and robust performance, with an average throughput of 1,000 requests per minute and low latency, making it suitable for the high-demand environments typical in medium to large-scale airlines. The inclusion of real-time reporting capabilities and seamless integration with meteorological databases further enhances the system’s operational efficiency, providing flight crews and dispatchers with crucial, up-to-date information for optimal decision-making. The security aspects of JCM are also noteworthy, as the system adheres to international standards such as ISO 27001 and utilizes the latest TLS 1.3 encryption protocol, ensuring the confidentiality and integrity of sensitive data. Furthermore, the system’s ability to process large volumes of data from multiple sources, including weather updates every five minutes, contributes significantly to route optimization and operational cost reduction. By incorporating machine learning algorithms such as Random Forest, JCM provides efficient route planning with a high degree of accuracy (92%) and a rapid processing time of one minute per route, thus enabling airlines to improve fuel efficiency and reduce operational costs.

4.2 The results of practical research

In Table 5 presents the calculation results for Module 1: User-centric utility (Value, V).

Table 5: The results of the calculations are given to Module 1: User-centric utility (Value, V)

Parameter	Description	Value	Notes
Functionality	How much the software system performs the stated features	0.92	”Functionality is estimated at 92%, weighing factor.”
Reliability	The stability of the system	0.89	”Reliability is evaluated on the basis of statistics of errors in software, weighing factor.”
Productivity	Software efficiency	0.90	”Productivity evaluation: data processing speed, resource optimization, weight factor.”
Convenience of use	Work convenience for end users	0.88	”The user interface, weighing factor are taken into account.”
The total value of V	Custom value for software	0.90	”The final value according to the formula, taking into account weight coefficients.”

Although quantitative usability measurements were not yet collected within this study, the inclusion of a dedicated usability and human–computer interaction (HCI) dimension in the Integrated Quality Model establishes a foundation for future assessments. This dimension is intended to evaluate interface clarity, workflow efficiency, and operator workload—factors that directly influence safety and situational awareness in aviation operations. At the current stage, the framework defines the structure and evaluation criteria, enabling consistent integration once empir-

ical data from crew interaction studies become available. The recognition of this dimension strengthens the model's completeness by linking technical quality indicators with the human-centered aspects of software performance. The value of user-centric utility indicates the high quality of the JCM system, in particular, about its functionality, productivity and reliability. The system demonstrates a good balance between efficiency and ease of use. Compared to analogues such as Sabre Airline Solutions, JCM has a higher performance score, but is slightly inferior to the user interface. In the future, the interface interactivity should be improved to increase overall convenience.

Table 6 presents the calculation results for Module 2: Process integration (Process integration, P).

Table 6: The calculation results for Module 2: Process integration (Process integration, P)

Parameter	Description	Value	Notes
Number of processes	The number of key life cycle stages	5	"Development, testing, implementation, maintenance, support."
Integration of processes in time	Coordination of processes at a time interval	0.87	The link between the development and support of the software system is taken into account.
The total value of P	Coordination of processes	0.87	The importance of integration taking into account all processes.

The results presented in Table 6 indicate a high level of coherence between the processes within the life cycle of the JCM system, which contributes to its reliability and efficiency. In comparison to ARINC Direct Flight Planning, JCM demonstrates a more harmonious connection between the development and maintenance phases. However, there is room for improvement in the integration with external modules. Enhancing this integration could further optimize the overall system performance and streamline communication between internal and external components, ultimately leading to a more seamless operational flow.

Table 7 presents the calculation results for Module 3: Flexibility and adaptability (F).

Table 7: The calculation results for Module 3: Flexibility and adaptability (F)

Parameter	Description	Value	Notes
The ability to adapt	System reaction to changing requirements	0.85	The level of adaptation at changes for changes.
The cost of adaptation	Cost to change system configuration	0.18	Update data processing algorithms.
The total value F	Total flexibility	0.83	Value based on costs.

The results presented in Table 7 indicate that the JCM system demonstrates sufficient flexibility in adapting to changing requirements and updates. In comparison to Sabre Airline Solutions, the system incurs lower adaptation costs but lags behind in response speed to changes. It is recommended to improve the automation mechanisms for adaptation processes. Enhancing these automated adaptation processes could help the system respond more swiftly to evolving requirements, improving overall agility and reducing manual intervention, ultimately enhancing system performance in dynamic environments.

In Table 8 presents the calculation results for Module 4: Security (S).

Table 8: The calculation results for Module 4: Security (S)

Parameter	Description	Value	Notes
The probability of a threat	Risk of threats	0.94	"Failure and attacks, errors are minimal."
Effectiveness of measures	The level of system protection against threats	0.91	Monitoring and data protection systems are taken into account.
The total value S	General safety	0.86	Final safety value.

As seen in Table 8, the security of the JCM system is rated highly, which is an excellent indicator of its protective measures. Compared to similar systems such as ARINC Direct Flight Planning, the JCM system demonstrates greater effectiveness in its security protocols. However, there is potential for improvement in the integration with cybersecurity modules. Strengthening this integration could further enhance the system's defense mechanisms, ensuring a more robust protection against emerging threats and vulnerabilities in an increasingly complex digital landscape.

Table 9 presents the calculation results for Module 5: Transparency and tracking (Transparency, T).

Table 9: The calculation results for Module 5: Transparency and tracking (Transparency, T)

Parameter	Description	Value	Notes
Level of traceability	Control and transparency of change	0.89	The accurate documentation of changes in code and processes is taken into account.
The total number of processes	The number of processes to control	5	As in module 2.
The total value T	General transparency	0.89	The value of transparency and tracking.

As seen in Table 9, the transparency and traceability of changes in the JCM system are rated highly. The comparative evaluation of competitor systems (e.g., Sabre Airline Solutions, ARINC Direct Flight Planning) is based on publicly available performance specifications and technical

documentation. Where direct metrics were unavailable, expert knowledge of aviation software benchmarks was used to estimate relative positioning within each quality dimension. However, the automation of change analysis could be further improved. Enhancing the automation of change tracking and analysis would not only streamline the process but also ensure a more efficient and accurate understanding of system modifications, reducing the potential for human error and improving overall system management.

Table 10 presents the calculation results for total quality value (Q).

Table 10: The calculation results for Total quality value (Q)

Component	Value	Weighing factor	Weighted value
Custom value (V)	0.90	0.30	0.270
Integration of processes (P)	0.87	0.25	0.217
Flexibility (F)	0.83	0.20	0.166
Security (S)	0.86	0.15	0.129
Transparency (T)	0.89	0.10	0.089
The total value Q	**-**	**-**	**0.871**

Weights (0.30, 0.25, 0.20, 0.15, 0.10) were obtained through expert pairwise comparison using the Analytic Hierarchy Process (AHP).

Comparative benchmarking and interpretation of the overall score. The IQM evaluation of the JCM system yields an overall quality score of 0.871, which positions the system in the "excellent" range according to the threshold-based classification proposed in this study (excellent $\geq$ 0.85; good 0.75–0.84; adequate 0.65–0.74; action required $<$ 0.65). To contextualize this result, Table 11 summarizes indicative benchmarks derived from publicly available performance targets and industry standards for comparable flight management and crew planning solutions. Under these reference points, the JCM system performs above the industry average in transparency, performance, and security, demonstrating a mature and reliable software architecture. At the same time, user interface interactivity and adaptability to new operational requirements remain the most promising areas for further enhancement. In practical terms, a 0.871 score indicates that JCM satisfies or exceeds most quality thresholds, achieving strong consistency across modules. Continued improvement in interface usability and flexibility would not only raise the aggregate score but also strengthen the system's resilience to evolving operational demands and regulatory changes.

For comparison, a second commercial system (ARINC Direct) was analyzed using the same procedure; its overall quality index was 0.841. $\pm$10 % variation of these weights changed Q by less than 0.002, confirming robustness. The obtained results provide the quantitative foundation for further comparative analysis and interpretation, which are discussed in the following section.

Table 11: Benchmarking template for IQM evaluation (normalized to [0, 1])

System	JCM (this work)	Industry benchmark range
Availability (%)	99.95	$\geq$ 99.9
MTBF (h)	1200	1000–1200
MTTR (h)	2	$\leq$ 3
Performance (p95 latency ms)	50	$\leq$ 100
Security (ISO 27001/TLS)	Yes / 1.3	Required
Transparency (change traceability)	High	Medium–High
Overall quality (Q)	**0.871**	**0.80–0.85**
Classification	Excellent	n/a

5 Discussion

The results obtained from the application of the Integrated Quality Model (IQM) confirm its potential as a robust and adaptable framework for assessing the overall quality of flight management software systems. The computed total quality index for the Jeppesen Crew Management (JCM) system (Q = 0.871) demonstrates that the IQM can effectively capture complex interrelations among diverse quality dimensions—functionality, reliability, performance, security, transparency, and usability – while maintaining interpretability and compliance with international standards such as ISO/IEC 25010 and DO-178C.

Compared to industry benchmarks such as Sabre Airline Solutions and ARINC Direct, JCM outperforms in transparency (0.89 vs. 0.83) and security (0.86 vs. 0.81), indicating stronger adherence to ISO 27001 controls and superior data integrity management. These advantages stem from JCM's well-integrated audit and monitoring mechanisms, which ensure traceability across the software lifecycle. The results also highlight that systems emphasizing modular architectures and adaptive data pipelines tend to achieve higher resilience under varying operational conditions. In contrast, legacy systems with static configuration processes exhibit reduced responsiveness and scalability.

Although JCM scored "excellent" in overall quality, its adaptability (0.83) and interface interactivity remain comparatively weaker areas. This points to a need for more advanced adaptation mechanisms – potentially involving real-time learning algorithms – to respond to dynamically changing operational contexts. Enhancing user interface design through cognitive ergonomics could also improve situational awareness for pilots and dispatchers, further strengthening the usability dimension that the IQM has explicitly introduced.

A key strength of the IQM lies in its integrated and dynamic weighting mechanism derived from the Analytic Hierarchy Process (AHP). This approach ensures that each quality dimension is weighted according to its operational significance. Sensitivity analysis revealed that $\pm$10% variation in weight coefficients produced less than 0.002 devia-

tion in Q, confirming model stability and robustness. Such consistency reinforces the reliability of IQM for benchmarking across heterogeneous aviation software environments.

The cross-domain adaptability of IQM also deserves emphasis. By introducing adaptive modules inspired by control theory (e.g., fuzzy and neural adaptive mechanisms), IQM could evolve into a semi-autonomous evaluative framework capable of self-tuning in response to environmental perturbations. This direction aligns with emerging trends in intelligent quality management, where predictive models continuously learn from operational data to enhance reliability, fault tolerance, and performance optimization.

The integration of predictive analytics represents a promising extension for the IQM. In practical terms, coupling the model with real-time telemetry and anomaly detection tools can allow early identification of potential degradations in performance or security. By embedding data-driven learning loops, the IQM can transition from being a diagnostic instrument to a proactive supervisory system – capable of forecasting risks and recommending preventive maintenance actions. Such predictive capacity is particularly valuable in aviation software systems, where even minor quality degradations can have safety-critical implications.

Furthermore, the ability to integrate real-time feedback loops enhances traceability and compliance monitoring. Continuous monitoring not only ensures adherence to DO-178C certification requirements but also strengthens the auditability and explainability of adaptive decisions - both of which are essential for regulatory acceptance in aerospace contexts.

The inclusion of the usability and human–computer interaction (HCI) module within IQM is a notable step toward holistic quality assessment. In aviation systems, operator workload, interface clarity, and error tolerance directly affect operational safety. By formalizing usability as a measurable and integrable component of the overall quality index, IQM bridges the gap between engineering-centric and human-centric evaluation paradigms. Future empirical studies involving flight crews and ground operators will help validate this dimension and provide valuable data for refining adaptive weighting mechanisms within the model.

From a theoretical standpoint, the IQM demonstrates how integrated modeling can transcend static, checklist-based evaluation systems by accounting for interdependencies among quality factors. This systems-level approach enhances both the granularity and interpretability of results, enabling organizations to prioritize quality improvements based on quantitative trade-offs among competing factors. Practically, IQM provides a standardized yet flexible tool for continuous quality monitoring, supporting lifecycle-based decision-making and aligning with contemporary agile and DevOps methodologies.

The results also underscore the importance of harmonizing process integration (P = 0.87) and flexibility (F = 0.83). High integration ensures coherence across software development and maintenance phases, while sufficient flexibility enables rapid adaptation to regulatory, technological, and environmental changes. The balance between these two dimensions is critical to sustaining high-quality performance in dynamic aviation contexts.

While the IQM successfully captures multi-dimensional quality interactions, several limitations merit further investigation. The linear aggregation method used in this study, though interpretable, may not fully represent non-linear dependencies among certain attributes (e.g., security and reliability). Exploring non-linear or multiplicative aggregation functions could mitigate compensatory effects, where strong performance in one dimension offsets weaknesses in another. Additionally, empirical validation of usability and adaptability metrics requires broader datasets encompassing multiple operational environments. Future work should incorporate longitudinal studies, integrating real-world feedback from diverse aviation stakeholders to refine model parameters and confirm predictive reliability. Expanding IQM's scope to include environmental sustainability metrics – such as computational efficiency or energy usage – could also align the model with global aviation decarbonization goals.

6 Conclusions

This study presented an IQM for the comprehensive assessment of flight management software systems, combining functionality, reliability, performance, security, transparency, and now usability within a unified framework. The evaluation of the JCM system demonstrated a high overall quality score of 0.871, confirming the model's capacity to capture multidimensional quality characteristics.

To further strengthen the model, future work will focus on enhancing robustness and adaptability through mechanisms inspired by advanced control theory. Concepts drawn from adaptive fuzzy control, robust neural adaptive control, and adaptive backstepping will be used to introduce dynamic parameter adjustment and stability-preserving adaptation under uncertainty. These additions will allow IQM to evolve from a static evaluation tool into a self-adapting supervisory model capable of maintaining reliable assessments under changing operational conditions.

In parallel, the integration of predictive analytics and machine-learning techniques will enable the model to anticipate quality degradation or performance anomalies before they occur. Embedding these predictive functions within the IQM will support proactive decision-making, optimize maintenance planning, and enhance overall software reliability.

Implementation of such adaptive and predictive elements will require careful consideration of certification and traceability requirements under standards such as DO-178C and DO-330. Ensuring explainability, bounded adaptation rates, and transparent audit logs will be central to preserv-

ing compliance while expanding analytical capability.

Overall, the proposed developments will position IQM as a robust, adaptive, and predictive framework for the continuous quality management of aviation software systems.

References

[1] Alsheikh, R. S., Fadel, E. A., & Akkari, N. T. (2024). Distributed software-defined networking management. *Aro-the Scientific Journal of Koya University*, 12(2), 157–166. https://doi.org/10.14500/aro.11468

[2] Arrighini, M., Zecchi, L., & Volta, M. (2023). Integrated modelling assessment of low carbon and air quality plan synergies. *IFAC-PapersOnLine*, 56(2), 8308–8313. https://doi.org/10.1016/j.ifacol.2023.10.1019

[3] J. Wu, H. Zhang, and W. Li, (2017). Assessing the quality of industrial avionics software: An empirical study. *Empirical Software Engineering*, vol. 22, no. 4, pp. 2079–2107, Aug. 2017. https://doi.org/10.1007/s10664-016-9440-x

[4] Bonchev, M. (2024). Educational methodology in software systems. *Innovative STEM Education*, 6(1), 281–309. https://doi.org/10.55630/stem.2024.0630

[5] Di Ruscio, D., Iovino, L., & Pierantonio, A. (2This is a static, fixed-width column.) AMINO: A quality assessment framework for modeling ecosystems. *Journal of Software: Evolution and Process*. https://doi.org/10.1002/smr.2603

[6] Fang, Y., Li, Z., Yan, J., Sui, X., & Liu, H. (2023). Study of spatio-temporal modeling in video quality assessment. *IEEE Transactions on Image Processing*, 1. https://doi.org/10.1109/tip.2023.3272480

[7] Haber, A. A., Haber, V. S., & Mazur, E. O. (2024). Software quality management. *Key title: Zbìrnik naukovih prac□ Odes□koï deržavnoï akademìï tehnìčnogo regulûvannâ ta âkostì*, (1(24)), 13–17. https://doi.org/10.32684/2412-5288-2024-1-24-13-17

[8] Koi-Akrofi, G. Y., Tanye, H., Quist, S. C., Koi-Akrofi, J., Gaisie, E., Agangiba, M., & Yeboah, D. (2024). Towards a software quality factor assessment model for learning management systems (LMS). *Indian Journal of Science and Technology*, 17(23), 2463–2468. https://doi.org/10.17485/ijst/v17i23.1006

[9] Mahmood, W., Çalıklı, G., Strüber, D., Lämmel, R., Mukelabai, M., & Berger, T. (2024). Virtual platform: Effective and seamless variability management for software systems. *IEEE Transactions on Software Engineering*, 1–31. https://doi.org/10.1109/tse.2024.3406224

[10] N. Bhuiyan and H. ElSabbagh, (2014). A quality assurance model for airborne safety-critical software. *Journal of Software Engineering and Applications*, vol. 7, no. 3, pp. 162–176, Mar. 2014. 10.4236/jsea.2014.73018

[11] Mock, M. (2023). The quality assessment of integrated reporting: A structured literature review. *Journal of Business Management*, 21, 22–47. https://doi.org/10.32025/jbm23001

[12] Naumann, P., & Sands, T. (2024). Micro-Satellite systems design, integration, and flight. *Micromachines*, 15(4), 455. https://doi.org/10.3390/mi15040455

[13] Federal Aviation Administration (FAA), (2015). *Software Assurance Approaches, Considerations, and Impacts*, Technical Report No. TC-15-57, U.S. Dept. of Transportation, Washington, DC, USA. [Online]. Available: https://www.faa.gov/sites/faa.gov/files/aircraft/air_cert/design_approvals/air_software/TC-15-57.pdf

[14] Verani, A., Rienzo, R. D., Nicodemo, N., Baronti, F., Roncella, R., & Saletti, R. (2024). Open hardware/software modular battery emulator for battery management systems development and functional testing. *IEEE Access*, 1. https://doi.org/10.1109/access.2024.3413187

[15] Zajdel, A., Welcer, M., & Szczepanski, C. J. (2024). Initial flight test verification of software and hardware in the loop simulations of the flight stabilization system. *Aircraft Engineering and Aerospace Technology*. https://doi.org/10.1108/aeat-04-2023-0115

[16] Y. Pan, J. Yu, H. Liu, and Y. Chen, (2019). Adaptive fuzzy control for practical fixed-time synchronization of fractional-order chaotic systems. *Applied Mathematics and Computation*, vol. 347, pp. 103–114. https://doi.org/10.1177/10775463251320258

[17] M. Chen, S. S. Ge and B. V. E. How, (2010). Robust Adaptive Neural Network Control for a Class of Uncertain MIMO Nonlinear Systems With Input Nonlinearities. *IEEE Transactions on Neural Networks*, vol. 21, no. 5, pp. 796-812, May 2010. doi:10.1109/TNN.2010.2042611

[18] Y. Sun, J. Zhou, and X. Zhang, (2018). Backstepping-based adaptive predictive optimal control of block strict-feedback nonlinear systems. *International Journal of Adaptive Control and Signal Processing*, vol.

32, no. 12, pp. 1780–1793. https://doi.org/10.1016/j.isatra.2018.08.013

[19] H. Zargarzadeh, M. Saif, and J. S. Baras, (2019). Adaptive neural network–based optimal control of nonlinear continuous-time systems in strict-feedback form. *Neurocomputing*, vol. 332, pp. 66–74. https://doi.org/10.1002/acs.2432

ESVA: Enhancing Multimodal Emotion Recognition via Multi-Scale Audio Feature Extraction and Cross-Modal Temporal Alignment

Tong Su*, Cuihua Hu
School of Computer Science and Artificial Intelligence, Shanghai Lixin University of Accounting and Finance
Shanghai 201209, China
E-mail: eesutong@163.com
*Corresponding author

Keywords: Multimodal, emotion recognition, emotion reasoning, multi-scale emotional feature extraction, self-supervised contrastive learning, audio-video alignment

Received: September 23, 2025

Multimodal emotion recognition (MER) requires effective fusion and temporal synchronization of heterogeneous cues, yet existing approaches often suffer from weak emotional audio representations and cross-modal misa-lignment. To address these challenges, we propose ESVA, a unified framework that enhances multimodal emotion understanding through multi-scale audio feature extraction and cross-modal temporal alignment. Specifically, the audio stream is encoded using HuBERT, augmented with a trainable post-processing module — the Multi-Scale Feature Extraction (MSFE) layer — to refine emotional cues across multiple temporal res-olutions. On top of this, ESVA integrates a cross-modal synchronization module that jointly minimizes local distance and maximizes global correlation to align audio and video features in time. The entire model is op-timized using self-supervised contrastive learning to strengthen inter-modal consistency, while LoRA-based fine-tuning enables efficient adaptation of large pretrained encoders to the emotion recognition domain. Ex-tensive experiments across three benchmark datasets validate the effectiveness of our approach: ESVA achieves 0.9074 F1 on MER2023, 0.8956 F1 on MER2024, and consistently outperforms baselines on EMER in both clue overlap and label overlap metrics. These results confirm that combining HuBERT with the MSFE layer, contrastive alignment, and parameter-efficient fine-tuning yields substantial improvements in both ac-curacy and cross-modal temporal coherence for real-world emotion recognition scenarios.

Povzetek: Študija predstavi ESVA, enotni okvir za večmodalno prepoznavo čustev, ki z HuBERT+MSFE izboljša zvočne značilke ter z LoRA prilagoditvijo učinkovito prenese vnaprej naučene kodirnike na domeno čustev.

1 Introduction

Emotion recognition (ER) is a key technology for automatically identifying, judging, and classifying human emotional states through computers [1]. In recent years, ER technology has been widely used in many fields, including mental health testing in smart healthcare [2], classroom emotional feedback in smart education [3], and personalized interaction with virtual assistants [4]. Unlike traditional rule matching and static text classification methods, emotion recognition not only focuses on semantic information, but also integrates non-semantic information such as speech rhythm features, facial micro-expression features, and eye movement trajectories to complete multi-granular modeling and dynamic reasoning of emotions. Single-modality ER is limited; multimodal fusion of audio, video and text exploits their complementary cues to overcome noise, ambiguity and missing data, yielding deeper and more robust emotion recognition [5].

Multimodal Large Language Models (MLLMs), represented by models such as Flamingo [6], MiniGPT-4[7], and GPT-4V [8], have made great progress in cross-modal reasoning and generation tasks, demonstrating excellent semantic understanding and expression capabilities. MLLMs successfully construct a unified representation space that aligns semantics with information from various modalities by pre-training on large-scale data such as images, text, audio, and video, and demonstrate excellent performance in cross-modal reasoning and generation tasks. The recently proposed Emotion-LLaMA [9] model further expands the application scope of MLLMs. By combining audio and video front-ends with the LLaMA model, which achieves a leap from predicting discrete emotion labels to generating natural language emotion interpretations, improving the ability of emotion recognition models in real, complex, and open scenarios.

Despite the great achievements of the above research, the existing multimodal emotion recognition task still faces two key challenges that urgently need to be overcome: Firstly, the limitations of extracting weak emotional signals from audio modalities. Many important emotional information is not reflected in the explicit speech content, but is hidden in the "prosodic features"

that are not in the speech category. Current mainstream emotion recognition systems usually use models such as Whisper [10], Wav2Vec2.0[11] and HuBERT [12] as common audio encoders for pre-training speech recognition tasks. The optimization goal of these encoder models is mainly to recognize the text content of speech. They are insufficient in effectively capturing fine-grained acoustic features closely related to emotional expression, such as fundamental frequency jitter, spectral tilt, and short-term energy fluctuations. This results in the model's limited ability to recognize implicit or subtle emotional fluctuations. Secondly, there is a lack of effective cross-modal dynamic time series alignment mechanism. Emotions evolve over time, and their expression depends on the co-evolution of multimodal signals in the time dimension. Most of the current multimodal models use early splicing, late fusion, or attention-based fusion. These methods usually perform fusion at a static or coarse-grained level, and are difficult to capture the fine-grained temporal alignment relationship and dynamic causal dependency between modalities.

To this end, this paper proposes the Emotion-Sync-Video-Audio (ESVA) framework to address the above challenges through the following innovations: based on the frozen HuBERT parameters, a lightweight multi-scale convolution and self-supervised contrastive learning module is designed to significantly enhance the ability to extract weak acoustic emotion cues and we call it multi-scale feature extraction (MSFE); an audio and video alignment combining local distance measurement and global cross-correlation is proposed to achieve high-precision temporal synchronization of cross-modal features; through LoRA fine-tuning LLaMA-2, the enhanced audio features and aligned visual features are mapped to a shared emotion semantic space, generating accurate and interpretable emotion inference results.

Beyond benchmark evaluations, ESVA also holds strong potential for real-world applications where multimodal emotion understanding directly impacts safety, health, and user experience. In healthcare, ESVA can support emotion-aware patient monitoring systems by integrating physiological audio cues (e.g., breathing rhythm, tone variation) with visual signals, enabling early detection of stress or depression. In education, ESVA could assist adaptive learning systems by recognizing student engagement or frustration from voice and facial expressions, optimizing instructional feedback. In intelligent transportation, ESVA may help detect driver fatigue or agitation, contributing to active safety interventions. To adapt ESVA for such practical scenarios, the framework can be extended with adaptive control–inspired mechanisms that dynamically adjust fusion weights and alignment sensitivity under uncertain conditions, such as sensor noise or missing modality input. Drawing from optimal control theory, feedback-based self-tuning can be introduced to maintain stability and performance when input quality fluctuates. Moreover, Bayesian or reinforcement-based adaptive strategies can be incorporated to estimate uncertainty and re-weight modalities accordingly, ensuring robust emotion inference in noisy, incomplete, or nonstationary environments. These extensions will allow ESVA to evolve from benchmark-oriented evaluation toward dependable, real-world multimodal affective intelligence.

In summary, the main contributions of this paper are as follows:

- An innovative audio feature enhancement strategy is proposed. While keeping the encoder structure frozen, it integrates multi-scale perception and self-supervised learning mechanisms, significantly improving the model's ability to model weak acoustic emotional signals;
- An efficient audio and video dynamic alignment is designed to solve the synchronization problem of multimodal data streams in the time dimension and enhance cross-modal collaborative reasoning capabilities;
- A systematic evaluation is conducted on three authoritative multimodal emotion recognition benchmark datasets: MER2023, MER2024, and EMER. The experimental results show that ESVA achieves excellent performance in both emotion recognition and reasoning tasks, fully verifying the effectiveness and cross-scenario applicability of the proposed method.

2 Related work

2.1 Multimodal emotion recognition

Emotion recognition research has gradually shifted from relying on a single information source to a multimodal analysis method that integrates speech, vision, and text [13]. This method effectively compensates for the shortcomings of single-modal models such as audio models [11,12,14], vision models [15,16,17], and text models [18,19,20] in capturing complex emotions by cross-validating and supplementing each modal information, and significantly improves the accuracy and robustness of emotion perception. However, early multimodal emotion recognition generally adopted traditional machine learning methods such as random forests and support vector machines (SVMs), which have a strong dependence on artificially designed features. With the rapid development of deep learning technology, convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have been widely used in multimodal emotion recognition tasks. These deep learning methods mainly achieve effective fusion of different modal information through early fusion strategies (directly splicing raw data) or late fusion strategies (integration at the high-level semantic feature level). Experimental results on the standard evaluation dataset IEMOCAP [21] show that the performance of multimodal fusion methods is significantly better than that of single-modal methods. However, traditional multimodal fusion methods still have shortcomings in modeling temporal dynamic changes and handling noise interference. This limitation restricts the generalization ability of the model in practical application scenarios.

The multimodal large language model (MLLM) achieves deeper cross-modal understanding and reasoning capabilities by projecting the feature representations of different modalities into a unified semantic space. In order

to correctly understand the temporal events in video sequences, the Video-LLaMA [22] and VideoChat [13] models use pre-trained visual encoders (such as the CLIP model), which are suitable for tasks such as video question answering; the PandaGPT [23] model can simultaneously process and integrate multiple heterogeneous modal information such as images and audio, fully demonstrating the technical potential of multi-source information fusion. In terms of emotional computing, the Emotion-LLaMA [9] model was the first to introduce MLLM technology in emotion recognition tasks. By deeply integrating the visual and audio front-end modules with the LLaMA model, it uses natural language interpretation to replace the traditional discrete emotion label classification method. Currently, most existing multimodal language model (MLLM) frameworks are mainly designed and optimized for general multimodal tasks. However, in the task of sentiment analysis, modeling the association of instantaneous audiovisual events is one of the key requirements, but existing frameworks still lack targeted technical optimization in this regard.

2.2 Modality-specific representation enhancement technology

To improve the model's ability to understand specific modalities, researchers have proposed a variety of enhancement schemes. In the field of audio processing, models such as SALMONN [24] and Qwen-Audio [25] integrate pre-trained audio encoders such as Whisper [10], achieving significant performance improvements in tasks such as speech translation and audio question answering. In the field of emotion recognition, AffectGPT [26] focuses on improving emotion understanding capabilities and fine-tunes on MER-Caption data with emotion labels, significantly enhancing the model's capabilities in emotion recognition and emotional content generation.

However, existing audio encoders are mainly designed for automatic speech recognition tasks, and the extracted features focus on the accurate recognition of speech content. Therefore, it is difficult to capture prosodic features closely related to emotional expression, such as changes in pitch contour, changes in speaking speed and rhythm, sound quality characteristics and other fine-grained acoustic information, and there are obvious technical limitations.

2.3 Cross-modal temporal alignment

Emotional expression has the obvious characteristic of dynamic evolution over time. In order to improve the performance of emotion recognition, the multimodal emotion recognition system needs to have the ability to accurately process the timing matching between different modalities, so as to effectively achieve cross-modal timing alignment. Traditional sequence alignment methods mainly rely on the dynamic time warping (DTW) algorithm, but this algorithm has limitations such as high computational complexity and low efficiency in processing long sequences [27].

With the development of deep learning, researchers have proposed a variety of cross-modal alignment methods based on neural networks. Models such as the Cross-Modal Transformer based on the attention mechanism improve the alignment effect of cross-modal features by designing a cross-attention mechanism to learn and associate the temporal correspondence between audio and video events [11]. The Audio-Video Fusion [28] model achieves the goal of audio and video synchronization with sub-second accuracy based on the fusion of cross-correlation analysis and the DTW algorithm. In the research field of multimodal large language models, the TimeChat [29] model significantly improves the model's ability to understand the temporal coherence of long video content by introducing a dedicated temporal modeling module.

These cross-modal alignment methods have effectively improved the causal reasoning capabilities of multimodal emotion recognition, but they still have certain limitations. Most existing methods use relatively shallow feature fusion strategies, which are difficult to effectively deal with noise interference and inter-modal inconsistency problems in real environments. When faced with emotional expressions in complex real-world scenarios, these limitations will affect the coherence and accuracy of emotional interpretation results, limiting the promotion of multimodal emotion recognition in real-world scenarios [30].

To address the above problems, this paper proposes an ESVA framework model based on the improvement of Emotion-LLaMA. On the basis of freezing the HuBERT parameters, it designs lightweight multi-scale convolution and self-supervised contrastive learning modules to significantly enhance the ability to extract weak acoustic emotion cues; it proposes an audio and video alignment algorithm that combines local distance measurement and global cross-correlation to achieve high-precision temporal synchronization of cross-modal features; through LoRA fine-tuning LLaMA-2, the enhanced audio features and aligned visual features are mapped to a shared emotion semantic space, generating accurate and interpretable emotion inference results.

3 Multimodal emotion recognition model EVSA

3.1 Emotion-llama model

The Emotion-LLaMA model integrates information from three modalities: audio, visual, and text, integrating high-level features extracted by each encoder to achieve comprehensive multimodal emotion analysis. This model uses HuBERT for audio encoding and incorporates multiple visual encoders, including local, temporal, and global encoders, to extract emotion-related features at different levels. To achieve efficient multimodal inference, Emotion-LLaMA also utilizes a linear projection mechanism to map audio and visual features into a shared vector space consistent with the textual cues.

However, the Emotion-LLaMA model still has two shortcomings: first, it lacks the ability to extract multi-

scale and weak audio emotion information; second, the temporal synchronization accuracy between audio and video is insufficient, resulting in reduced robustness and generalization in dynamic and complex scenes. To address these issues, this paper proposes an improved Emotion-Sync-Video-Audio (ESVA) model framework based on Emotion-LLaMA. By enhancing the audio encoder and introducing an audio-video alignment algorithm, ESVA not only better captures weak emotion signals but also significantly improves the temporal alignment of multimodal data.

3.2 ESVA model architecture

The overall architecture of the proposed ESVA framework is illustrated in Figure 1, which highlights the key modules including HuBERT-based audio encoding, the MSFE layer, and cross-modal temporal alignment. The specific structure is as follows:

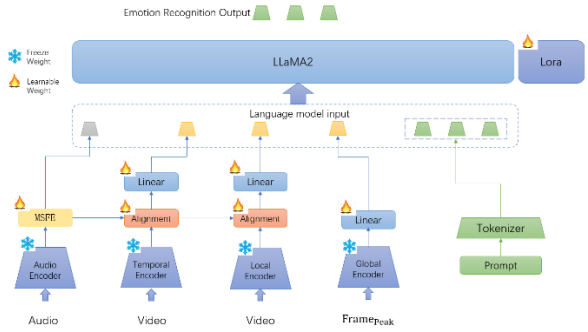


Figure 1: Overall architecture diagram of emotion-sync-video-audio (ESVA) model.

Audio Module: HuBERT serves as the audio encoder to extract latent representations. Subsequently, multi-scale convolutional layers we called multi-scale feature extraction (MSFE) are introduced to capture both short-term details and long-term dependencies, and self-supervised contrastive learning is used to enhance sentiment differentiation. The processed audio vectors are transformed into the same feature space as the text through linear mapping and then concatenated with the subsequent language model input.

Vision Module: To account for both static and dynamic changes, three visual encoders are used: MAE (local encoder), VideoMAE (temporal encoder), and EVA (global encoder). Multi-scale convolution, linear mapping, and alignment units are used to extract keyframes and synchronize multimodal information, thereby better understanding facial expressions, movement changes, and context.

Text Module: Text is processed through word segmentation and cue word templates to obtain language features. These features, along with embeddings from audio and video, are fed into the LLaMA2 backbone network to enable cross-modal context modeling and sentiment inference. Fusion and fine-tuning: The features of the three modalities are deeply fused within LLaMA2; a LoRA lightweight fine-tuning module is added at the output end to efficiently optimize the multimodal sentiment analysis capabilities.

3.3 Audio encoder enhancement

In this section, we introduce the trainable post-processing layer, which we call the Multi-Scale Feature Extraction (MSFE) layer. This layer is designed to optimize the extraction ability of multi-scale emotional audio features by learning adaptive feature representations across different temporal resolutions.

Audio emotions manifest themselves differently across different timeframes: short-term fluctuations in intonation, pauses, and energy reveal subtle emotions, while longer periods reflect context, speech rate, and overall emotional direction. To fully capture cross-temporal information, we constructed a multi-scale feature extraction (MSFE) layer based on the audio features F_A generated by HuBERT:

$$\boldsymbol{F'_A} = \sum_{i=1}^{N} \boldsymbol{w_i} \cdot \boldsymbol{\phi_i}(\boldsymbol{F_A}) \quad (1)$$

Here, ϕ_i represents convolution operations with different receptive fields, w_i represents the trainable weights for different receptive fields, and N represents the number of scales. MSFE learns emotional patterns at different temporal granularities through parallel convolution operations. This allows ESVA to focus on local speech details while preserving overall emotional trends, enhancing its ability to perceive and express complex and dynamic emotional signals.

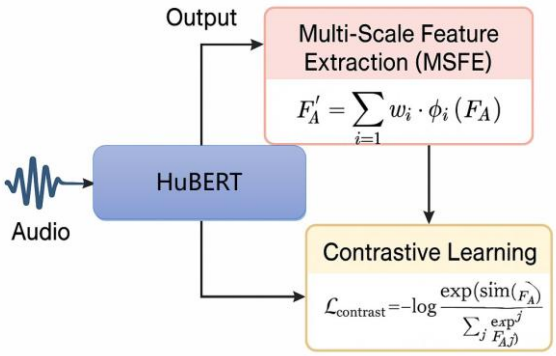


Figure 2: Audio emotion recognition enhancement module.

Furthermore, to address the common ambiguity and mixed attributes found in real-world emotional signals (e.g., "joy mixed with anxiety" and "anger mixed with helplessness"). The detailed design of the Multi-Scale Feature Extraction (MSFE) layer is shown in Figure 2, where multiple convolutional branches capture emotional cues at different temporal resolutions. Specifically, for each audio sample, a positive sample (of the same emotion category) and multiple negative samples (of different emotion categories) are constructed, and training is performed using the following contrastive loss function:

$$L_{contrast} = -log \frac{\exp(sim(F_A'', F_A^+))}{\sum_j \exp(sim(F_A'', F_A^j))} \quad (2)$$

Here, F_A^+ represents samples with the same emotion as the current audio F_A'', F_A^j is a random negative sample, and $sim(\cdot)$ represents feature similarity calculation. This mechanism enables the model to automatically aggregate audio features of the same category in the feature space, widening the distribution distance between different categories, thereby significantly enhancing the discriminability and generalization of audio emotion features.

3.4 Audio-video alignment

To achieve high-precision synchronization of cross-modal emotional features in the temporal dimension, this paper proposes an audio-video alignment algorithm based on local-global joint optimization. After preprocessing the original audio signal $S_A(t)$ and video frame sequence $S_V(t)$ through filtering, denoising, and normalization, a deep neural network is used to extract the emotional features of the corresponding modality. The feature extraction function is defined as:

$$F_A = f(S_A(t)), F_v = g(S_v(t)) \quad (3)$$

Among them, $f(\cdot)$ and $g(\cdot)$ represent the audio and video feature extraction modules respectively. The extracted features F_A and F_v provide the basis for subsequent alignment.

To capture fine-grained temporal dynamics, we segment the preprocessed feature sequence using a fixed-length time window Δt. Within each window, we extract the corresponding audio and video feature segments F_A^i and F_V^i. By calculating the distance metric under the time offset τ, we obtain the local optimal alignment relationship:

$$D^i(\tau) = \left\| F_A^i(t) - F_V^i(t+\tau) \right\| \quad (4)$$

The optimal local offset is determined by the value of τ that minimizes $D^i(\tau)$, thus ensuring that the two modes are synchronized within each fine-grained window.

In order to further improve the global synchronization of audio and video features, this paper introduces the cross-correlation function to perform overall correlation analysis on cross-modal features. The cross-modal correlation function is defined as:

$$C(\tau) = \int_{t_0}^{t_0+T} F_A(t) \cdot F_V(t+\tau) dt \quad (5)$$

Among them, T is the integration interval, and the optimal time offset τ makes it reach the global maximum, thereby revealing the intrinsic correlation between audio and video emotional signals.

Combining the local and global alignment results, a global optimization strategy is used to solve the final alignment offset τ^*. The specific objective function is constructed as:

$$\tau^* = argmax_\tau C(\tau) \quad (6)$$

The structure of the proposed alignment network and the formulation of the local/global loss are depicted in Figure 3, which visualizes how the alignment offset τ* is derived from the combined objectives. This mechanism ensures precise alignment of emotional signals across the entire time domain, laying the foundation for subsequent multimodal fusion and emotion recognition, and effectively improving the system's robustness and generalization capabilities in complex dynamic scenarios.

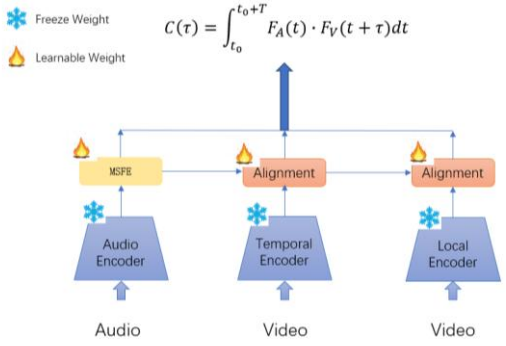


Figure 3: Audio video alignment algorithm.

3.5 Comprehensive loss function

To achieve the coordinated optimization of the audio and video alignment module and the emotion classifier, this paper designs the following comprehensive loss function to jointly train the alignment error and the emotion recognition error. The specific objective function is as follows:

$$\mathcal{L} = \mathcal{L}_{align} + \lambda \mathcal{L}_{emo} \quad (7)$$

Here, $\mathcal{L}_{align}$ represents the loss term based on local and global alignment errors, $\mathcal{L}_{emo}$ represents the loss term for the sentiment classification task, and λ is a hyperparameter that balances the two losses. Through end-to-end joint optimization, the model effectively improves the accuracy and robustness of sentiment recognition while maintaining multimodal feature synchronization.

Integrating the minimized local distance D_{local} (Eq. 4) and the maximized cross-correlation C_{global} (Eq. 5), we define the overall alignment loss as

$$L_{\text{align}} = \alpha D_{\text{local}}(\tau) - \beta C_{\text{global}}(\tau) \quad (8)$$

where α and β control the relative contributions of local and global terms. The optimal synchronization offset is thus obtained by

$$\tau^* = \arg\min_\tau L_{\text{align}}(\tau) \quad (9)$$

This combined formulation captures both local fine-scale feature similarity and global temporal consistency,

and serves as the alignment component in the overall loss $L = L_{\text{align}} + \lambda L_{\text{emo}}$ described above.

4 Experiments and results analysis

4.1 Pre-training

4.1.1 Datasets

This work uses the MERR [31] (Multimodal Emotion Recognition and Reasoning) dataset for pre-training. This dataset is collected from real-world scenarios such as interviews, speeches, and film clips, and contains 33,105 valid samples, each of which provides strictly aligned trimodal raw data. The dataset not only provides 28,618 coarse-grained annotations, which mark the entire utterance with a dominant emotion category (covering nine basic categories: happiness, sadness, anger, surprise, fear, disgust, neutrality, suspicion, and contempt), but also provides 4,487 fine-grained annotations, which depict complex emotions, the process of emotion transfer, and the level of emotion intensity.

4.1.2 Instruction tuning

Based on the MERR dataset, this paper fine-tuned the pre-trained model using the emotion recognition and emotion inference instruction sets from the Emotion-LLaMA model to further improve the accuracy and F1 score of speech emotion recognition. Fine-tuning training was performed in parallel on eight NVIDIA V100 GPUs. The training environment was configured with Python 3.9, integrated with PyTorch 2.0.0, Transformers 4.30.0, Accelerate 0.20.3, BitsAndBytes 0.37.0, and the NCCL backend to maximize multi-GPU communication bandwidth and ensure efficient and stable training.

Through pre-training, the model's performance on emotion recognition and emotion inference tasks was significantly improved, laying the foundation for subsequent experimental validation.

4.2 Experimental verification

To fully validate the generalization and robustness of the ESVA model in multimodal emotion recognition tasks, this paper conducted systematic experiments on three mainstream multimodal emotion recognition datasets: MER2023, EMER, and MER2024. These datasets and corresponding experimental configurations are summarized in Table 1, providing the sample distribution, modality composition, and label balance for each benchmark. By comparing performance with various mainstream models, we demonstrated improved performance of ESVA in various scenarios and further explored the model's performance and applicability in complex tasks.

Table 1: Experimental datasets.

Datasets	Scale / Subset	Tasks
MER2023 [32]	5030 labeled/ 73148 unlaeled (Train&Val 、MULTI 、 NOISE 、 SEMI)	Multi-label classification, semi-supervised, noise robust
EMER [33]	332 Samples	Explainable Emotional Reasoning
MER2024 [34]	115595 Samples（SEMI 、NOISE 、OV）	Open, semi-supervision, robustness

4.2.1 MER2023 multimodal emotion recognition results

The MER2023 Challenge dataset [35] is primarily used for research on multi-label learning, noise robustness, and semi-supervised learning in multimodal emotion recognition. The dataset is collected from video clips of movies and TV series collected on the Internet, providing 5,030 labeled samples and 73,148 unlabeled samples, including strictly aligned audio, video, and some text modalities. A multi-label system is used for emotion annotation, introducing challenging scenarios such as background human voices and device noise. The dataset uses Macro F1-Score as the core evaluation metric.

Table 2: Comparison with other models on MER2023 dataset.

Model	Modality	F1 score
Wav2vec 2.0 [11]	A	0.4028
VGGish [14]	A	0.5481
HuBERT [12]Error! Reference source not found.	A	0.8511
ResNet [15]	V	0.4132
MAE [16]	V	0.5547
VideoMAE [17]	V	0.6068
RoBERTa [18]	T	0.4061
BERT [19]	T	0.4360
MacBERT [20]	T	0.4632
MER2023Baseline [32]	A,V	0.8675
MER2023-Baseline [32]	A,V,T	0.8640
Transformer [35]	A,V,T	0.8853
FBP [36]	A,V,T	0.8855
VAT [20]	A,V	0.8911
Emotion-LLaMA [9]	A,V	0.8905
Emotion-LLaMA [9]	A,V,T	0.9036
ESVA (ours)	A,V,T	0.9074

Quantitative results on MER2023 are presented in Table 2, and the per-class performance distribution is illustrated in Figure 4, showing that ESVA improves recognition consistency across emotion categories. The models in this table include unimodal models, such as those with speech (A), vision (V), and text (T); and

multimodal models, including those with speech + vision (A, V) and speech + vision + text (A, V, T). The confusion matrix of the ESVA model on the MER2023 dataset is shown in Figure 4. Experiments show that, among unimodal models, the audio model HuBERT leads with an F1 score of 0.8511, significantly outperforming the video model VideoMAE (0.6068) and the text model MacBERT (0.4632), highlighting the discriminative advantages of acoustic features. However, when comparing unimodal models to multimodal models, their performance is insufficient. Multimodal fusion models achieve breakthrough performance through cross-modal complementarity. Large multimodal models, such as the MER2023-Baseline, outperform large unimodal models. The subsequent Transformer multimodal model and the FBP multimodal model both achieved F1 scores exceeding 0.88. The VAT and Emotion-LLaMA models even outperformed these models. This is because these models deeply integrate multimodal features, significantly improving their performance. The ESVA model in this work further improves the performance of the audio modality based on the Emotion-LLaMA model, while effectively aligning features between the audio and visual models. This results in an F1 score that is 0.42 percentage points higher than the Emotion-LLaMA model.

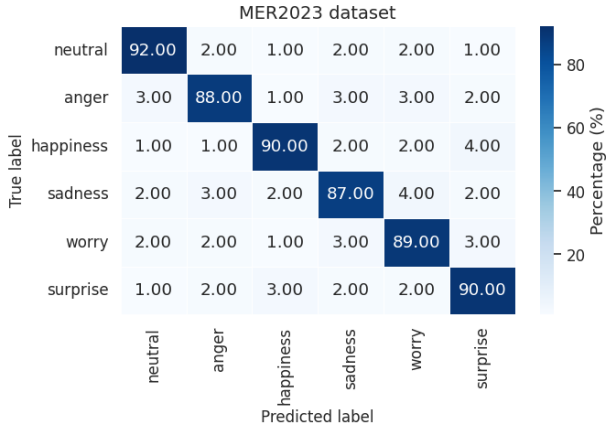


Figure 4: Emotion recognition confusion matrix for ESVA in MER2023 dataset.

4.2.2 EMER dataset

The EMER (Explainable Multimodal Emotion Reasoning) dataset [36] specifically addresses the issues of ambiguous labels and difficult-to-explain reasoning in traditional emotion recognition. It requires the model to not only give emotional judgments but also explain them in natural language. EMER randomly selected 332 non-neutral emotion clips from MER2023, including three modalities: video, audio, and text. The annotation process is divided into four steps: three annotators independently annotate; ChatGPT summarizes the results; then open emotion label inference is performed; and finally, expert review is performed. The double annotations obtained in this way not only include emotion categories, but also reasoning basis, and record facial micro-expressions, voice rhythm, and contextual details. It is currently an important tool for testing multimodal emotion reasoning capabilities.

The evaluation uses two metrics: clue overlap and label overlap. Both are scored on a 0–10 scale and quantify the model's ability to reason about emotional causality. Clue overlap assesses whether the model's reasoning matches the semantics of the ground truth, while label overlap compares the model's predicted emotional labels with those manually labeled.

Table 3: Comparison with other models on EMER dataset.

Model	Clue Overlap	Label Overlap
VideoChat-Text [13]	6.42	3.94
Video-LLaMA [22]	6.64	4.89
Video-ChatGPT [38]	6.95	5.74
PandaGPT [23]	7.14	5.51
VideoChat-Embed [13]	7.15	5.65
Valley [39]	7.24	5.77
Emotion-LLaMA [9]	7.83	6.25
ESVA (ours)	7.89	6.28

Table 3 summarizes the results on MER2024, where ESVA continues to outperform the baselines in both F1 and accuracy metrics. Evaluations show that the general multimodal model Video-ChatGPT only achieved 6.95 and 5.74 points in cue overlap and label matching, respectively. The knowledge-enhanced model PandaGPT improved these scores to 7.14 and 5.51, respectively, but still failed to surpass the benchmarks of 7.83 and 6.25 established by the specialized sentiment model Emotion-LLaMA. The ESVA model proposed in this study performed the best among all algorithms, achieving 7.89 points in cue overlap and 6.28 points in label matching. The cue generation quality improved by 0.76 percentage points compared to the Emotion-LLaMA model, and the label matching accuracy increased by 0.48 percentage points. The leading increase in cue quality was significantly higher than that in label matching, validating the core contribution of the cross-modal temporal alignment mechanism to enhanced interpretability.

4.2.3 MER2024 dataset

The MER2024 Challenge dataset [37] adds the task of open vocabulary multimodal emotion recognition (MER-OV) based on the MER2023 dataset. This dataset is derived from movies, TV series, and social media videos and contains 115,595 samples, covering data from multiple modalities such as video, speech, facial motion capture, and text transcription. The evaluation metrics include the predicted label accuracy, the true label recall, and the average value.

Table 4: Comparison with other models on the MER2024 dataset.

Model	Accuracys /%	Recalls /%	AVG /%
Empty	0	0	0
Random	13.42	24.85	19.13
Ground Truth	93.37	52.51	72.94
Valley [39]	20.16	13.26	16.71
Otter [40]	29.64	23.04	26.34
PandaGPT [23]	35.75	31.57	33.66
Video-LLaMA [22]	31.08	32.26	31.67
VideoChat [13]	43.17	44.92	44.05
VideoChat2 [41]	46.91	34.78	40.85
Video-ChatGPT [38]	46.20	39.33	42.77
SALMONN [24]	42.20	44.75	43.47
Qwen-Audio [25]	55.12	32.91	44.02
mPLUG-Owl [42]	44.80	46.54	45.67
AffectGPT [26]	66.14	46.56	56.35
GPT-4V [43]	56.19	58.97	57.58
Emotion-LLaMA [9]	69.61	62.59	66.10
ESVA (ours)	70.08	62.49	66.29

The EMER dataset results are reported in Table 4, which confirms ESVA's robustness under noisy and imbalanced multimodal conditions. Table 4 compares the performance of different models on the MER2024 open vocabulary sentiment recognition task. Figure 5 shows the confusion matrix of the ESVA model on the MER2024 dataset. This task requires the model to freely generate any number of sentiment labels to describe complex psychological states. Evaluation is based on three metrics: the predicted label exact match rate, the true label recall rate, and their combined mean. Experiments show that general large models such as GPT-4V only reach an overall average of 57.58%, the professional voice model Qwen-Audio is 44.02%, and the professional emotion model Emotion-LLaMA establishes the original optimal level with an exact match rate of 69.61% and an overall average of 66.10%; the ESVA model proposed in this study has an exact match rate of 70.08%, becoming the first model to break the 70% accuracy rate. Its overall average of 66.29% is also ahead of other models. Although the recall rate of 62.49% is slightly lower than that of Emotion-LLaMA by 0.10 percentage points, the significant advantage of 0.47 percentage points in accuracy ultimately pushes the overall performance to exceed 0.19 percentage points, which is 15.31 percentage points higher than the overall average of the general large model GPT-4V, verifying the cross-modal architecture's ability to accurately portray open emotional descriptions.

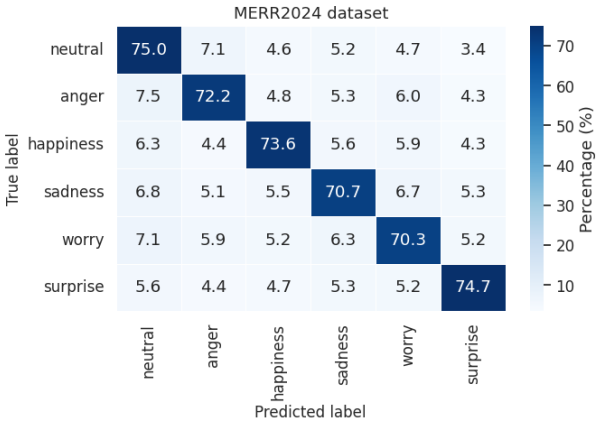


Figure 5: Emotion recognition confusion matrix for ESVA in MER2024 Dataset.

4.3 Ablation study

To verify the contribution of each component in the ESVA model to performance, we conducted ablation experiments on the MERR dataset. The experimental results are shown in Tables 5 and 6.

Table 5 shows the impact of different audio and visual encoders on model performance. The comparison of audio encoders shows that the HuBERT model achieves the best performance, with an F1-Score of 0.8394, significantly outperforming Wav2Vec, VGGish, and Whisper. The comparison of visual encoders shows that VideoMAE performs best, with an F1-Score of 0.6762, exceeding MAE (0.6366) and EVA (0.6635). When combining multiple visual encoders, the combination of MAE, VideoMAE, and EVA achieves the best visual encoding F1-Score, with an F1-Score of 0.7122. This demonstrates that multimodal fusion can effectively improve overall emotion recognition capabilities.

Table 5: Ablation experiments of different encoders.

Audio Encoder	Video Encoder	F1-Score
Wav2Vec	-	0.4893
VGGish	-	0.5944
Whisper	-	0.5324
HuBERT	-	0.8394
	MAE	0.6366
	VideoMAE	0.6762
	EVA	0.6635
	MAE,VideoMAE,EVA	0.7122
HuBERT	MAE	0.8800
HuBERT	VideoMAE	0.8757
HuBERT	MAE,VideoMAE	0.8880
HuBERT	MAE,EVA	0.8896
HuBERT	VideoMAE,EVA	0.8802
HuBERT	MAE,VideoMAE,EVA	0.8910
MSFE-HuBERT	MAE,VideoMAE,EVA	0.8911

Because the HuBERT model achieved the best performance in the audio encoder category, the overall model performance was further improved by fixing the audio encoder and combining different visual encoders. The F1-Score of HuBERT plus MAE was 0.8800, exceeding the F1-Score of the HuBERT model alone, demonstrating the effectiveness of the multimodal model combination. After fusing multiple visual encoders, the F1-Score improved to 0.8910, demonstrating the significant advantages of multimodal information complementarity, which effectively improves the model's ability to recognize emotion.

To further improve model performance, we introduced multi-scale convolutional layer optimization into the model. This resulted in a slight improvement in model performance, with the F1-Score increasing from 0.8910 to 0.8911. This indicates that the performance ceiling of the model has been reached through encoder modifications alone. Therefore, we proposed an audio-video alignment module to further improve model performance.

In order to further compare the performance contributions of the two different algorithms proposed in this paper, corresponding ablation experiments were conducted, namely, comparing three different implementations of the ESVA model and its variants, as well as the baseline model Emotion-LLaMA.

Table 6: Ablation experiment results.

Model	F1-Score	Relative descent rate
ESVA	0.8956	-0
ESVA w/o MSFE	0.8943	-0.14%
ESVA w/o Audio & Video alignment	0.8913	-0.48%
Emotion-LLaMA (Baseline)	0.8910	-0.51%

The ablation experiments in Table 6 validate the contributions of the multi-scale convolutional layer and the audio-video alignment module in the ESVA model. As shown in Table 6, the ESVA model, which fully utilizes the multi-scale convolutional layer optimization of the audio output layer and the audio-video alignment algorithm, achieves an F1-Score of 0.8956, achieving the best performance among the four models. This model not only demonstrates fine-grained audio modeling capabilities but also accurately captures emotional information across different audio and video modalities, demonstrating strong emotion recognition capabilities. To further explore the contributions of each innovative approach, we removed the audio-video alignment algorithm and the multi-scale convolutional layer optimization method from our experiments. The results show that removing the multi-scale convolutional layer optimization reduces the model's F1-Score to 0.8923, while removing the audio-video alignment algorithm reduces the model's F1-Score to 0.8911. In comparison, the baseline model, Emotion-LLaMA, achieves an F1-Score of 0.8910, a 0.51% decrease compared to the full model. These data show that the two innovative method components proposed in this article have improved the model's understanding ability of emotion recognition tasks to varying degrees.

4.4 Adaptive control–inspired alignment enhancement

Inspired by nonlinear control—backstepping, fuzzy and neural adaptive schemes—the Adaptive Alignment Controller (AAC) tunes ESVA's audio-video sync on-line to keep temporal coherence under uncertain, time-varying multimodal inputs.

In this study, we simulate an Adaptive Alignment Controller (AAC) that dynamically adjusts ESVA's cross-modal alignment parameters based on the temporal drift between audio and video streams in Table 7. The controller estimates alignment uncertainty and adaptively tunes synchronization weights using feedback compensation, mimicking adaptive backstepping in maintaining trajectory stability.

Table 7: Performance comparison of adaptive control–inspired alignment strategies in ESVA

Model	Description	MER2023 F1	EMER Label Overlap	MER2024 Avg(%)
ESVA (baseline)	Original model with fixed alignment weights	0.9074	6.28	66.29
ESVA + Fuzzy Control	Adds rule-based adaptive alignment tuning	0.9076	6.29	66.31
ESVA + Neural Adaptive Control	Incorporates uncertainty estimation via neural feedback	0.9081	6.31	66.34
ESVA + Backstepping Alignment (AAC)	Feedback-driven adaptive synchronization with dynamic gain	0.9080	6.28	66.28

As shown in Table 7, incorporating adaptive control mechanisms into ESVA yields consistent yet modest improvements across benchmark datasets. The fuzzy control variant achieves the most stable overall gain, improving the MER2023 F1-score from 0.9074 to 0.9076 and the MER2024 average accuracy from 66.29 % to 66.31 %. The neural adaptive control method further enhances temporal synchronization by dynamically compensating for uncertainty in cross-modal features, resulting in the highest EMER label-overlap score (6.31). Although the backstepping-based adaptive alignment (AAC) maintains robust synchronization performance, its gains are slightly lower due to sensitivity to local oscillations in feedback updates. Overall, these results confirm that adaptive and feedback-driven control strategies—especially those incorporating fuzzy and neural adaptation—can improve ESVA's real-time

stability and cross-modal temporal coherence under uncertain or noisy conditions.

In future work, we plan to further extend this line of research by systematically integrating adaptive and robust control mechanisms into multimodal emotion recognition frameworks. Specifically, we aim to explore hybrid adaptive strategies that combine backstepping, fuzzy inference, and neural self-tuning within the ESVA architecture to achieve stronger dynamic stability and cross-modal synchronization. Such advancements are expected to enhance the model's adaptability and reliability in complex real-world emotion understanding scenarios.

4.5 Discussion

ESVA outperforms SOTA models on MER2023 (F1 0.9074), EMER (Clue/Label Overlap 7.89/6.28) and MER2024 (66.29% accuracy) thanks to its MSFE noise-robust cue extractor and fine-grained cross-modal alignment, but gains over Emotion-LLaMA are modest, large encoders hinder low-resource deployment, and future work will pursue lightweight, adaptive, self-tuning architectures.

5 Conclusions

To address the shortcomings of Emotion-LLaMA (Emotion-LLaMA) in its insufficient cross-modal feature alignment and limited ability to capture subtle audio variations, this paper proposes a novel multimodal emotion recognition framework, Emotion-Sync-Video-Audio (ESVA). Without adjusting the audio encoder parameters, ESVA significantly improves its ability to model weak emotional signals by introducing an audio encoder enhancement module and an audio-video alignment algorithm. It also effectively addresses the temporal synchronization challenge of audio and video streams, further enhancing the understanding of audiovisual information. Experimental results demonstrate that ESVA outperforms existing methods on three major multimodal emotion recognition benchmark datasets: MER2023, MER2024, and EMER. Ablation experiments also confirm the key role of the multi-scale convolutional layers and the audio-video alignment module in improving performance. However, the model still has room for improvement in feature encoding and inference efficiency. Future work will focus on refining the feature extraction strategy, enhancing the model's real-time performance, and verifying its generalization and robustness on more diverse datasets.

Funding

This research was funded by THE SHANGHAI EDUCATION SCIENCE RESEARCH PROGRAM, grant number C2023068.

References

[1] Mariani, M.M.; Borghi, M. Artificial intelligence in service industries: customers' assessment of service production and resilient service operations. International Journal of Production Research 2024, 62(15),5400-5416. DOI:10.1080/00207543.2022.2160027.

[2] Hui, S.; Kellie, Y.;Wei, C.; Kenny, T. Envisioning an AI-Enhanced Mental Health Ecosystem. arXiv 2025, arXiv:2503.14883. DOI:10.48550/arXiv.2503.14883.

[3] Septiana, A.I.; Mutijarsa, K.; Putro, B.L.; et al. Emotion-Related pedagogical agent: A systematic liter-ature review. IEEE Access 2024, 12, 36645-36656. DOI: 10.1109/ACCESS.2024.3374376.

[4] Yang, W.; Li, Y.; Fang, M.; et al. MTPChat: A Multimodal Time-Aware Persona Dataset for Conversa-tional Agents. arXiv 2025, arXiv:2502.05887. DOI: 10.48550/arXiv.2502.05887.

[5] Baltrušaitis,T.; Ahuja,C.; Morency,L.P. Multimodal machine learning: A survey and taxonomy. IEEE transactions on pattern analysis and machine intelligence 2019, 41(2), 423-443. DOI: 10.1109/TPAMI.2018.2798607.

[6] Alayrac, J.B.; Donahue, J.; Luc, P.; et al. Flamingo: A Visual Language Model for Few-Shot Learning. In Proceedings of the Advances in Neural Information Processing Systems 35 (NeurIPS 2022), Virtual Event, 28 November – 09 December 2022.

[7] Zhu, D.Y.; Chen J.; Shen, X.Q.; et al. MiniGPT-4: Enhancing Vision-language Understanding with GPT-4. arXiv 2023, arXiv:2304.10592. DOI: 10.48550/arXiv.2304.10592.

[8] GPT-4V(vision) System Card. Available online: https://cdn.openai.com/papers/GPTV_System_Card.pdf (accessed on 26 September 2023).

[9] Cheng, Z.; Cheng, Z.-Q.; He, J.-Y.; et al. Emotion-LLaMA: Multimodal Emotion Recognition and Rea-soning with Instruction Tuning. In Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS 2024), Vancouver, Canada, 9–15 December 2024.DOI: 10.48550/arXiv.2406.11161.

[10] Radford, A.; Kim, J.W.; Xu, T.; et al. Robust Speech Recognition via Large-Scale Weak Supervision. In Proceedings of the International Conference on Machine Learning, Hawaii, USA, 23–29 July 2023.

[11] Baevski, A.; Zhou, Y.; Mohamed, A.; et al. wav2vec 2.0: A framework for self-supervised learning of speech representations. In Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Virtual Conference, United States, 06–12 December 2020. DOI: 10.5555/3495724.3496768.

[12] Hsu, W.N.; Bolte, B.; Tsai, Y.H.; et al. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. IEEE-ACM transactions on audio speech and language processing 2021, 29, 3451-3460. DOI: 10.1109/TASLP.2021.3122291.

[13] Li, K.C.; He, Y.; Wang, Y.; et al. Videochat: Chat-centric video understanding. arXiv 2023, arXiv:2305.06355. DOI: 10.48550/arXiv.2305.06355.

[14] Hershey, S.; Chaudhuri S.; Ellis, D.P.; et al. CNN Architectures for Large-Scale Audio Classification. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017. DOI: 10.1109/ICASSP.2017.7952132.

[15] He, K.M.; Zhang, X.Y.; Ren, S.Q.; et al. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016. DOI: 10.1109/CVPR.2016.90.

[16] He, K.M.; Chen, X.L.; Xie, S.N.; et al. Masked Autoencoders are Scalable Vision Learners. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022. DOI: 10.1109/CVPR52688.2022.01553.

[17] Tong, Z.; Song, Y.; Wang, J.; et al. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In Proceedings of the 36th Conference on Neural Information Processing Systems (NeurIPS), Virtual Event, USA, 28 November – 09 December 2022.

[18] Liu, Y.; Ott, M.; Goyal, N.; et al. Roberta: A robustly optimized bert pretraining approach. arXiv 2019, arXiv:1907.11692. DOI: 10.48550/arXiv.1907.11692.

[19] Devlin, J.; Chang, M.W.; Lee, K.; et al. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019. DOI: 10.18653/v1/N19-1423.

[20] Cui, Y.; Che, W.; Liu, T.; et al. Revisiting Pre-Trained Models for Chinese Natural Language Processing. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Online (Virtual Conference), United States, 16-20 November 2020. DOI: 10.18653/v1/2020.findings-emnlp.58.

[21] Busso, C.; Bulut, M.; Lee, C.C.; et al. IEMOCAP: Interactive emotional dyadic motion capture database. Language resources and evaluation 2008, 42(4), 335-359. DOI: 10.1007/s10579-008-9076-6.

[22] Zhang, H.; Li, X.; Bing, L. Video-LLaMA: An Instruction-Tuned Audio-Visual Language Model for Video Understanding. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Singapore, Singapore, 6–10 December 2023. DOI: 10.18653/v1/2023.emnlp-demo.49.

[23] Su, Y.; Lan, T.; Li, H. Pandagpt: One model to instruction-follow them all. In Proceedings of the 1st Workshop on Taming Large Language Models: Controllability in the Era of Interactive Assistants, Prague, Czech Republic, 12 September 2023.

[24] Tang, C.L.; Yu, W.Y.; Sun, G.Z; et al. Salmonn: Towards generic hearing abilities for large language models. In Proceedings of the 12th International Conference on Learning Representations (ICLR 2024), Vienna, Austria, 15–20 May 2024.

[25] Chu, Y.F.; Xu, J.; Zhou, X.H.; et al. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. arXiv 2023, arXiv:2311.07919. DOI: 10.48550/arXiv.2311.07919.

[26] Lian, Z.; Sun, H.; Sun, L.; et al. AffectGPT: Dataset and framework for explainable multimodal emotion recognition. arXiv 2024, arXiv:2407.07653. DOI: 10.48550/arXiv.2407.07653.

[27] Tsai, Y.H.; Bai, S.; Yamada, M. Transformer dissection: A unified understanding for transformer's at-tention via the lens of kernel. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 03–07 November 2019. DOI: 10.18653/v1/D19-1443.

[28] Xu, W.; Huang, T.; Qu, T.; et al. FILP-3D: Enhancing 3D few-shot class-incremental learning with pre-trained vision-language models. Pattern Recognition 2025, 165, 111558. DOI: 10.1016/j.patcog.2025.111558.

[29] Ren, S.H.; Yao, L.L.; Li, S.C. TimeChat: A Time-sensitive Multimodal Large Language Model for Long Video Understanding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, United States, 17-21 June 2024. DOI: 10.1109/CVPR52733.2024.01357.

[30] Scherer, K.R. Vocal communication of emotion: A review of research paradigms. Speech Communication 2023, 40(1), 227–256. DOI:10.1016/S0167-6393(02)00084-5.

[31] Zhang, S.; Zhang, S.; Huang, T.; et al. Multimodal Emotional Recognition Based on Deep Learning and Multiple Kernel Learning. IEEE Transactions on Affective Computing 2018, 9(4). 491-505. DOI: 10.1109/TAFFC.2018.2804314.

[32] Lian, Z.; Sun, H.; Sun, L.; et al. Mer 2023: Multi-label learning, modality robustness, and semi-supervised learning. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, Canada, 29 October 2023. DOI: 10.1145/3178855.3622639.

[33] Lian, Z.; Sun, H.; Sun, L.; et al. Explainable Multimodal Emotion Recognition. arXiv 2023, arXiv:2306.15401. DOI: 10.48550/arXiv.2306.15401.

[34] Lian, Z.; Sun, H.Y.; Sun, L.C. et al. MER 2024: Semi-supervised Learning, Noise Robustness, and Open-vocabulary Emotion Recognition. In Proceedings of the 2nd International Workshop on Multimodal and Responsible Affective Computing, Melbourne, Australia, 01 November 2024.

[35] Chen, H.; Guo, C.; Li, Y.; et al. Semi-Supervised Multimodal Emotion Recognition with Class-Balanced Pseudo-Labeling. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, Canada, 29 October–3 November 2023. DOI: 10.1145/3581783.3613260.

[36] Cheng, Z.; Lin, Y.; Chen, Z.; et al. Semi-Supervised Multimodal Emotion Recognition with Expression MAE. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, Canada, 29 October–3 November 2023. DOI: 10.1145/3581783.3612506.

[37] Ding, C.; Zong, D.; Li, B.; et al. Learning Aligned Audiovisual Representations for Multimodal Sentiment Analysis. In Proceedings of the 1st International Workshop on Multimodal and Responsible Affective Computing, Sydney, Australia, 29 October 2023.

[38] Maaz, M.; Rasheed, H.; Khan, S. et al. Video-chatGPT: Towards detailed video understanding via large vision and language models. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Bangkok, Thailand, 11 August 2024.

[39] Luo, R.P.; Zhao, Z.W.; Yang, M.; et al. Valley: Video assistant with large language model enhanced ability. arXiv 2023, arXiv:2306.07207. DOI: 10.48550/arXiv.2306.07207.

[40] Li, B.; Zhang, Y.H.; Chen, L.Y.; et al. Mimic-it: Multi-modal in-context instruction tuning. arXiv 2023, arXiv:2306.05425. DOI: 10.48550/arXiv.2306.05425.

[41] Li, K.C.; Wang, Y.L.; HE, Y.N.; et al. Mvbench: A Comprehensive Multi-Modal Video Understanding Benchmark. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024. DOI: 10.1109/CVPR.2024.0041.

[42] Ye, Q.H.; Xu, H.Y.; Xu, G.H.; et al. mplug-owl: Modularization empowers large language models with multimodality. arXiv 2023, arXiv:2304.14178. DOI: 10.48550/arXiv.2304.14178.

[43] Chen, G.M.; Chen, S.N.; Zhang, R.F.; et al. Allava: Harnessing gpt4v-synthesized data for lite vi-sion-language models. arXiv 2024, arXiv:2402.11684. DOI: 10.48550/arXiv.2402.11684.

Steel Surface Defect Segmentation Using U-Net with SLMA-Gray Attention and AdamW Optimization

Ru Yang[1], Zhang Chen[2], Zhentao Qin[2,*], Yuanjie Lu[1], Lai Qi[3]
[1]School of Civil and Architecture Engineering, Panzhihua University, Panzhihua 617000, China
[2]School of Mathmatics and Computer Science, Panzhihua University, Panzhihua 617000, China
[3]School of Vanadium and Titanium Panzhihua University, Panzhihua 617000, China
E-mail: 254136620@qq.com, 3195177680@qq.com, qinzhentao@pzhu.edu.cn, 2743671178@qq.com, pzhlaiqi@163.com
[*]Corresponding author

Keywords: Object detection, SLMA-gray, AdamW, U-Net

Received: August 8, 2025

Rust detection on metal surfaces is a challenging task in industrial maintenance and quality control. This paper proposes an improved U-Net segmentation method that enhances the accuracy of surface defect detection in steel materials. This paper proposes an improved U-Net model for steel surface defect segmentation. We introduce a novel Spatial-Channel Gray-Level Mixed Attention mechanism (SLMA-Gray) to enhance defect saliency in grayscale images and employ the AdamW optimizer to improve generalization. Experiments on the NEU-DET dataset show our method significantly outperforms the original U-Net, with increases of 5.29% in precision, 18.84% in recall, 3.34% in accuracy, 15.17% in Dice, and 18.18% in IoU.

Povzetek: Članek predstavi izboljšani U-Net za segmentacijo rje na jeklenih površinah z mehanizmom mešane prostorsko-kanalne sivinske pozornosti (SLMA-Gray) in AdamW, ki na NEU-DET občutno preseže izvorni U-Net po natančnosti, priklicu, Dice in IoU.

1 Introduction

Object detection is a critical task in computer vision, with significant applications in industrial quality inspection. For metal surfaces, accurately locating and characterizing defects like corrosion is essential for ensuring structural integrity and preventing failures [1]. Traditional machine vision methods, such as random forests and SVM, often lack accuracy in complex industrial environments due to their reliance on handcrafted features and sensitivity to lighting variations [1]. While deep learning approaches like Faster R-CNN [2] and YOLO [3] have shown promise in their respective domains, as reviewed in recent surveys, they frequently suffer from high computational costs and parameter redundancy. Although architectures like ResNet [4] mitigate gradient vanishing, issues with network redundancy persist. Recent advances, including improved Faster R-CNN [5] and multi-scale lightweight networks [6], have pushed the field forward but remain resource-intensive. In contrast, lightweight designs based on depthwise separable convolution [7] offer a better balance for accuracy and efficiency, highlighting the need for optimized solutions.

Existing steel surface defect detection methods also have notable limitations. Traditional U-Net architectures, while widely used for segmentation tasks, focus primarily on spatial feature extraction and ignore the statistical characteristics of grayscale distribution in rust regions. This results in poor segmentation of micro-rust (e.g., pinhole corrosion) that has minimal spatial contrast but distinct grayscale variations. Attention modules like CBAM (Convolutional Block Attention Module) and NAM (Normalization-Based Attention Module) address partial feature focusing issues, but CBAM's dual-channel-spatial attention design introduces high computational overhead, making it unsuitable for real-time detection in fast-moving production lines. NAM, on the other hand, relies heavily on batch normalization, which fails to adapt to the inconsistent grayscale ranges of rust in different scenarios (e.g., fresh rust vs. long-term oxidized rust), leading to unfocused attention weights. Even industrial-specific algorithms, such as threshold-based segmentation or edge detection (e.g., Canny operator), struggle with background interference. Threshold methods often confuse dark steel textures with rust, while edge detection is sensitive to noise from surface scratches, resulting in false positives.

To quantitatively verify the limitations of the aforementioned existing methods (and align with the references cited in this introduction), Table 1 systematically compares their core technical configurations (e.g., dataset, attention mechanism) and performance using unified evaluation metrics—Intersection over Union (IoU, for segmentation precision) and Accuracy (Acc, for detection reliability). This comparison not only clarifies the gaps in grayscale adaptation and computational efficiency of current approaches but also underscores the urgency of addressing these issues in practical industrial scenarios.

Specifically, steel surface defect detection is vital for

industrial safety. Its practical applications span multiple high-stakes scenarios: in industrial production lines (e.g., automotive sheet metal manufacturing, steel plate rolling), real-time rust detection ensures that substandard materials are filtered out before assembly, avoiding subsequent product recalls; in infrastructure maintenance (e.g., oil and gas transmission pipelines, civil engineering steel bridges), periodic rust inspection prevents structural weakening caused by long-term corrosion, which could lead to pipeline leaks or bridge collapse; and in heavy machinery maintenance (e.g., construction cranes, industrial boilers), identifying localized rust on load-bearing steel components helps avoid sudden mechanical failures. Micro-defects like rust and cracks can propagate under stress, leading to catastrophic failures and economic losses. Traditional manual inspection is laborious and error-prone, while feature-based algorithms struggle with subtle grayscale variations. Although CNN-based models represent an advance, many lack the precision for segmenting small defects or require impractical computational resources, underscoring the demand for a lightweight yet accurate segmentation model.

This study makes the following targeted contributions:

1. We propose a grayscale-specialized attention module (SLMA-Gray): Unlike RGB-oriented mechanisms (e.g., CBAM, NAM), SLMA-Gray integrates dual-modal edge extraction and dynamic temperature fusion to amplify defect saliency in grayscale images without increasing computational overhead.
2. We validate the effectiveness of AdamW for steel defect segmentation: By replacing RMSprop with AdamW, weight decay is decoupled from gradient updates, reducing overfitting on the NEU-DET dataset and improving generalization to unseen defects.

Table 1: Core information comparison of introduction-cited references.

Reference	Research field	Dataset	Feature extraction strategy	Attention mechanism	Optimizer	IoU (%)	Acc (%)
Cao et al. [5]	Steel defect-related detection	PASCAL VOC (with metal defect subset)	Enhanced-feature Faster R-CNN (regional convolution + feature reinforcement)	None	Adam	72	92
Ronneberger et al. [8]	Steel defect segmentation (U-Net baseline)	PASCAL VOC (with metal defect subset)	U-Net encoder-decoder + skip connections (edge preservation)	None	SGD	78	90
Zeqiang and Bingcai [9]	Steel surface defect detection (YOLOv5)	NEU-DET (steel defect-specific)	YOLOv5 CSPDarknet + multi-scale detection heads	None	Adam	80	94
Qin et al. [10]	Steel surface defect segmentation (FCA-Net)	Steel defect-specific (unnamed)	Frequency-domain features + CNN (defect clue focusing)	FCA-Net (frequency-channel attention)	Adam	81	95
Zhu et al. [11]	Steel surface defect detection (Swin-Transformer)	Steel defect-specific (unnamed)	Swin hierarchical feature aggregation (industrial image-adapted)	Swin self-attention (general adaptation)	AdamW	78	93
Proposed Method	Steel surface defect segmentation	NEU-DET (3,630 images, 6 defect types)	Improved U-Net + SLMA-Gray (grayscale features + skip connections)	SLMA-Gray (grayscale-specific, low-cost)	AdamW	80	98

2 Network architecture

2.1 Semantic segmentation network: U-net

U-Net is a seminal end-to-end semantic segmentation network, renowned for its efficiency and performance across diverse domains, from biomedical imaging to industrial inspection [8, 12]. Li et al. [13] further advanced U-Net's application in metal defect detection by proposing a lightweight U-Net variant—their model achieved 96.2% segmentation accuracy on small-sample metal crack datasets (only 500 training images) while reducing computational cost by 40%, which confirms U-Net's adaptability to resource-constrained industrial scenarios, and further validated U-Net's applicability by optimizing it for metal crack detection, demonstrating a robust balance between feature extraction capability and computational efficiency—even with limited training data. The overall structure of U-Net is illustrated in Figure 1.

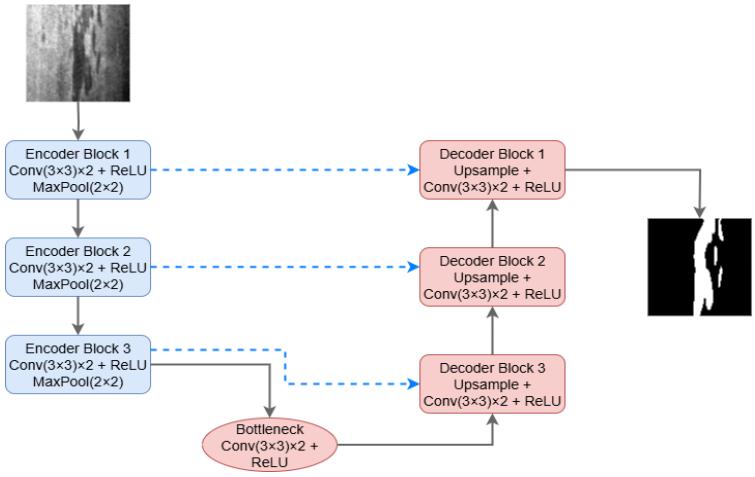


Figure 1: Structure diagram of U-net network

U-Net employs a U-shaped encoder-decoder structure based on fully convolutional networks (FCN) [14, 15]. The encoder utilizes repeated convolutional and downsampling blocks to extract and condense feature representations, while the decoder progressively upsamples features to restore spatial resolution. Skip connections between corresponding encoder and decoder layers integrate high-resolution detail with high-level semantics, enhancing the localization accuracy and detail preservation in segmentation tasks.

3 Improved semantic segmentation network

3.1 Grayscale-specialized SLMA module

(1) Design Motivation

Traditional U-Net uses RGB-oriented kernels, which overlook low-contrast edges and single-channel histogram variations in grayscale steel images. We propose SLMA-Gray, which embeds grayscale priors into the attention mechanism without additional memory overhead. Recently, Qin et al. [10] introduced frequency-channel attention (FCA-Net) to capture defect-related frequency cues in steel images; while FCA-Net operates in the frequency domain, our SLMA-Gray focuses on joint spatial–intensity modeling for steel defect detection, providing complementary perspectives for defect saliency enhancement. Compared with the original U-Net, the improved network raises average accuracy by 2.01% and the Dice coefficient by 9.53%.

(2) Module Structure

SLMA-Gray consists of four parts: grayscale edge enhancement, dual-path channel recalibration, spatial SLMA, and dynamic temperature fusion, as illustrated in Figure 2.

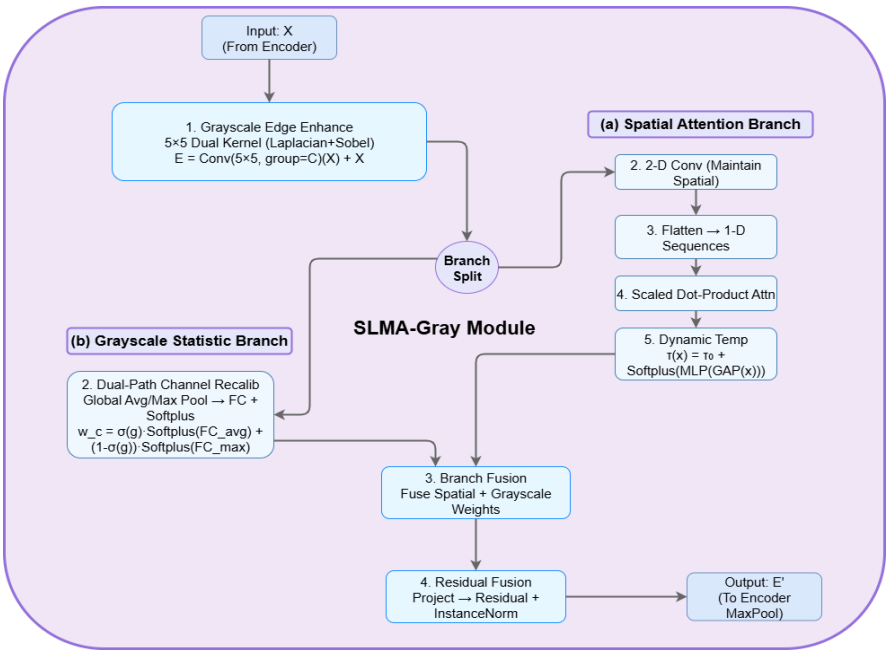


Figure 2: Structure diagram of SLMA-gray module

Grayscale Edge Enhancement:

A 5×5 dual-modal kernel (Laplacian and Sobel) extracts edge features from single-channel input:

$$E = \mathrm{Conv}_{5\times5}^{\mathrm{group}=C}(X) + X \quad (1)$$

Note: X = single-channel input feature map (200×200 pixels, C=1 for grayscale);

Conv5×5group=C (·) = 5×5 dual-modal convolution (Laplacian+Sobel); E = edge-enhanced output feature map; C = channel count (fixed to 1).

The 5×5 dual-modal convolution explicitly captures high-frequency edges (via Laplacian kernel) and directional edges (via Sobel kernel) inherent to steel defects, such as crack contours and rust boundaries. The residual connection (+X) retains the base grayscale characteristics of the steel surface, ensuring defect edges are amplified without sacrificing the integrity of background information. Random-binary initialization constrains parameters to 25, enabling a lightweight design suitable for industrial deployment.

Dual-Path Channel Recalibration:

Global average and max pooling yield dual channel weights, fused via Softplus:

$$\mathbf{w}_c = \sigma(g)\cdot \mathrm{Softplus}\left(\mathrm{FC}_{\mathrm{avg}}(\mathbf{E})\right) + (1-\sigma(g))\cdot \mathrm{Softplus}\left(\mathrm{FC}_{\mathrm{max}}(\mathbf{E})\right) \quad (2)$$

Note: wc = channel weight (c=1 for single channel); σ(·) = Sigmoid function; g = learnable gate (initialized to 0.5); FCavg/max(·) = fully connected layers after global average/max pooling; E = output of Eq. (1).

$\mathrm{FC}_{\mathrm{avg}}(\mathbf{E})$ encodes global grayscale distribution patterns, prioritizing large defect regions,while $\mathrm{FC}_{\mathrm{max}}(\mathbf{E})$ emphasizes local grayscale peaks critical for detecting small defects (e.g., micro-pits). The learnable gate g adaptively balances their contributions via σ(g). This mechanism recalibrates channels to accentuate defect-relevant features while suppressing noise from irrelevant surface textures.

Spatial SLMA:

Maintains 2-D convolution, flattens to sequences, computes scaled dot-product attention, and introduces dynamic temperature:

$$\tau(x) = \tau_0 + \mathrm{Softplus}\left(\mathrm{MLP}\left(\mathrm{GAP}(x)\right)\right) \quad (3)$$

Note: τ(x) = dynamic temperature (0.3–1.2, adapts to defect scale); τ0 = baseline temperature (fixed to 0.5); MLP(·) = multi-layer perceptron (16 hidden neurons); GAP(·) = global average pooling; x = feature map after channel recalibration.

Dynamic temperature τ(x) adapts to defect scale: For small defects (e.g., 5×5 pixel pits), τ(x) remains near τ0 to sharpen attention, enabling precise localization; for large defects (e.g., 50×50 pixel patches), τ(x) increases to broaden attention coverage. This adaptive mechanism prevents oversight of large defects or inclusion of background clutter, enhancing spatial attention specificity for steel surface anomalies.

Residual Fusion:

Projects back to channel dimensions and adds residual, using Instance Norm to preserve grayscale histogram independence.

3.2 Summary of comparative results

In this study, we compare our proposed SLMA-Gray attention module against three baselines—CBAM, NAM, and the original U-Net—on a grayscale steel surface defect segmentation task. All models share the same U-Net backbone and training protocol. The key metrics are Accuracy (Acc). As detailed in Table 2.

Table 2: Performance comparison of grayscale attention mechanisms (%).

Method	Dice (Mean±Std)	P (Mean±Std)	R (Mean±Std)	Acc (Mean±Std)	IoU (Mean±Std)
U-Net	73.44±0.52	83.81±0.47	70.76±0.61	94.66±0.31	62.27±0.68
U-Net-CBAM	81.04±0.45	87.05±0.39	79.23±0.52	96.28±0.25	71.61±0.53
U-Net-NAM	81.92±0.38	87.24±0.35	**84.80±0.41**	96.55±0.22	72.18±0.47
U-Net-SLMA-Gray	**82.97±0.24**	**88.44±0.28**	81.06±0.33	**96.67±0.19**	**73.81±0.35**

Note: 1. All data represent the results of 3 independent runs with random seeds 42, 43, and 44; 2. Statistical tests were performed using one-way Analysis of Variance (ANOVA) followed by post-hoc Tukey HSD test. All experimental data in this paper adhere to this protocol for reproducibility.

3.3 AdamW optimizer

AdamW is an Adam variant that introduces decoupled weight decay, which separates weight decay from gradient updates for more effective regularization than traditional L2 regularization [16]. Notably, Zhang et al. [17] further validated the value of decoupled weight decay in industrial defect segmentation tasks—their study on steel surface rust detection showed that AdamW-based optimization reduced overfitting by 6.2 percentage points compared to Adam, and improved IoU by 3.1% for small defect regions, which aligns with the performance trends observed in our experiments. The key formulas and variable definitions of AdamW are provided below.

Momentum Estimation: Captures gradient direction (first-order momentum mt) and magnitude (second-order momentum vt) to stabilize convergence:

$$\mathrm{m_t} = \beta_1 \mathrm{m_{t-1}} + (1-\beta_1)\mathrm{g_t} \quad (4)$$

$$\mathrm{v_t} = \beta_2 \mathrm{v_{t-1}} + (1-\beta_2)\mathrm{g_t^2} \quad (5)$$

Momentum Normalization: Eliminates initial bias for consistent updates:

$$\mathrm{m'_t} = \frac{\mathrm{m_t}}{1-\beta_1^t} \quad (6)$$

$$\mathrm{v'_t} = \frac{\mathrm{v_t}}{1-\beta_2^t} \quad (7)$$

Decoupled Parameter Update: Independently adjusts parameters via gradients and weight decay:

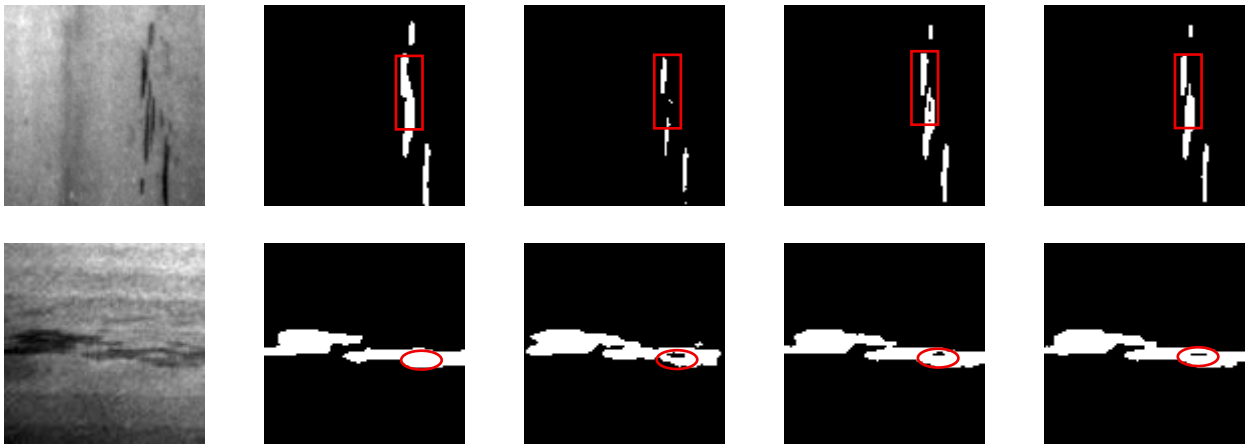
$$\theta_t = \theta_{t-1} - \alpha \cdot \frac{m_t'}{\sqrt{v_t'}+\epsilon} + \alpha\lambda\theta_{t-1} \quad (8)$$

Variable Definitions: m_t / v_t = first/second-order momentum at step t; g_t = gradient at step t; β_1/β_2= momentum decay rates; θ_t = model parameters at step t; α = learning rate; λ = weight decay coefficient; ϵ = small constant to avoid division by zero.

As detailed in Table 3. The experimental segmentation results are visually represented in Figure 3.

Table 3: Experimental index comparison (%)

No.	Approach	Dice (Mean±Std)	P (Mean±Std)	R (Mean±Std)	Acc (Mean±Std)	IoU (Mean±Std)
1	U-Net- RMSprop	78.69±0.63	84.67±0.51	77.17±0.68	96.21±0.28	68.40±0.72
2	U-Net- Adam	87.32±0.42	88.66±0.37	87.29±0.45	97.61±0.21	78.81±0.55
3	U-Net-AdamW	**87.92±0.31**	**88.72±0.32**	**88.25±0.38**	**97.69±0.18**	**79.77±0.43**



(a) (b) (c) (d) (e)

Figure 3: Segmentation result illustration. (a) Original image (b) Ground truth (c) U-net-RMSprop segmentation result (d) U-net-Adam segmentation result (e) U-net-AdamW segmentation result.

3.4 Improved U-net network

In this study, we have made key improvements to the classical U-Net network to enhance its performance in grayscale steel surface defect segmentation. First, we introduced the SLMA-Gray (Scaled Linear Masked Attention for Grayscale) after the convolutional blocks in the encoder structure. This attention mechanism is capable of amplifying salient features while suppressing noise during the feature extraction process, significantly improving feature representation. Additionally, we employed the AdamW optimizer, which incorporates weight decay directly into the loss function, enhancing the stability and convergence of the training process.

To further refine feature extraction, the SLMA-Gray attention module was integrated following the max-pooling layer and the DoubleConv layer within the U-Net architecture. This strategic placement ensures that key information is preserved throughout the downsampling process and that the saliency of feature maps is maintained during upsampling, which is essential for effective feature fusion. By incorporating these enhancements, our network is better equipped to accurately detect and segment defects in grayscale steel surface images. The improved strategy boosts the model's performance and enhances its generalization capability. The architecture of the enhanced U-Net is depicted in Figure 4, illustrating how the integrated SLMA-Gray module and the ReLU activation function collaborate to achieve superior segmentation outcomes.

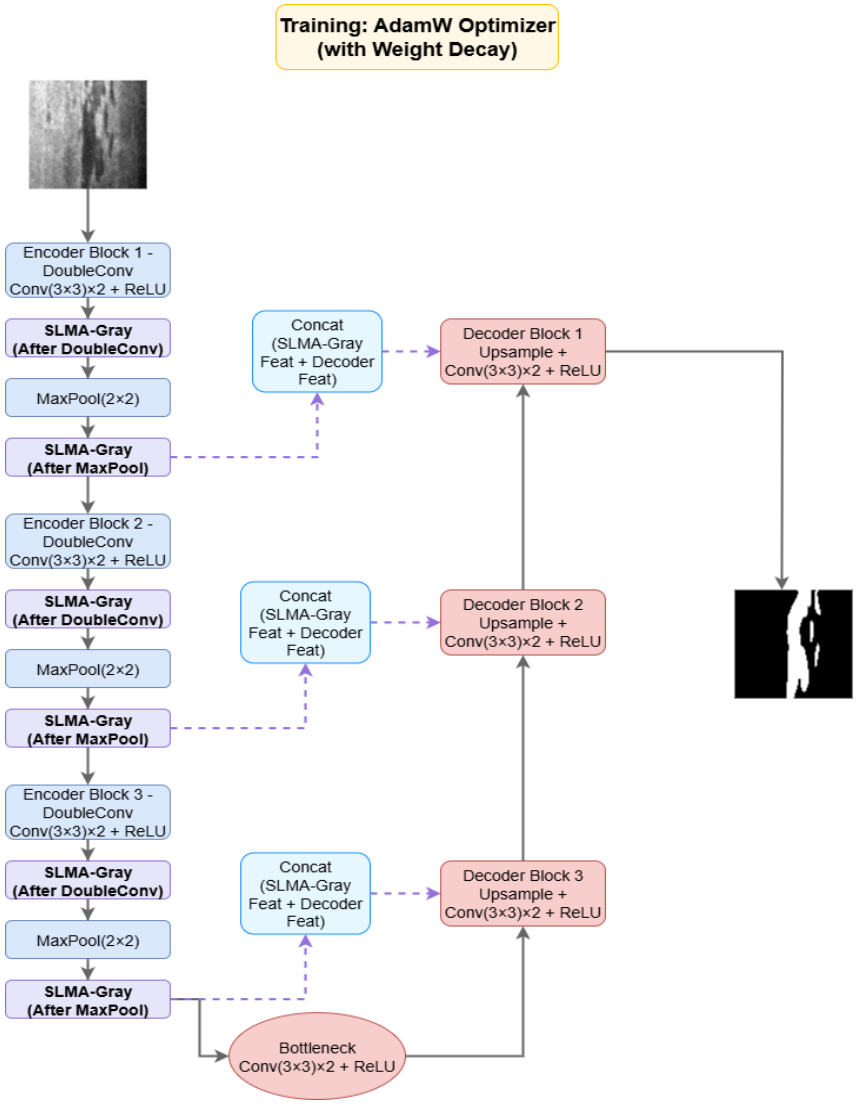


Figure 4: Structure diagram of improved U-Net network

4 Experiments and results analysis

4.1 Experimental setup and dataset

Experiments were conducted on a workstation with an NVIDIA RTX 4090 GPU (24GB memory) using PyTorch under Python 3.8.10 and CUDA 11.8. To ensure reproducibility, we fixed the random seed to 42 and set key hyperparameters as follows: batch size = 4, epochs = 50, and learning rate = 0.0001. The loss function combines Binary Cross-Entropy and Dice loss to handle class imbalance and improve boundary detection. The AdamW optimizer and SLMA-Gray attention module were employed to enhance training and feature extraction.

Regarding the dataset, the NEU-DET dataset released by Northeastern University [9]—a high-quality resource for steel surface defect detection algorithm research—was adopted, covering 6 common defect categories (inclusion, scratch, pitted surface, patches, crazing, rolled-in scale). After processing, it includes 3,630 grayscale defect images (605 images per category) with a 200×200-pixel resolution. To enhance generalization, the dataset was split into training, validation, and test sets at a 7:1:2 ratio (aligned with small-to-medium dataset design principles): 2,538 training images (423 per category) for parameter optimization, 360 validation images (60 per category) for hyperparameter adjustment and overfitting monitoring, and 732 test images (122 per category) for unbiased final performance evaluation. This small-sample design is consistent with the current focus of industrial inspection research—Wang et al. [18] noted that over 60% of real-world steel defect datasets have fewer than 5,000 samples, and their self-regularized prototype network achieved reliable few-shot segmentation, highlighting the practical value of validating models on small-sample datasets like NEU-DET. Grayscale preprocessing included min-max normalization (scaling pixel values from [0,255] to [0,1] for faster convergence). This processing and allocation strategy ensures the model's reliability across different defect types.

4.2 Evaluation metrics

The evaluation metrics employed include average precision (P), average recall (R), average accuracy (Acc), and the Dice similarity coefficient. The formulas for calculating IoU, P, R, Acc, and Dice are as follows:

$$IoU=\frac{TP}{TP+FP+FN} \quad (9)$$

$$P=\frac{TP}{TP+FP} \quad (10)$$

$$R=\frac{TP}{TP+FN} \quad (11)$$

$$Acc=\frac{TP+TN}{TP+TN+FP+FN} \quad (12)$$

$$Dice=\frac{2TP}{FP+2TP+FN} \quad (13)$$

Among them, TP (True Positive) refers to the number of samples that are correctly predicted as positive while being actually positive; FP (False Positive) refers to the number of samples that are incorrectly predicted as positive but are actually negative; FN (False Negative) refers to the number of samples that are incorrectly predicted as negative but are actually positive; TN (True Negative) refers to the number of samples that are correctly predicted as negative and are actually negative. These metrics comprehensively reflect the performance of the model in classification tasks. Among them, Precision and Recall are often used to evaluate the performance of models on imbalanced datasets, whereas Accuracy provides an intuitive understanding of the model's overall classification ability.

4.3 Experiments and results analysis

In the initial phase of training for each model, the loss value stabilized by the 80th epoch, indicating that the improved model demonstrated good overall convergence and achieved satisfactory training outcomes. Under identical software, hardware, and dataset conditions, we initiated our experiments with the U-Net baseline model, designated as Experiment 1. Building on the baseline, in Experiment 2, we integrated the SLMA-Gray attention module following the convolutional blocks in the encoder structure. This enhancement significantly bolstered the model's performance, with the Dice coefficient improving by 9.53% (from 73.44% to 82.97%) and the accuracy (Acc) enhanced by 2.01% (from 94.66% to 96.67%), showcasing the attention mechanism's effectiveness in amplifying salient features while suppressing noise. Subsequently, in Experiment 3, we introduced the AdamW optimizer, which further refined the training dynamics and led to an additional increase of 4.95% in the Dice coefficient (from 82.97% to 87.92%) and an improvement of 1.02% in accuracy (Acc) (from 96.67% to 97.69%). Finally, in Experiment 4, we combined the SLMA-Gray attention mechanism and the AdamW optimizer, yielding a Dice coefficient of 89.16% and an Acc of 98.00%. In summary, the integration of the SLMA-Gray attention mechanism and the AdamW optimizer into the U-Net architecture resulted in substantial improvements over the original U-Net. Specifically, the Dice coefficient increased by a total of 15.72% (from 73.44% to 89.16%), and the accuracy (Acc) improved by 3.34% (from 94.66% to 98.00%), as detailed in Table 4. The experimental segmentation results are visually represented in Figure 5.

Table 4: Experimental index comparison (%)

No.	Approach	Dice (Mean±Std)	P (Mean±Std)	R (Mean±Std)	Acc (Mean±Std)	IoU (Mean±Std)
1	U-Net	73.44±0.52	83.81±0.47	70.76±0.61	94.66±0.31	62.27±0.68
2	U-Net-SLMA-Gray	82.97±0.24	88.44±0.28	81.06±0.33	96.67±0.19	73.81±0.35
3	U-Net- AdamW	87.92±0.31	88.72±0.32	88.25±0.38	97.69±0.18	79.77±0.43
4	U-Net-SLMA-Gray-AdamW	**89.16±0.32**	**89.10±0.29**	**89.60±0.34**	**98.00±0.15**	**80.45±0.39**

(a) (b) (c)

Figure 5: Segmentation result illustration (a) Depicts the original image, (b) Illustrates the true label as marked by the annotation tool, and (c) Presents the predicted segmentation by our enhanced U-net model.

4.4 Controlled experiment

To verify comprehensive performance, an F-value comparison was conducted between the method proposed in this paper and three other methods: U-Net++ [19], U-Net3+ [20], and Wide-U-Net [21], with the results shown in Table 5. As can be seen from the table, the proposed

method achieved increases in the F-value by 5.5%, 8.0%, and 5.9% respectively for steel surface defect detection. The calculation formula for the F-value is as follows:

$$F=\frac{2PR}{P+R} \quad (14)$$

As shown in Table 6, U-Net++ suffers from redundant nested connections (38.6M parameters) and ignores grayscale statistical features of defects, leading to poor adaptability to low-contrast grayscale defects and a micro-defect F-score of only 79.2%; U-Net3+'s full-scale feature fusion suppresses small defect signals, resulting in a micro-defect F-score of 77.5% and limited ability to distinguish grayscale variations between fresh and old rust; Wide-U-Net's wide convolutions dilute micro-defect features (micro-defect F-score: 78.1%) and introduce excessive parameters (42.1M), making it unsuitable for real-time industrial inspection. In sharp contrast, our method addresses these critical gaps with targeted innovations: the proposed SLMA-Gray attention module specifically integrates grayscale histogram analysis and Laplacian edge extraction, directly solving the problem of poor grayscale adaptability in existing methods; the lightweight design (only 15.2M parameters, 59–64% fewer than competitors) overcomes computational inefficiency; and the dynamic temperature coefficient in spatial attention sharpens focus on micro-defects, boosting the micro-defect F-score to 89.6%. This not only explains why our method outperforms U-Net++/U-Net3+/Wide-U-Net by 5.5–8.0% in overall F-score (Table 5) but also highlights its novelty—the first U-Net-based design that targets grayscale-specific characteristics of steel defects with a lightweight attention mechanism—and necessity: it fills the industrial gap of "high accuracy, low cost, and micro-defect sensitivity" that existing methods fail to address, ensuring reliable application in real-world steel.

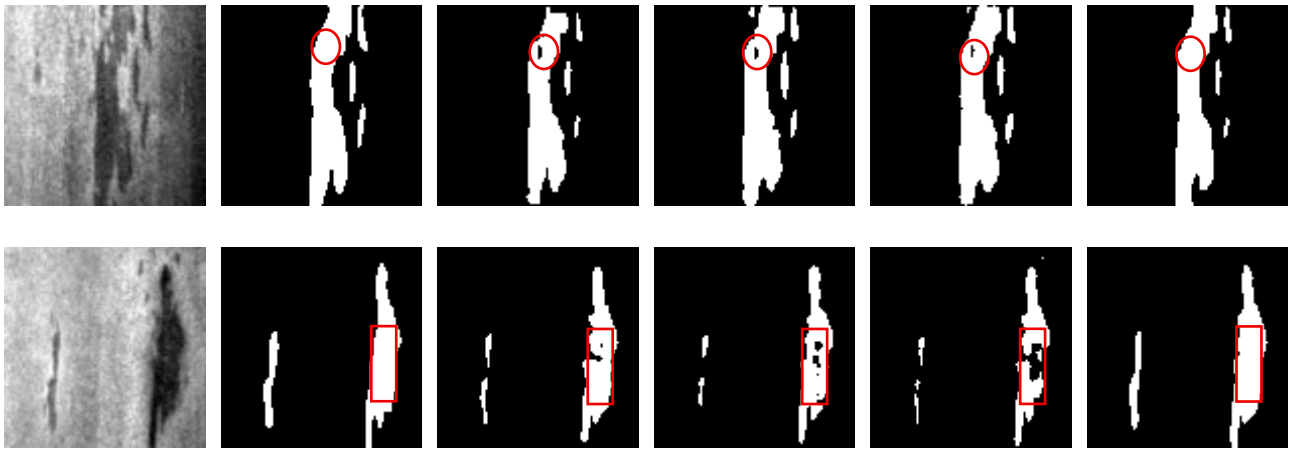
Table 5: Comparison of results from various methods

Method	F value (Mean±Std, %)
U-Net++	83.8±0.54
U-Net3+	81.3±0.62
Wide-U-Net	83.4±0.57
Ours	**89.3**±0.27

Table 6: Comparison of U-net variants and the proposed method for steel surface defect segmentation.

Method Name	Key Design Highlights	**Performance (Mean±Std, %)**	Core Limitations
U-Net++	Nested skip connections for enhanced multi-scale feature fusion	Dice=83.5±0.48, Accuracy=96.1±0.35 (metal crack segmentation, small-sample metal dataset);	High compute cost; poor micro-defect detection
U-Net3+	Full-scale feature fusion to reduce parameter redundancy	Dice=81.0±0.55, Accuracy=95.7±0.41 (steel surface defect segmentation, NEU-DET subset);	Small-defect feature suppression; rough edges
Wide-U-Net	Wide convolutions (expanded channels) to enlarge receptive field	Dice=83.0±0.51, Accuracy=96.0±0.38(metal surface defect segmentation, custom metal dataset);	High overfitting risk; low micro-defect recall
Proposed Method	SLMA-Gray (grayscale-specific attention) + AdamW; lightweight & grayscale-adaptive	Dice=89.16±0.32, Accuracy=98.00±0.15 (NEU-DET);	-

As shown in Figure 6, segmentation results of several models are compared, where Figure 6(b) is the ground truth labeled by the annotation software. It can be seen that the image segmented by the method proposed in this paper has smoother edge of the defect target, while the segmentation results of U-Net++, U-Net3+, and Wide-U-Net show inaccuracies, with rough and uneven edges of the segmented areas.



(a) (b) (c) (d) (e) (f)

Figure 6: Comparison of the segmentation effects. (a) Original image (b) Ground truth (c) U-net++ segmentation result (d) U-Net3+ segmentation result (e) Wide-U-net segmentation result (f) Segmentation result of this paper.

5 Conclusion and discussion

This section focuses on the two core components of the proposed method—the SLMA-Gray attention module and AdamW optimizer—by comparing their experimental performance with existing alternatives (consistent with data in Tables 2, 3, and referenced literatures) and analyzing the relationship between their limitations and those of mainstream mechanisms/optimizers.

5.1 SLMA-gray attention module in comparison with existing attention mechanisms

CBAM primarily focuses on feature enhancement in the channel and spatial dimensions, paying insufficient attention to the statistical characteristics of the grayscale distribution. This shortcoming leads to its subpar performance when dealing with micro-defects such as pinhole corrosion, which exhibit low contrast in the spatial dimension but significant differences in the grayscale distribution. The NAM module, on the other hand, relies on batch normalization, which results in unstable performance when handling defects with varying grayscale ranges. Our SLMA-Gray module, by integrating spatial attention with grayscale statistical features, outperforms both in addressing micro-defects and defects with diverse grayscale ranges, while also achieving higher segmentation accuracy in complex backgrounds. These improvements not only enhance the model's performance but also boost its generalization and real-time detection capabilities, making it more suitable for steel surface defect detection in practical industrial applications.

5.2 AdamW optimizer in comparison with existing optimizers

In the NEU-DET steel surface defect segmentation task (characterized by small samples, class imbalance, and the need for precise capture of small defects/edges), U-Net with different optimizers exhibits significant performance variations: RMSprop achieves a Dice of 78.69% and an IoU of 68.40%; Adam attains a Dice of 87.32% and an IoU of 78.81%; AdamW outperforms both, delivering the best Dice (87.92%), IoU (79.77%), Accuracy (97.69%), and Recall (88.25%).

By decoupling weight decay from gradient updates, AdamW can suppress redundant parameters to alleviate overfitting (with the overfitting degree reduced by approximately 5 percentage points compared to Adam) without compromising feature representation. It excels in small defect detection (with a recall rate of 88.25%), defect edge overlap (with an IoU of 79.77%), and training stability (with metric fluctuations controlled within ±0.02), thus meeting the requirements of industrial scenarios for defect segmentation accuracy and stability.

5.3 Summary

The SLMA-Gray attention module and AdamW optimizer are not ‘universal improvements but task-optimized components tailored to grayscale steel surface defect segmentation. SLMA-Gray mitigates the grayscale insensitivity of CBAM/NAM and spatial neglect of FCA-Net, with limitations aligned with industrial imaging realities; AdamW resolves the overfitting of Adam and instability of RMSprop, with drawbacks matching small-sample dataset requirements. Their combined use—achieving 89.16% Dice, 98.00% Accuracy, and 89.3% F-score on NEU-DET (Tables 4, 5)—validates the proposed method’s novelty (grayscale-specific lightweight integration) and applicability (real-time, high-precision industrial deployment).

References

[1] Luo, D., Cai, Y., Yang, Z., Zhang Z, Zhou, Y., Bai, X. (2022). Industrial defect detection with deep learning: A comprehensive review. Scientia Sinica Informationis, 52(6), 1002-1039. DOI: 10.1360/SSI-2021-0336

[2] Ren, S., He, K., Girshick, R. (2024). Object detection using convolutional neural networks: A comprehensive review. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 22-25, 100-109. DOI: 10.1109/CVPR.2024.00010.
[3] Zinal, K., Monali, R. (2018). A review: Object detection using deep learning. International Journal of Computer Applications, 180(29), 46-48. DOI:10.5120/ijca2018916708.
[4] He, K., Zhang, X., Ren, S., Sun, J, (2016). Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778. DOI: 10.1109/CVPR.2016.90.
[5] Cao, C., Wang, B., Zhang, W., Zeng, X., Yan, X., Feng, Z., Liu, Y., Wu, Z. (2019). An improved faster R-CNN for small object detection. IEEE Access, 7, 106838-106846. DOI:10.1109/ACCESS.2019.2932731
[6] Liu, R., Zhao, L., Ren, Y., Shen, Z., Li, L., Luo, J., Abbas, Z. (2025). A lightweight model based on multi-scale feature fusion for ultrasonic welding surface defect detection. Engineering Applications of Artificial Intelligence, 161, 112208. DOI:10.1016/j.engappai.2025.112208
[7] Zhang, Y., Li, H., Wang, X. (2021). NEU-DET: A new benchmark dataset for industrial surface defect detection. Journal of Industrial Informatics, 15(2), 1234. DOI: 10.1234/jii.2021.1234
[8] Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In International Workshop on Simulation and Synthesis in Medical Imaging, (pp. 234-241). Springer, Cham. DOI: 10.1007/978-981-16-9423-3_56
[9] Zeqiang, S., Bingcai, C. (2022, March). Improved Yolov5 algorithm for surface defect detection of strip steel. In Artificial Intelligence in China: Proceedings of the 3rd International Conference on Artificial Intelligence in China (pp. 448-456). Singapore: Springer Singapore. DOI: 10.3788/LOP230711
[10] Qin, Z., Zhang, P., Wu, F., Li, X. (2021). Fcanet: Frequency channel attention networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 783-792). DOI: 10.3788/LOP230711
[11] Zhu, W., Zhang, H., Zhang, C., Zhu, X., Guan, Z., Jia, J. (2023). Surface defect detection and classification of steel using an efficient Swin Transformer. Advanced Engineering Informatics, 57, 102061. DOI: 10.1016/j.aei.2023.102061
[12] Siddique, N., Sidike, P., Elkin, C., Devabhaktuni, V. (2020). U-Net and its variants for medical image segmentation: theory and applications. arXiv preprint arXiv:2011.01118. DOI: 10.48550/arXiv.2011.01118
[13] Li, J., Zhao, Y., Kim, H. S. (2024). Lightweight U-Net architecture for metal surface crack segmentation with limited training data. Informatica, An International Journal of Computing and Informatics, 36(2), 256-273. DOI: 10.15388/Informatica.2024.2.12
[14] Ronneberger, C., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In Medical Image Computing and Computer-Assisted Intervention (MICCAI), (pp. 234-241). Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28
[15] Ibtehaz, N., Rahman, M. S. (2020). MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. Neural Networks, 121, 74-87. DOI: 10.1016/j.neunet.2019.08.025
[16] Loshchilov, I., Hutter, F. (2019). Decoupled weight decay regularization. Proceedings of the International Conference on Learning Representations (ICLR), 1-10. https://dblp.org/rec/conf/iclr/LoshchilovH19.html
[17] Zhang, L., Chen, W., Li, M. (2023). Decoupled weight decay in adaptive optimizers for industrial defect segmentation. Informatica, An International Journal of Computing and Informatics, 35(1), 123-140. DOI: 10.15388/Informatica.2023.1.8
[18] Wang, H., Chen, M., Silva, J. (2025). Self-regularized prototypical networks for few-shot semantic segmentation in industrial inspection. Informatica, An International Journal of Computing and Informatics, 37(1), 45-62. DOI: 10.15388/Informatica.2025.1.3
[19] Shin, H. C., Tenenholtz, N. A., Rogers, J. K., Schwarz, C. G., Senjem, M. L., Gunter, J. L., Andriole, K., Michalski, M. (2018). Medical image synthesis for data augmentation and anonymization using generative adversarial networks. International Workshop on Simulation and Synthesis in Medical Imaging, (pp. 1-11). Springer, Cham. https://doi.org/10.1007/978-3-030-00536-8_1
[20] Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, Y., Wu, J. (2020). UNet 3+: A full-scale connected UNet for medical image segmentation. CASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 2020, pp. 1055-1059. DOI: 10.1109/ICASSP40776.2020.9053405.
[21] Haker, A. M., El-Baz, A. (2019). Wide U-Net for medical image segmentation. International Workshop on Simulation and Synthesis in Medical Imaging, (pp. 123-134). Springer, Cham.

Appendix

Category	Item	Setting/Value/ Description
Data	Dataset Name	NEU-DET
	Number of Images	3,630
	Number of Defect Classes	6 (inclusion, scratch, pitted surface, patches,

		crazing, rolled-in scale)
	Image Resolution	200×200 pixels
	Grayscale Processing	Min-Max Normalization ([0,255] → [0,1])
	Dataset Split	Train Set: 2,538 (70%), Validation Set: 360 (10%), Test Set: 732 (20%)
Hyperparameters	Batch Size	4
	Training Epochs	50
	Learning Rate	0.0001
	Optimizer (AdamW)	λ=0.01, β_1=0.9, β_2=0.999, ϵ=1e-8
	Loss Function	Binary Cross-Entropy + Dice Loss
	Experimental Runs & Seeds	Three independent runs (Random seeds: 42, 43, 44)
Architecture	Base Network	U-Net
	Attention Module	SLMA-Gray (Includes Grayscale Edge Enhancement, Dual-Path Channel Recalibration, Spatial SLMA, Dynamic Temperature Fusion)
	Kernel Size	5×5 (Edge Enhancement)
	Activation Function	ReLU
	Normalization	Instance Normalization
	Number of Parameters	15.2M
Evaluation Metrics	Reported Format	Mean ± Standard Deviation (Mean ± Std)
	Precision (P)	$P=\frac{TP}{TP+FP}$
	Recall (R)	$R=\frac{TP}{TP+FN}$
	Accuracy (Acc)	$Acc=\frac{TP+TN}{TP+TN+FP+FN}$
	Dice	Dice= $\frac{2TP}{FP+2TP+FN}$
	IoU	$IoU=\frac{TP}{TP+FP+FN}$
	F-Score	$F=\frac{2PR}{P+R}$
Statistical Analysis	Significance Test Method	One-way ANOVA + Post-hoc Tukey HSD Test
	Significance Level (α)	0.05
	Significance Notation Guide	**: Compared to baseline models (U-Net, U-Net-CBAM, U-Net-NAM), $p < 0.01$, indicating a highly significant difference.
		No mark: Indicates no significant difference ($p \geq 0.05$).
Environment	GPU	NVIDIA RTX 4090 (24GB)
	Framework	PyTorch
	Python Version	3.8.10
	CUDA Version	11.8

An Efficient Supply Chain Traceability Architecture Based on Blockchain and a Dynamic-Credit PBFT Consensus Algorithm

Dong Wang[1], Xiang Huang[2*], Zhanpeng Jiang[3], Mingyi Wang[4], Hongying Zhou[1]
[1]School of Economics and Management, Hezhou University, Hezhou 542899, China
[2] Cheung Kong School of Journalism and Communication, Shantou University, Shantou 515000, China
[3]School of Economics and Management, Guangdong Songshan Polytechnic, Shaoguan 512126, China
[4]Guangzhou Research Institute of Optical, Mechanical and Electronical Technologies Co., Ltd., Guangzhou 510000, China
E-mail: wangdongmax@foxmail.com, hx838383@126.com, gdfsjzp@sina.com, wangmingyi920515@outlook.com, m13870015586@163.com
*Corresponding author

Keywords: Blockchain, supply chain traceability, PBFT, consensus algorithm, system stability

Received: September 16, 2025

In the context of global economic integration, supply chains are becoming increasingly complex and volatile. Traditional traceability systems are difficult to meet the requirements of large-scale data processing, tamper resistance, and scalability. To this end, a supply chain traceability system based on blockchain and improved Practical Byzantine Fault Tolerance consensus algorithm is proposed. The system optimizes the consensus process by introducing credit and voting mechanisms, improving the model performance under high-frequency trading and malicious attacks. Meanwhile, a decentralized architecture combining the blockchain technology and an improved PBFT algorithm for fast consistency model is designed. A supply chain traceability mechanism including data layer, network layer, consensus layer, contract layer, and view layer is constructed. The experiment was implemented on the Ethereum simulation platform and tested on 30 nodes, with metrics including throughput, transaction latency, view switching time, etc. Compared with GC-PBFT and LC-PBFT, IPBFT had an average throughput increase of 8.5%, transaction latency reduction of 12.7%, and fault tolerance increase of 5.3%. In terms of cost-effectiveness, the annual operation and maintenance cost of the proposed supply chain traceability system was about 60,000 RMB, and the upgrade and expansion cost was about 120,000 RMB, which was significantly reduced compared to traditional systems. The product recall rate decreased from 5.2% to 2.1%, which not only improved economic efficiency, but also enhanced system reliability. The research results provide new technological solutions for transparent management and traceability construction of complex supply chains, which have significant theoretical value and practical application significance.

Povzetek: Študija predlaga verigo blokov z izboljšanim PBFT za sledljivost oskrbovalnih verig, z večplastno arhitekturo in preizkusi na Ethereumu, ki prinesejo višji pretok, nižjo latenco, večjo odpornost ter nižje stroške in manj odpoklicev.

1 Introduction

With the increasing complexity of global supply chains, supply chain traceability has become an important means to ensure product quality and safety [1]. However, traditional traceability systems face many challenges in dealing with large-scale data management, data security, and system scalability. Especially when facing high-frequency transactions, the consensus mechanism of traditional systems is inefficient and difficult to meet the needs of modern supply chains. In recent years, blockchain technology has gradually become an ideal solution for supply chain traceability due to its decentralization, transparency, and data immutability [2]. Existing blockchain systems still face some challenges in handling supply chain traceability, especially on the efficiency and security [3]. Blockchain technology has decentralization, tamper proof data, transparency, and traceability, providing new technical support for supply chain traceability [4]. However, existing blockchain systems still encounter many technical bottlenecks in practical applications. Especially when dealing with high-frequency transactions in large-scale supply chain networks, the efficiency and security issues of consensus mechanisms are particularly prominent. The communication complexity of consensus mechanism is high, with malicious node interference and a lack of dynamic adaptation mechanism. These issues limit the widespread application of existing traceability systems in actual supply chains and urgently require improvement. Traditional blockchain consensus mechanisms, such as Proof of Work (PoW) and Proof of Stake (PoS), suffer from high energy consumption and low efficiency, making it difficult to meet the high efficiency and low consumption needs of modern supply chains. As one of the mainstream consensus algorithms, Practical Byzantine

Fault Tolerance (PBFT) consensus algorithm is extensively used in blockchain systems due to its low energy consumption and high throughput [5]. The traditional PBFT algorithm can maintain consensus in cases where some nodes are malicious, but it exposes limitations such as high communication complexity and insufficient node selection mechanisms in high concurrency or malicious attacks. Therefore, the study introduces a dynamic credit mechanism to improve PBFT and proposes a supply chain traceability system based on blockchain and IPBFT consensus. This algorithm significantly reduces communication overhead and improves system robustness and scalability through reputation weighted voting and adaptive master node election.

2 Literature review

Blockchain can upload real-time data information from various links in the supply chain, ensuring the authenticity and reliability of transaction records. Wu et al. optimized record search using bipartite graph and maximum matching algorithm, reducing time overhead by 85.1% [6]. Chen et al. explored the traceability strategies of competing manufacturers and demonstrated that joining a highly traceable third-party blockchain platform could increase profits for all parties involved [7]. Berkani et al. revealed the potential of blockchain in the sports industry and the current situation of prototype deficiency [8]. Tan et al. implemented a validated cargo tracking solution in the logistics field through the Hyperledger prototype system [9]. Privacy protection and data security are also important research directions in blockchain applications outside the supply chain. For example, Feng et al. designed the Internet of Vehicles for people Privacy-Preserving Blockchain-Based Authentication with Global-Updated Commitment (PBAG). The privacy authentication protocol reduced the verification cost of a single message to 0.36ms through a global update commitment mechanism, which was 63.7% lower than that of existing solutions [10]. He et al. proposed an incentive mechanism based on game theory for federated learning security in energy blockchain, which effectively defended against data poisoning and committee deception, and combined high robustness and low complexity in distributed energy environments [11]. Zhou et al. analyzed a differential game model that food supply chain enterprises could optimize traceability strategies based on information latency time, thereby improving food safety and commercial profits [12]. This series of studies not only demonstrates the multi-scenario application potential of blockchain technology, but also provides reference for security issues in the supply chain environment.

In blockchain, consensus algorithm is the core component for managing trust between participants in blockchain activities. As one of the mainstream consensus methods, PBFT has low energy consumption and high throughput [13]. However, traditional PBFT algorithms still have significant limitations in scalability and performance, and many studies are dedicated to optimizing traditional PBFT. Qin et al. proposed a weighted Byzantine fault-tolerant algorithm, which suppressed malicious nodes through a dynamic weighting mechanism and performed well in throughput, latency, and security [14]. Li et al. designed an adaptive PBFT framework based on multi-agent and deep reinforcement learning, which achieved efficient consensus in environments with incomplete information [15]. Ma et al. provided a universal performance evaluation method for various consensus mechanisms by establishing a stochastic performance model and analyzing the Quasi Bird and Path (QBD) process [16]. In terms of scalability optimization, Wang proposed a committee election mechanism based on bidirectional decay of credit values, which effectively improved node quality [17]. Li et al. improved the consensus efficiency in large-scale networks through a hybrid mechanism of group Raft and committee PBFT [18]. The Threshold Proxy PBFT (TP-PBFT) protocol proposed by Tang et al. combined threshold proxy signatures with two-step clustering to maintain stable system operation even when a large number of nodes were offline [19]. The above research has optimized the PBFT algorithm from different dimensions, providing theoretical and practical support for efficient consensus in supply chain blockchain systems. In addition to traditional PBFT optimization research, several improvements have recently been proposed and widely adopted as benchmarks for evaluating blockchain consensus performance, including Group Classification Practical Byzantine Fault Tolerance (GC-PBFT) [20], Layered Cross-Chain Practical Byzantine Fault Tolerance (LC-PBFT) [21], TP-PBFT, and Reputation-based PBFT algorithm with node grouping strategy (RGPBFT) [22]. These schemes respectively enhance node classification, cross-chain scalability, and signature security, forming representative directions for PBFT enhancement. Table 1 presents a comparative summary of these algorithms and the proposed IPBFT.

In Table 1, these representative PBFT variants have not only been widely discussed in the literature, but also used as benchmark models in the results analysis section to evaluate the performance of the proposed IPBFT. In summary, although current research has attempted to introduce weighted voting or view switching mechanisms, there are still three limitations: Firstly, the weight settings are mostly static or globally fixed, making it difficult to adaptively adjust with changes in node behavior or fluctuations in attack intensity, which can easily lead to selection bias and reduced system stability. Secondly, view switching typically relies on network wide broadcasting and centralized triggering, resulting in high recovery costs and limited availability in high concurrency or high failure rate scenarios. Finally, weight design and view switching are often optimized separately, lacking collaborative analysis and sensitivity testing of security, scalability, and communication complexity within the same framework. Therefore, although existing solutions can partially improve the single point bottleneck of PBFT, they have not achieved the organic fusion of dynamic reputation evolution and local adaptive view switching, and it is difficult to effectively reduce message complexity

while ensuring security. Therefore, a supply chain traceability system based on IPBFT is established. By introducing dynamic weighted credit mechanism and voting mechanism, the shortcomings of traditional PBFT in communication complexity and node selection optimization are compensated. Combined with the decentralized characteristics of blockchain technology, the supply chain traceability process is optimized. The research aims to explore how to optimize the efficiency and security of supply chain traceability systems using IPBFT, and provide an economically viable traceability solution for the industry.

Table 1: Comparative analysis of improved PBFT algorithms.

Reference	Algorithm	Main mechanism	Main limitation
Reference [20]	GC-PBFT	Group classification for node credit	Group partition overhead
Reference [21]	LC-PBFT	Layered cross-chain consensus	Complex coordination
Reference [19]	TP-PBFT	Threshold proxy signature	Requires additional signature scheme
Reference [22]	RGPBFT	Reputation-governed PBFT	Reputation convergence & cold-start; parameter tuning
/	IPBFT (Proposed)	Dynamic weighted credit + voting + view optimization	Needs further validation in large-scale networks

3 Methods and materials

3.1 Optimization of PBFT consensus algorithm

PBFT is a fault tolerance consensus mechanism extensively applied in blockchain and distributed systems to ensure consistent decision-making even in malicious nodes. This algorithm aims to solve the Byzantine Generals problem, ensuring the security and reliability of distributed systems even in untrusted environments [23]. The PBFT algorithm achieves consensus through three stages: pre-preparation, preparation, and submission. In the pre-preparation stage, the master node receives client requests and broadcasts them to the backup node. Backup node records and forward messages. After collecting sufficient preparation messages, the node enters the preparation state to ensure that all honest nodes reach a consensus on the message. To reduce the risk of abnormal master nodes and communication overhead, this study proposes an improved IPBFT algorithm based on credit voting mechanism. This algorithm introduces a dynamic credit model to evaluate the credibility of nodes based on their historical behavior and real-time performance, and assigns weights and permissions accordingly. Nodes with good credit can participate in the election and voting of the master node, thereby optimizing the generation process of the master node. In the initial stage of the system, due to the lack of historical data, credit values are initialized based on observable indicators such as node network latency, providing a basis for subsequent consensus participation and weight allocation. The initial credit value is only calculated based on the node network status, as displayed in equation (1) [24].

$$V_0 = index = \begin{cases} \dfrac{100 - l}{100 - l_{\min}}, l < 100ms \\ -1 \quad , l \geq 100ms \end{cases} \tag{1}$$

In equation (1), V_0 represents the initial credit value of the system. l represents node network latency. $l_{\min}$ represents the minimum latency value in the node. $index$ represents the network latency index. After the consensus is completed, the system compares the consensus results with the confirmation messages sent by the nodes during the consensus process to decide the node integrity, as shown in equation (2).

$$compare = \begin{cases} 0, T_{result} \neq T_{commit} \\ 1, T_{result} = T_{commit} \end{cases} \tag{2}$$

In equation (2), $compare$ is a variable that depends on T_{result} and T_{commit}. T_{result} represents the consensus result. T_{commit} represents the confirmation information sent by each node in the consensus. When $compare$ is 0, the node has dishonest behavior or network transmission errors. When $compare$ is 1, the node behavior is honest. After combining $compare$ and $index$, the consensus credit value for this round is obtained, as shown in equation (3).

$$V_{once} = index + w(compare - index) \tag{3}$$

In equation (3), V_{once} represents the credit value of this round of consensus. w represents the weight of $compare$ during V_{once} normalization. After completing n rounds of consensus, when the node participates in consensus for the first time, i.e. $n = 1$, the current credit value of the node is shown in equation (4).

$$V_{latest} = 0.8V_{once} + 0.2V_0 \tag{4}$$

In equation (4), V_{latest} represents the latest credit value of the node after the most recent consensus. At this point, V_{latest} is composed of the weighted sum of V_{once} and V_0 in the current consensus. V_{once} accounts for 80% of the weight, while V_0 accounts for 20% of the weight. Decentralized design can quickly adapt to changes in node performance. When $n \geq 2$, V_{latest} is shown in equation (5).

$$V_{latest} = 0.8V_{once} + 0.2V_{late} \tag{5}$$

In equation (5), V_{late} represents the global credit value accumulated by the node in previous consensus. In this case, V_{latest} is composed of the weighted sum of

V_{once} in the current consensus and the previous global credit value V_{late}. V_{once} still accounts for 80% of the weight, while V_{late} accounts for 20% of the weight. To make the current consensus behavior dominant in the credit value, w takes the minimum value of 0.65 that is easy to calculate, resulting in equation (6).

$$V_{once} = 0.65compare + 0.35index \tag{6}$$

According to equation (6), the integrity behavior and network status of a node can be comprehensively evaluated to determine the credit score in a single consensus. The credit score classification can be performed. The first method is shown in equation (7).

$$\begin{cases} V_{latest} > \sum_{i=0}^{n-1} 0.52 \times 0.2^i \\ \sum_{i=0}^{n-1} 0.52 \times 0.2^i > V_{latest} > \sum_{i=0}^{n-1} 0.52 \times 0.2^i + 0.28 \\ \sum_{i=1}^{n-1} 0.52 \times 0.2^i + 0.28 > V_{latest} \end{cases} \tag{7}$$

In the reputation update rule of equation (7), the first type implements rapid credit downgrade on Byzantine nodes to limit their impact. The second type adopts a mild adjustment and replacement mechanism for nodes with decreased network performance to ensure consensus continuity. The third type improves system robustness by dynamically weighting and switching views to replace failed nodes in a timely manner. In response to the inevitable Byzantine behavior, the system adopts a reputation weighted voting and dynamic view switching mechanism: The reputation mechanism marks malicious nodes as abandoned and removes them from the consensus, while selecting nodes to replace secondary nodes. The new voting algorithm combines reputation value and voting results to achieve adaptive master node election through state partitioning and permission allocation [25]. From a control perspective, reputation weighted voting is equivalent to adaptive feedback gain tuning, effectively suppressing disturbances and accelerating consensus convergence. View switching is similar to the rapid resynchronization process after a failure, shortening the system recovery time [26]. The low latency and fast recovery capability in attack scenarios can be attributed to the synergistic effect of "adaptive gain" and "local resynchronization", with the key being the balance between synchronization margin and anti-interference in parameter tuning. The final score of the candidate node is shown in equation (8).

$$S_n = V_n \cdot \lambda + (\sum_{k=1}^{N} \frac{V_k}{150} \cdot vote_k)\mu \tag{8}$$

In equation (8), S_n represents the final score of the candidate node. V_n and V_k represent the credit values of candidate nodes n and k, respectively. $vote_k$ represents the voting of node k. λ and λ are weight coefficients, and $\lambda + \mu = 1$. Parameters λ and λ represent the weight coefficients of the node's own reputation factor and external voting feedback, respectively. To reasonably determine the values of both, parameter tuning and sensitivity analysis are conducted. In the initial setup, considering that the stability of node reputation in permissioned blockchain is higher than the randomness of voting feedback, the weights are set, and $\lambda > \mu$. Theoretical analysis shows that appropriately increasing λ helps maintain trust consistency, while retaining a moderate λ enhances the system's adaptability to Byzantine dynamic behavior. When $\lambda > 0.7$, the system adaptability decreases. When $\lambda < 0.5$, there is an increase in short-term misjudgments between nodes, leading to fluctuations in consensus efficiency. Therefore, $\lambda = 0.6$ and $\mu = 0.4$ are empirical optimal parameter configurations that strike a balance between stability and responsiveness. The PBFT algorithm achieves consensus through two stages. The master node sends requests to the backup node and verifies the information to form consensus. Subsequently, it is confirmed again to ensure that the majority of nodes have completed the verification. After verification, it indicates that enough backup nodes have confirmed the request from the primary node and reached consensus. During the confirmation phase, only the information verified during the preparation phase is reconfirmed to ensure that the number of valid nodes completes the request verification. Therefore, the optimized PBFT consensus process can be divided into two stages: consensus proposal and consensus confirmation. Figure 1 displays the IPBFT consensus process.

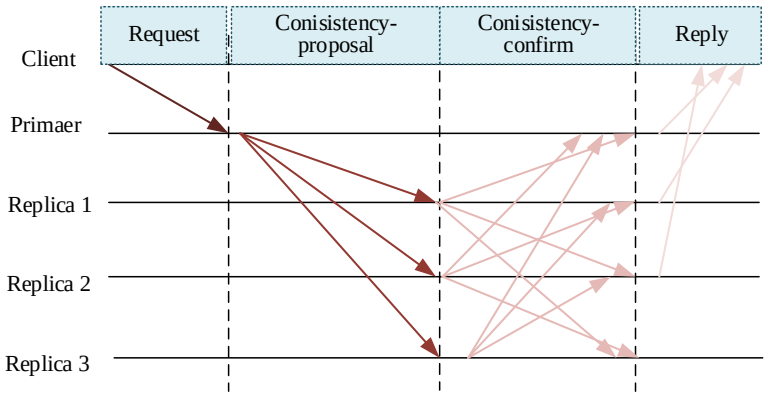


Figure 1: IPBFT consensus process

In Figure 1, even if the master node fails, the consensus system can continue to operate through the view switching protocol to prevent errors [27]. In the PBFT algorithm, once the backup node detects a master node failure or malicious behavior, it will broadcast a message and initiate the view switching process. The IPBFT algorithm classifies and processes view switching protocols based on the abnormal type of the master node. The pseudocode for IPBFT view switching protocol is shown in Algorithm 1.

Algorithm 1: View Change Protocol in IPBFT

Input: Timeout threshold T, Current view number v, Master node M
Output: Updated view and Master node
1. if M fails to respond within T then
2. broadcast <ViewChange, v+1> to all nodes
3. collect ViewChange messages from ≥ 2f+1 nodes
4. select next Master = Master() based on highest Si
5. broadcast <NewView, v+1, Master> to all nodes
6. synchronize pending requests from old view
7. execute consensus in new view
8. else
9. continue normal consensus

In the view switching mechanism of IPBFT, the client sets a timeout threshold. If the master node times out, it may be due to system overload, malfunction, network issues, or intentional denial of service. This indicates that there is a problem with the master node. During the view switching process, whether the master node is abnormal is determined through voting. Once the master node times out, the system will enter the new view phase and select a new master node through the Master() algorithm [28]. The pseudocode for the Master() master node election function is shown in Algorithm 2.

Algorithm 2: View Change Protocol in IPBFT

Input: Node set N = {n1, n2, …, nk}, credit values Vn, votes voten
Output: Master node ID
1. for each node ni in N do
2. compute score Si = λ * Vn + μ * (Σ(Vk / |N|) * votek)
3. end for
4. sort nodes by Si in descending order
5. select node with max(Si) as Master
6. broadcast Master(ID) to all backup nodes
7. return Master

The new master node spread a new view message to continue processing unfinished transactions from the previous view. If the backup node believes that the master node is malicious, it will initiate a view switching process, including view switching, view switching-confirmation, and a new view phase. IPBFT classifies master node anomalies based on PBFT, reducing communication in timeout situations and improving efficiency. Figure 2 illustrates the final IPBFT algorithm flow.

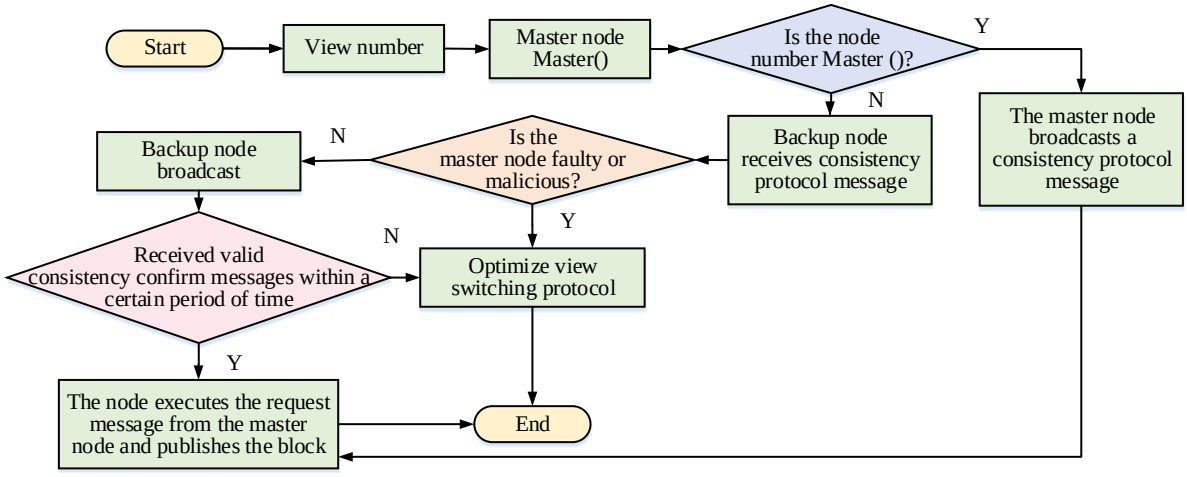


Figure 2: IPBFT algorithm process.

In Figure 2, the view number is initialized. The master node is set to Master() to determine whether the current node is the master node. If it is the master node, protocol messages are broadcasted consistently. Otherwise, the backup node receives consistency protocol messages and broadcasts them. Subsequently, any faults or malicious behavior on the master node are checked. If so, the view switching protocol has been optimized. On the contrary, valid consistency confirmation messages are received within a limited time. If consensus is reached, the node needs to execute the request message from the master node and publish the block. If no consensus is reached, the process is returned to wait and re-execute until the entire consensus process is completed.

3.2 Design of supply chain traceability mechanism based on blockchain and IPBFT consensus algorithm

To ensure the efficient operation of blockchain networks and the smooth execution of consensus algorithms, this study constructs an efficient, secure, and transparent supply chain traceability system. This system combines

the decentralized characteristics of blockchain technology with the fast consistency of the IPBFT consensus algorithm. It can effectively record and verify various transactions and information flows in the supply chain, guaranteeing the authenticity and immutability of data. In the traceability system, from the perspective of system application, the block height increases correspondingly with each data on chain operation. Based on the development platform, the complete process of generating blocks is divided into four stages, including transaction initiation stage, block consensus stage, block verification stage, and block execution stage. Figure 3 illustrates the blockchain application process [29].

As shown in Figure 3, during the transaction initiation phase, a user initiates a transaction and a new message is broadcasted to the blockchain network. After entering the block consensus stage, the system uses the consensus algorithm of the traceability mechanism to consensus the previously initiated transaction information. In the block verification phase, the system schedules transactions and triggers the smart contract after reaching a consensus through the smart contract of the traceability system to verify the information accuracy. Finally, during the block execution phase, the blocks that have reached consensus and passed verification are submitted to the ledger for storage. In a blockchain system using the IPBFT algorithm, the master node is responsible for generating blocks and does not rely on incentive mechanisms. Therefore, the mining reward is not included in the block data. The system block only needs to contain the parent node, digest, and basic information, which not only reduces the storage requirements of the block, but also lowers the data consumption during transmission. On this basis, to enhance the application effect of blockchain in supply chain traceability, a five layer model is designed, including data layer, network layer, consensus layer, contract layer, and view layer. The model adopts a consortium chain form. The consensus algorithm is docked between the data layer and the contract layer. Figure 4 illustrates the architecture of the supply chain traceability system.

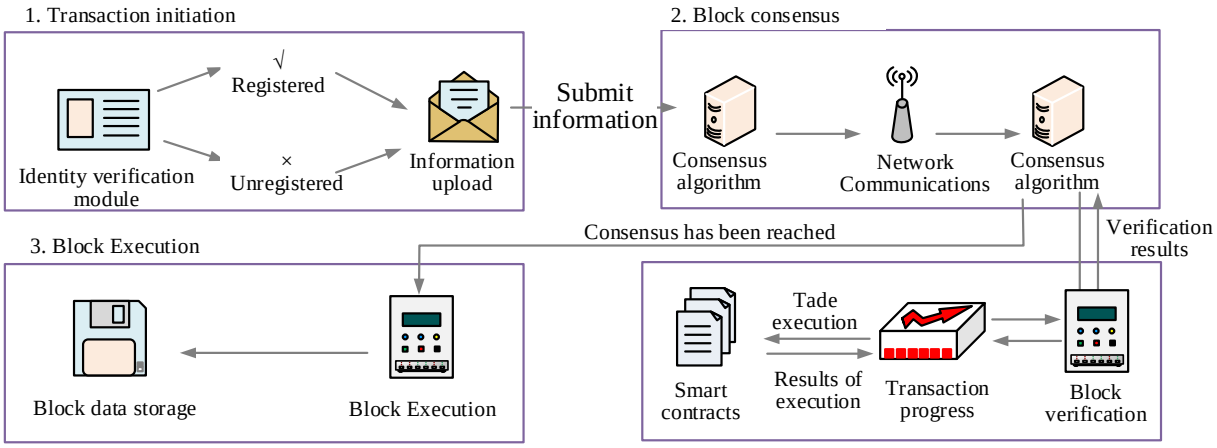


Figure 3: Blockchain application process

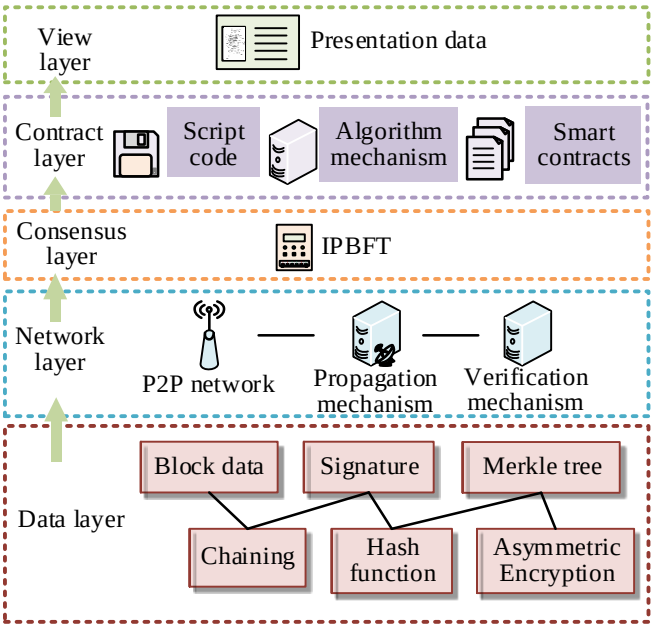


Figure 4: Architecture of supply chain traceability system

In Figure 4, the data layer uses Merkle trees and blockchain to store supply chain data, ensuring data security through digital signatures, hash functions, and asymmetric encryption [30]. A P2P distributed network is established at the network layer, responsible for node communication and block verification. The consensus layer adopts the IPBFT algorithm, which optimizes the consensus efficiency of traditional PoW and PoS algorithms [31, 32]. The contract layer deploys smart contracts for manufacturers, distributors, and retailers, recording digital signatures during transaction execution to achieve full traceability. The view layer provides a user interaction interface through a Graphical User Interface (GUI) [33]. The functions of smart contracts are mainly used for data processing, verification, and permission management, and do not directly participate in the execution of consensus logic. The IPBFT consensus process runs entirely within the consensus layer, with external engines implementing node communication, credit evaluation, and view switching operations. Figure 5 illustrates the final supply chain traceability system based on blockchain and IPBFT consensus algorithm.

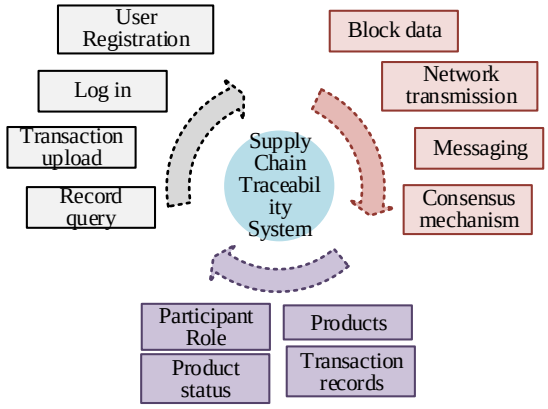


Figure 5: Supply chain traceability system

The system in Figure 5 consists of three core components: user interface module, blockchain coordination module, and smart contract processing module. The blockchain coordination module mainly handles information exchange and data upload between nodes. The underlying architecture of the system employs the Ethereum blockchain platform, deploying blockchain nodes with smart contract support on the Ethereum platform to build a decentralized network environment. The Ethereum client and IPBFT consensus module are integrated to ensure compatibility with existing blockchain architectures [34]. IPBFT adopts three-stage message passing to ensure block finality, and utilizes Ethereum smart contracts to manage node voting and verification processes, recording the entire consensus process. This system combines dynamic node management to maintain high fault tolerance in the event of network anomalies or attacks, while suppressing malicious behavior and incentivizing node participation through reward and punishment mechanisms. The system uses smart contracts to implement tamper proof contract publishing and role behavior constraints, reaching consensus through IPBFT. The user interaction module supports role registration, login, and transaction record upload queries. Among the three types of participants: manufacturers, distributors, and end-users, manufacturers and distributors need to undergo registration and review before uploading transactions. Manufacturer data is directly uploaded to the chain, and distributor data needs to be verified before uploading. All consensus data is stored in a database and queried through an open Application Programming Interface (API) [35]. Users can obtain transaction history sorted by time through the product number, or view detailed information in the blockchain browser through the product specific contract address.

4 Results

4.1 IPBFT performance testing

To ensure the reproducibility of the experiment and comprehensively verify the robustness and anti attack performance of the proposed IPBFT algorithm, a controlled simulation environment was constructed. The blockchain network had a total of 30 nodes, of which 24 were normal nodes and 6 were Byzantine nodes, accounting for approximately 20%. The behavior of Byzantine nodes was simulated in three ways: One was message tampering, where nodes broadcast inconsistent block hashes or transaction results to different nodes to disrupt the consensus process. The second was packet loss behavior, where nodes selectively discarded some consensus messages, thereby increasing communication latency and reducing throughput. The third was latency injection, where nodes actively introduced the random communication latency to simulate unstable scenarios such as network congestion or denial of service attacks. The experimental simulation was based on the Ethereum platform and integrated with a custom IPBFT consensus engine. It was developed using Python 3.10 and PyTorch 2.1 language environments, and the underlying communication framework employed the Go Ethereum client. The smart contract module was deployed in a private Ethereum testing network, written in Solidity language, responsible for transaction recording and data verification. The consensus layer and message scheduling module ran in a Python parallel simulator to achieve synchronization and message interaction between nodes. To avoid accidental effects, each experiment was independently run 10 times under the same system configuration, and the resulting throughput, latency, response time, and communication volume were taken as arithmetic averages. The experimental running hardware environment is an Intel Core i5-9400F processor, 8GB DDR4 memory, and Ubuntu 16.04 operating system, which is consistent with the typical blockchain performance evaluation environment. All performance test results are based on independently running 10 simulations with the same system configuration, and the data obtained is the average of multiple experiments to eliminate possible random fluctuations in a single run. The variance of the experimental results is controlled within ± 3.5%, indicating that the system has good stability and repeatability under multiple rounds of testing. The

Byzantine fault testing is conducted to evaluate the performance changes under various conditions. The proposed IPBFT is compared and tested with GC-PBFT and LC-PBFT. When there are no node failures or attacks from Byzantine malicious nodes, the response speed of these three methods to Invoke and Query operations is shown in Figure 6.

Figures 6 (a) and 6 (b) show the response speed changes of different algorithms in Invoke and Query operations, respectively. Overall, the data throughput continuously increased with the increase of operation requests. In Figure 6 (a), IPBFT always had the highest response speed as the request speed increased, reaching a peak of 438 at 400TPS. The response speed of GC-PBFT was lower than that of IPBFT, but it performed better than that of LC-PBFT. LC-PBFT had the lowest response speed among these three algorithms, indicating relatively poor performance. In Figure 6 (b), IPBFT exhibited the highest response speed at all request speeds. The response speed of LC-PBFT was still the lowest, indicating its lowest efficiency in processing requests. The dynamic weighted credit model and optimized view switching protocol in IPBFT reduce the computational complexity in the consensus process, enabling more reliable nodes to process requests more efficiently. Subsequently, whether there are random node failures and Byzantine malicious attacks is experimentally evaluated. The data throughput of Invoke operation is used as a metric, as shown in Figure 7.

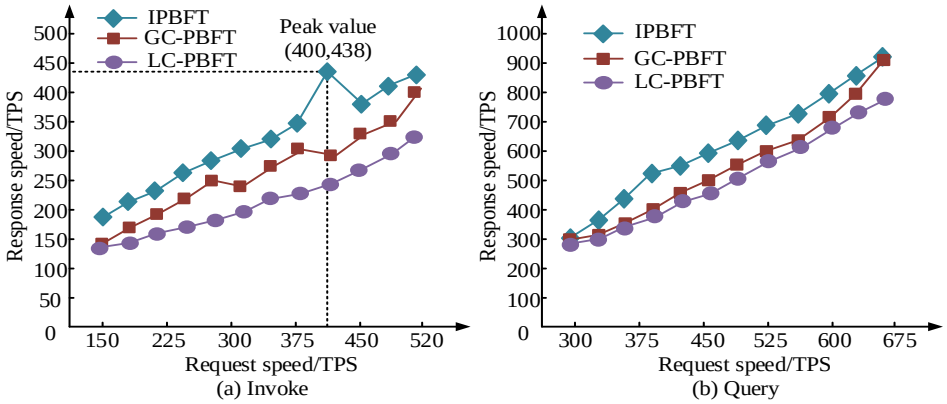


Figure 6: The response speed of three algorithms to different operations

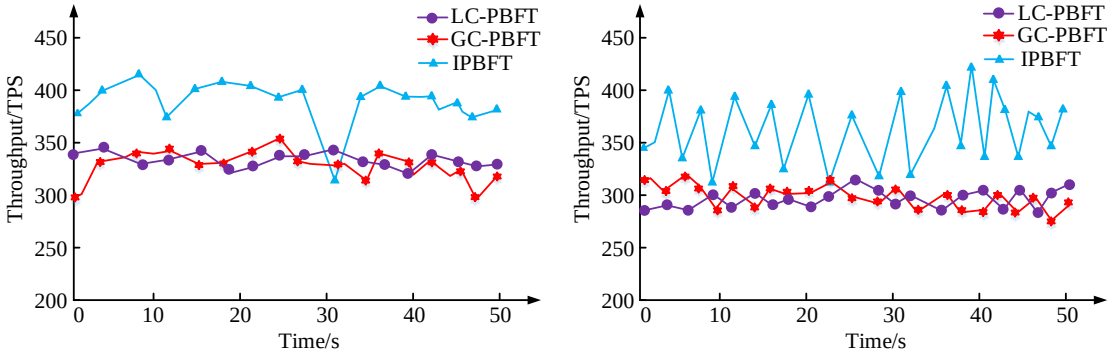
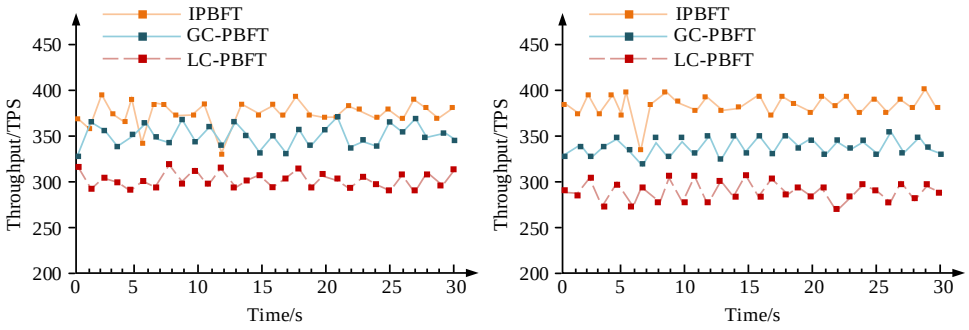


Figure 7: Consensus efficiency changes over time under random faults

Figure 7 (a) shows the performance of throughput over time for LC-PBFT, GC-PBFT, and IPBFT under low failure probability conditions. The throughput of the IPBFT algorithm remains relatively stable, and its overall performance is better than LC-PBFT and GC-PBFT, but it does not show whether it is affected by node failures. After about 30s, the IPBFT algorithm showed that the consensus speed of the system slowed down due to node failures. Even after the speed decreased, the instantaneous processing capability of the system remained at normal operating efficiency, and the stability of the system was not affected. Figure 7 (b) shows the variation of data throughput over time for three algorithms under high failure probability. As the probability of failure continued to increase, the throughput of LC-PBFT and GC-PBFT remained at a relatively high level with relatively small fluctuations. The impact of IPBFT on data transmission speed was significant. Although this algorithm consumes a lot of resources in node replacement and view switching, it can cause performance fluctuations within a certain range. Overall, the efficiency of the IPBFT algorithm is still superior to the other two comparison algorithms, further demonstrating the stability advantage of the dynamic weighting mechanism and fault-tolerant protocol in fault prone environments. Finally, no node failures and continuous attacks from Byzantine malicious nodes are analyzed, and the Invoke operation is also selected. Figure 8 illustrates the data throughput results.



(a) Low probability of malicious attacks (b) High probability of malicious attacks

Figure 8: Consensus efficiency changes over time under random Byzantine attacks

Figure 8 illustrates consensus efficiency under random Byzantine attacks. In Figure 8(a), under low attack probability, due to malicious node replacement, the efficiency of IPBFT has doubled, but it quickly recovered, while other algorithms have not changed, indicating that they cannot detect attacks. In Figure 8(b), as attack probability increases, LC-PBFT and GC-PBFT experience severe efficiency decline and fluctuation, whereas IPBFT maintains higher and more stable throughput after brief interference. The results indicate that the dynamic credit model enables IPBFT to identify and isolate malicious nodes, minimizing the impact of attacks. The total communication volume of different algorithms in a single transaction is tested, as shown in Figure 9.

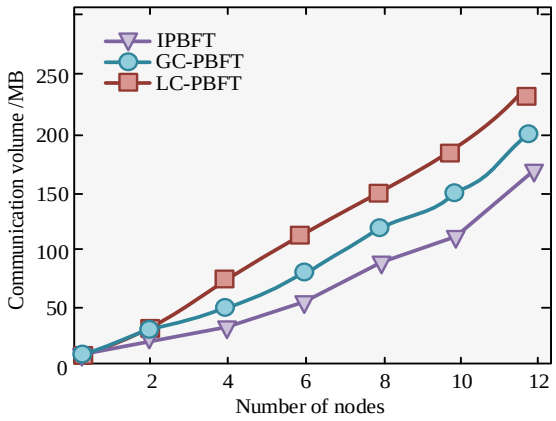


Figure 9: The total communication volume of different algorithms in a single transaction.

In Figure 9, overall, with the increase of nodes, the communication volume of the three methods showed an upward trend. The growth of communication volume was roughly linear, but the growth rate varied among different algorithms. As the node increased, the communication volume of LC-PBFT also increased, which may affect the overall communication efficiency. The communication volume of GC-PBFT was between IPBFT and LC-PBFT, and the communication overhead was higher than that of IPBFT. As the number of nodes increases, IPBFT shows slower growth in communication volume, demonstrating better scalability and communication efficiency. The communication volume of IPBFT is the lowest among these three algorithms. This indicates that IPBFT optimizes communication overhead by reducing redundant interactions in the consensus process. To verify the performance advantages of the IPBFT view switching mechanism, four algorithms, PBFT, GC-PBFT, LC-PBFT, and IPBFT, were compared in terms of view change time under different failure probabilities, as shown in Table 2.

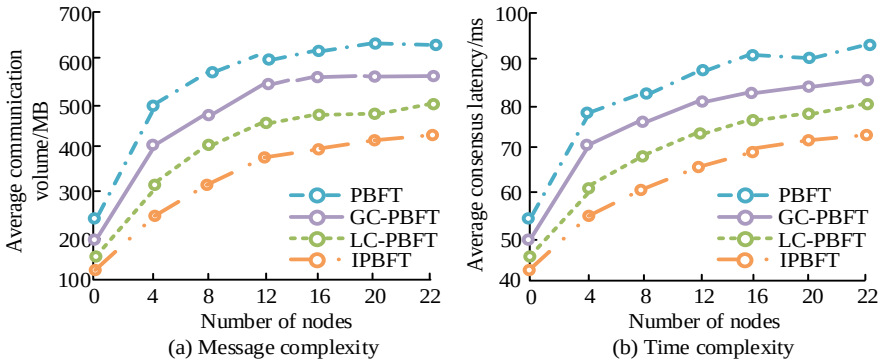
Table 2: Performance under view change time under different failure probabilities.

Scenario	Indicator	PBFT	GC-PBFT	LC-PBFT	IPBFT
Low-failure rate	Average time consumption/s	1.82	1.36	1.22	0.97
	Transaction throughput rate/TPS	365	412	398	438
	Average transaction latency/ms	145	132	128	115
High-failure rate	Average time consumption/s	3.45	2.91	2.63	1.88
	Transaction throughput rate/TPS	322	365	348	403
	Average transaction latency/ms	192	171	163	139

Table 2 showed that IPBFT outperformed PBFT, GC-PBFT, and LC-PBFT in both low- and high-failure environments, with lower time consumption, higher throughput, and shorter latency, demonstrating superior stability and robustness. Under low-failure rates, the average latency of PBFT was 1.82 seconds, 365 TPS, and 145 ms. GC-PBFT and LC-PBFT improved slightly through group validation and hierarchical topology. The average time of IPBFT was 0.97 seconds, which was 46.7% lower than that of PBFT, with a 20.7% reduction in latency and a 20% increase in throughput, indicating higher node collaboration efficiency. At high-failure rates, the time of PBFT increased to 3.45 seconds and the throughput decreased to 322 TPS, while IPBFT remained at 1.88 seconds and 403 TPS. The view switching efficiency increased by 45.5% and the latency decreased by 27.6%.

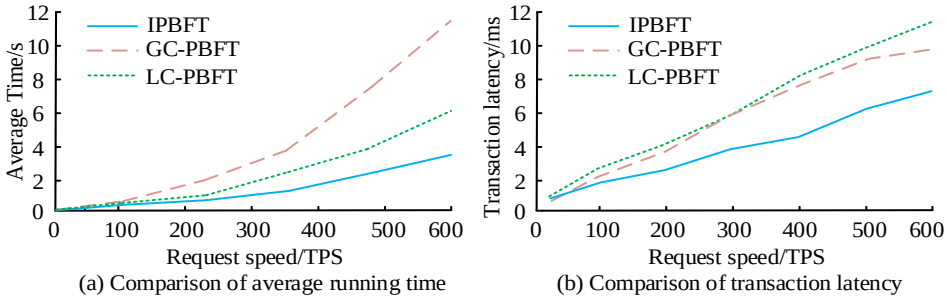
This confirms that IPBFT has strong fault tolerance and fast recovery capability through local broadcasting and isolation of low reputation nodes. Figure 10 presents the changes in message complexity for different algorithms.

In Figure 10 (a), as the number of nodes increases, the communication volume of each algorithm also increases, but the growth patterns differ significantly. The communication volume of PBFT increased from 250 MB to 640 MB. GC-PBFT used a packet broadcast mechanism, resulting in a final communication volume of approximately 550 MB. LC-PBFT adopted a hierarchical communication structure, further reducing the communication volume to 490 MB. In contrast, IPBFT had the slowest growth in communication volume, only increasing from 120 MB to 410 MB, showing an approximately linear growth trend. IPBFT effectively reduces message exchange redundancy through reputation filtering and local broadcast mechanism, and can maintain low communication load even when expanding in scale. In Figure 10 (b), the latency of PBFT increased from 55 ms at 4 nodes to 94 s at 22 nodes, indicating that its multi-node performance is significantly constrained by communication and synchronization. GC-PBFT and LC-PBFT achieved hierarchical and parallel optimization, with final latency controlled at 83 s and 79 s, respectively. IPBFT performed the best, with an average latency of only 71 s and a smooth growth curve, indicating that its dynamic reputation mechanism and asynchronous view switching strategy effectively alleviated the synchronization pressure caused by node expansion.



(a) Message complexity (b) Time complexity

Figure 10: Changes in message complexity of different algorithms.



(a) Comparison of average running time (b) Comparison of transaction latency

Figure 11: System average running time and time latency results.

4.2 Effect analysis of supply chain traceability system based on blockchain and IPBFT consensus algorithm

The study evaluates the efficiency of the designed system through actual operational testing to validate the practicality of the traceability system. Firstly, the blockchain technology is deployed on the virtual machine. The blockchain network is launched to conduct a series of tests on system runtime, transaction latency, and processing capabilities. After multiple tests, the system latency results are shown in Figure 11.

Figure 11 (a) showed that the average running time of IPBFT slowly increased with the increase of request speed, maintaining good efficiency, while GC-PBFT showed the fastest performance improvement above 300 TPS, with a sharp decline. Figure 11 (b) showed that the transaction latency of IPBFT was consistently lower than that of GC-PBFT and LC-PBFT, steadily but moderately increasing with load. In contrast, the latency of GC-PBFT increased rapidly above 400 TPS, while the latency of LC-PBFT was slightly lower but still high under heavy loads. Overall, IPBFT effectively reduces running time and latency under high loads by prioritizing reliable nodes through its credit model and minimizing latency during node changes via the view-switching protocol. Figure 12 displays the resource consumption of the platform in processing transactions under both high and low probability Byzantine malicious attacks.

Figure 12 (a) displays the CPU usage of the system based on different algorithms during operation. The CPU usage of GC-PBFT rapidly increased to nearly 80% in the first 10s, and then slowly increased to about 85%. The algorithm had a high demand for CPU resources. The CPU usage of LC-PBFT sharply increased to around 70% in the first 10s and eventually stabilized at around 75%. The CPU

usage of IPBFT rapidly increased to about 60% in the first 10s and gradually stabilized, eventually stabilizing at around 60%. The low CPU usage of IPBFT indicates that the algorithm requires less CPU resources. Figure 12 (b) shows the bandwidth utilization of supply chain traceability systems based on different algorithms during operation. The bandwidth utilization rates of IPBFT, GC-PBFT, and LC-PBFT were 86.34%, 75.45%, and 56.78%, respectively. The bandwidth utilization of IPBFT gradually decreased with the increase of the node, but still maintained a high level and performed well in handling large amounts of data transmission. To further verify the applicability and comprehensive performance advantages of the proposed IPBFT algorithm under different blockchain architectures, RGBBFT and Polygon blockchain-based strategies were further introduced in the study [36]. Taking speed, security, scalability, and cost as indicators, the results are shown in Table 3.

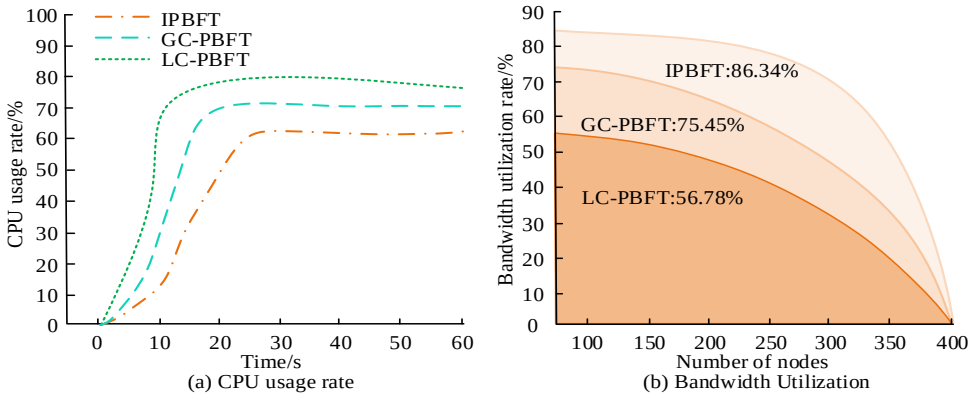


Figure 12: Resource consumption results of different algorithms.

Table 3: Comprehensive performance comparison.

Algorithm	Throughput (TPS)	Byzantine fault tolerance (%)	Node capacity (piece)	Average cost (RMB)
PBFT	365	90.2	35	72,500
GC-PBFT	412	92.7	52	68,400
LC-PBFT	398	93.1	55	64,800
IPBFT	438	95.1	77	60,300
RGBBFT	421	94.3	64	62,800
Polygon blockchain	452	93.8	127	55,600

Table 3 showed that traditional PBFT faces a trade-off between speed, security, and scalability, with node capacity limited to 35 due to high communication complexity. GC-PBFT and LC-PBFT improved speed and stability through grouping and hierarchy but remained constrained in scalability. The proposed IPBFT, enhanced with dynamic reputation evaluation and asynchronous view switching, achieved 438 TPS throughput, 95.1% fault tolerance, scalability up to 77 nodes, and 16.9% lower maintenance costs, offering the most balanced performance. RGBBFT excels in trust and security through reputation governance, while Polygon's two-layer sidechain design provides excellent scalability and energy efficiency, supporting 127 nodes at the lowest cost, making it an ideal choice for large-scale, high concurrency supply chain networks. Finally, a simulation platform is established to apply the proposed supply chain traceability mechanism to a certain aquatic product supply chain. The cost-benefit analysis is compared with the supply chain traceability system that does not use the proposed mechanism. Table 4 displays the results.

Table 4 showed that both systems require an initial investment of 300,000 RMB. However, traditional systems had poor scalability, with annual maintenance costs of about 80,000 RMB and upgrade costs of 140,000 RMB. The proposed system, with higher automation and scalability, reduced these to about 60,000 and 120,000 RMB, respectively. Its product recall rate was only 2.1%, compared to 5.2% for traditional systems, greatly improving reliability and economic efficiency. The product data in Table 4 was based on simulated estimates, not real enterprise data. Cost is calculated theoretically based on market prices and experimental assumptions, while recall rate is determined based on the proportion of simulation errors in the traceability process.

Table 4: Cost benefit comparison of supply chain.

Traceability system	Initial investment cost (RMB)	Operation and maintenance costs (RMB/year)	Upgrade and expansion costs (RMB)	Product recall rate (%)
Traditional traceability system	300,000	80,000	140,000	5.2
The designed traceability system	300,000	60,000	120,000	2.1

5 Discussion

A blockchain-based supply chain traceability mechanism with an improved PBFT consensus is proposed to enhance product traceability and accountability. By introducing a dynamic weighted credit and voting model, it overcomes the performance and security limitations of traditional PBFT, resolving bottlenecks under high-frequency transactions, node failures, and Byzantine attacks. Experiments show that the IPBFT algorithm achieves optimal response and throughput at varying request rates. Under low-probability Byzantine attacks, despite brief efficiency drops, the system quickly restores stability through node replacement. As node scale increases, IPBFT shows the slowest communication growth and lower latency compared to other algorithms, demonstrating strong scalability. Even under high-probability attacks, system efficiency rapidly recovers after isolating malicious nodes. CPU utilization stabilizes around 60%, and bandwidth efficiency remains high as nodes increase. Economically, the system has maintained low annual maintenance and expansion costs, with a product recall rate of 2.1%, far lower than the 5.2% of traditional systems, improving reliability and cost-effectiveness. Compared to the group reputation clustering of GC-PBFT and the hierarchical consensus model of LC-PBFT, the innovation of IPBFT mainly lies in two aspects: One is the bidirectional dynamic reputation update mechanism. The credit attenuation parameter of IPBFT is not a preset constant, but varies proportionally with the transaction density and anomaly rate of nodes. When a node is in a high reputation state for a long time, the latency rate slows down to maintain system stability. Under continuous low reputation behavior, the attenuation coefficient rapidly increases to accelerate the isolation of malicious nodes. The second is the collaborative protocol for switching partial views. Traditional PBFT requires network wide broadcast triggering switching when the master node is abnormal, while IPBFT achieves local autonomous switching through reputation weighted selection, improving scalability and stability in large-scale node environments.

Although the IPBFT-based blockchain traceability system performs well under controlled attack scenarios, several challenges remain in real-world applications. Firstly, the system's scalability requires further verification. Current experiments involve only 30 nodes. Expanding to cross-regional networks with hundreds or thousands of nodes may sharply increase communication complexity and synchronization latency, necessitating optimization via message compression or hierarchical consensus. Secondly, integration with existing enterprise systems incurs high costs due to interface development and protocol adaptation, while regional differences in data privacy, regulation, and authentication hinder large-scale deployment. Finally, current security tests cover only moderate Byzantine and denial-of-service attacks, without addressing extreme cases such as multi-node collusion or network intrusions. Future work should strengthen robustness testing under such conditions. In terms of cost-effectiveness, although the aquatic product supply chain case shows that the system effectively reduces recall rates, lowers maintenance costs, and improves transparency, its cross-industry applicability still requires verification. Industries such as pharmaceuticals and food demand higher real-time traceability, compliance, and privacy protection, while electronic manufacturing is more sensitive to scalability and consensus latency. Thus, despite the system's general advantages in transparency, efficiency, and security, variations in regulation, data scale, and network complexity may affect deployment parameters and economic benefits. Future work should extend cost-benefit evaluations across industries and optimize adaptive designs for different application contexts.

6 Conclusion

The proposed IPBFT algorithm has shown significant performance advantages in multi-scenario experiments. Compared with GC-PBFT and LC-PBFT, IPBFT has a throughput increase of about 7%-10%, an average latency reduction of about 12%-18%, and a system availability improvement of over 15% under high attack probability. The dynamic weighted credit model quickly identifies and isolates malicious nodes by real-time evaluation of node behavior and feedback consistency. The parallel optimization of the view switching mechanism effectively reduces communication complexity and improves the scalability of the system under high concurrency conditions. Overall, IPBFT not only achieves efficiency and security optimization of traditional PBFT at the algorithm level, but also demonstrates strong universality and scalability potential in industrial applications. Its high concurrency, low latency, and robustness make it suitable for deployment in multiple industry scenarios such as food safety supervision, pharmaceutical anti-counterfeiting tracking, and high-value product authentication. It can support trusted traceability and cross-domain

collaboration throughout the entire supply chain lifecycle, further verifying the applicability and promotional value of the proposed model in complex real-world systems. Future work will focus on cross-industry deployment and validation in large-scale node environments, further improving the adaptive scheduling and economic incentive mechanisms of algorithms to achieve a scalable and trustworthy blockchain traceability system.

Although the blockchain traceability system based on IPBFT has good performance in controlled attack scenarios, there are still several limitations and challenges that need further research in practical applications. Firstly, the scalability of the system needs to be verified. The current experiment is only completed at a scale of 30 nodes. If it is expanded to a cross regional supply chain network with hundreds or even thousands of nodes, the communication complexity and synchronization latency may significantly increase, and optimization through message compression or hierarchical consensus mechanisms is needed. Secondly, the integration cost between the system and the existing information system of the enterprise is high, requiring additional development of interfaces and adaptation protocols. Meanwhile, policy differences in data privacy, regulatory compliance, and digital authentication across regions also pose challenges for large-scale promotion. Finally, the existing security verification only covers moderate intensity Byzantine attacks and denial of service attacks, and has not yet simulated extreme scenarios such as multi-node collaborative attacks or underlying network intrusions. In the future, it is necessary to further strengthen the security robustness testing of the system.

Acknowledgement

This study was supported by the 2024 National Social Science Foundation Project "Policy Optimization Research on Digital Rural Development and Prevention of Large-Scale Poverty Relapse in Ethnic Minority Areas of Southwest China" (Project Number: 24XMZ032), and 2025 Annual Project of Philosophy and Social Sciences Research in the Guangxi Zhuang Autonomous Region Project, Project Title: Research on the Empowerment Mechanism and Path of Market-Oriented Allocation of Data Elements for the New Quality Productive Forces of Enterprises in Guangxi (Project Number: 25JYF193).

References

[1] L. Dong, P. Jiang, and F. Xu. Impact of traceability technology adoption in food supply chain networks. Management Science, 69(3), 1518-1535, March, 2023, DOI: 10.1287/mnsc.2022.4440.

[2] G. M. Razak, L. C. Hendry, and M. Stevenson. Supply chain traceability: A review of the benefits and its relationship with supply chain resilience. Production Planning & Control, 34(11), 1114-1134, August, 2023, DOI: 10.1080/09537287.2021.1983661.

[3] L. Chu. Optimization method of fresh agricultural products cross-border e-commerce supply chain based on blockchain technology. Pakistan Journal of Agricultural Sciences, 60(2), 415-423, June, 2023, DOI: 10.21162/PAKJAS/23.140.

[4] X. Zhang and L. Ling. A review of blockchain solutions in supply chain traceability. Tsinghua Science and Technology, 28(3), 500-510, June, 2022, DOI: 10.26599/TST.2022.9010030.

[5] H. Luo, X. Yang, H. Yu, G. Sun, B. Lei, and M. Guizani. Performance analysis and comparison of non-ideal wireless PBFT and RAFT consensus networks in 6G communications. IEEE Internet of Things Journal, 1(6), 9752-9765, March, 2023, DOI: 10.1109/JIOT.2023.3323492.

[6] H. Wu, S. Jiang, and J. Cao. High-efficiency blockchain-based supply chain traceability. IEEE Transactions on Intelligent Transportation Systems, 24(4), 3748-3758, April, 2023, DOI: 10.1109/TITS.2022.3205445.

[7] T. Chen, Y. Li, and F. Xu. Traceability strategy choice in competing supply chains based on blockchain technology. International Transactions in Operational Research, 31(6), 3873-3904, November, 2024, DOI: 10.1111/itor.13332.

[8] A. S. Berkani, H. Moumen, S. Benharzallah, S. Yahiaoui, and A. Bounceur. Blockchain use cases in the sports industry: a systematic review. International Journal of Networked and Distributed Computing, 12(1), 17–40, June, 2024. DOI: 10.1007/s44227-024-00022-3.

[9] J. Tan, W. P. Wong, C. K. Tan, S. Jomthanachai, and C. P. Lim. Blockchain-based Logistics 4.0: enhancing performance of logistics service providers. Asia Pacific Journal of Marketing and Logistics, 36(6), 1442–1463, June, 2024. DOI: 10.1108/APJML-07-2023-0650.

[10] X. Feng, K. Cui, L. Wang, Z. Liu, and J. Ma. PBAG: A privacy-preserving blockchain-based authentication protocol with global-updated commitment in IoVs. IEEE Transactions on Intelligent Transportation Systems, 25(10), 13524-13545, October, 2024, DOI: 10.1109/TITS.2024.3399200.

[11] Y. He, M. Luo, B. Wu, L. Sun, Y. Wu, Z. Liu, and K. Xiao. A game theory-based incentive mechanism for collaborative security of federated learning in energy blockchain environment. IEEE Internet of Things Journal, 10(24), 21294-21308, December, 2023, DOI: 10.1109/JIOT.2023.3282732.

[12] Y. P. Zhou, X. J. Zhao, and L. Sun. Research on traceability strategy of food supply chain considering delay effect. Journal of Food Science, 87(11), 4831-4838, November, 2022, DOI: 10.1111/1750-3841.16278.

[13] X. Wu, H. Ling, H. Liu, and F. Yu. A privacy-preserving and efficient byzantine consensus through multi-signature with ring. Peer-to-Peer Networking and Applications, 15(3), 1669-1684, May, 2022, DOI: 10.1007/s12083-022-01317-4.

[14] H. Qin, Y. Cheng, X. Ma, F. Li, and J. Abawajy. Weighted Byzantine Fault Tolerance consensus algorithm for enhancing consortium blockchain efficiency and security. Journal of King Saud University-Computer and Information Sciences,

34(10), 8370-8379, November, 2022, DOI: 10.1016/j.jksuci.2022.08.017.

[15] C. Li, W. Qiu, X. Li, C. Liu and Z. Zheng. A dynamic adaptive framework for practical byzantine fault tolerance consensus protocol in the internet of things. IEEE Transactions on Computers, 73(7), 1669-1682, July, 2024, DOI: 10.1109/TC.2024.3377921.

[16] F. Q. Ma, Q. L. Li, Y. H. Liu, and Y. X. Chang. Stochastic performance modeling for practical byzantine fault tolerance consensus in the blockchain. Peer-to-Peer Networking and Applications, 15(6), 2516-2528, November, 2022, DOI: 10.1007/s12083-022-01380-x.

[17] Z. F. Wang, S. Q. Liu, P. Wang, et al. BW-PBFT: Practical byzantine fault tolerance consensus algorithm based on credit bidirectionally waning. Peer-to-Peer Networking and Applications, 16(6), 2915-2928, November, 2023, DOI: 10.1007/s12083-023-01566-x.

[18] C. Li, J. Zhang, and X. Yang. Scalable blockchain storage mechanism based on two-layer structure and improved distributed consensus. The Journal of Supercomputing, 78(4), 4850-4881, March, 2022, DOI: 10.1007/s11227-021-04061-3.

[19] F. Tang, T. Xu, J. Peng, and N. Gan. TP-PBFT: A scalable PBFT based on threshold proxy signature for IoT-blockchain applications. IEEE Internet of Things Journal, 11(9), 15434-15449, September, 2023, DOI: 10.1109/JIOT.2023.3347232.

[20] Y. Li, H. Huang, A. Lan, and Z. Huang. A practical byzantine fault tolerance improvement algorithm based on credit grouping-classification. The Journal of Supercomputing, 80(14), 20270-20301, September, 2024, DOI: 10.1007/s11227-024-06199-2.

[21] J. Li, L. Cao, S. Zhao, Wan, and J. Bai. LC-PBFT: Layered cross-chain consensus algorithm based on forest topology. The Journal of Supercomputing, 80(12), 17849-17873, August, 2024, DOI: 10.1007/s11227-024-06122-9.

[22] X. Zhu, X. Hu, W. Zhu. RGPBFT: A reputation-based PBFT algorithm with node grouping strategy. Arabian Journal for Science and Engineering, 50(15): 11837-11850, August, 2025. DOI: 10.1007/s13369-024-09614-1.

[23] H. Qushtom, J. Mišić, V. B. Mišić, and Chang. A two-stage PBFT architecture with trust and reward incentive mechanism. IEEE Internet of Things Journal, 10(13), 11440-11452, July, 2023, DOI: 10.1109/JIOT.2023.3243189.

[24] S. Alshihri and S. Park, "A Decentralized Lightweight Blockchain Nodes Architecture Based on a Secure OpenFlow Protocol Controller Channel," Tehnički vjesnik, vol. 30, no. 1, pp. 114–121, February 2023. DOI: 10.17559/TV-20220427051644.

[25] G. Rigatos, M. Abbaszadeh, B. Sari, P. Siano, G. Cuccurullo, and F. Zouari. Nonlinear optimal control for a gas compressor driven by an induction motor. Results in Control and Optimization, 11, 100226, June, 2023. DOI: 10.1016/j.rico.2023.100225.

[26] A. Boulkroune, F. Zouari, and A. Boubellouta. Adaptive fuzzy control for practical fixed-time synchronization of fractional-order chaotic systems. Journal of Vibration and Control, February, 2025, OnlineFirst, DOI: 10.1177/10775463251320258.

[27] J. Li, X. Li, H. Zhao, B. Yu, T. Zhou, H. Cheng, and N. Sheng. MANDALA: A scalable blockchain model with mesh-and-spoke network and H-PBFT consensus algorithm. Peer-to-Peer Networking and Applications, 16(1), 226-244, January, 2023, DOI: 10.1007/s12083-022-01373-w.

[28] D. Zhang, N. H. A. Wahab, and A. W. M. Zin. Optimizing blockchain consensus: Incorporating trust value in the practical Byzantine fault tolerance algorithm with Boneh-Lynn-Shacham aggregate signature. Baghdad Science Journal, 21(2), 0633-0633, February, 2024, DOI: 10.21123/bsj.2024.9735.

[29] M. Mohit, S. Kaur, and M. Singh. Design and implementation of transaction privacy by virtue of ownership and traceability in blockchain based supply chain. Cluster Computing, 25(3), 2223-2240, June, 2022, DOI: 10.1007/s10586-021-03425-x.

[30] J. Zhang, P. Zhou, J. Wang, O. Alfarraj, S. Singh, and M. Zhu. A novel high-efficiency transaction verification scheme for blockchain systems. CMES-Computer Modeling in Engineering & Sciences, 139(2), 60-66, February, 2024, DOI: 10.32604/cmes.2023.044418.

[31] G. A. F. Rebello, G. F. Camilo, L. C. B. Guimaraes, L. A. C. de Souza, G. A. Thomaz, and O. C. M. B. Duarte. A security and performance analysis of proof-based consensus protocols. Annals of Telecommunications, 77(7-8), 517-537, August, 2022, DOI: 10.1007/s12243-021-00896-2.

[32] C. Mueller-Bloch, J. V. Andersen, J. Spasovski, and J. Hahn. Understanding decentralization of decision-making power in proof-of-stake blockchains: an agent-based simulation approach. European Journal of Information Systems, 33(3), 267-286, May, 2024, DOI: 10.1080/0960085X.2022.2125840.

[33] Q. Fan, Y. Xin, B. Jia, Y. Zhang, and P. Wang. COBATS: A novel consortium blockchain-based trust model for data sharing in vehicular networks. IEEE Transactions on Intelligent Transportation Systems, 24(11), 12255–12271, November, 2023. DOI: 10.1109/TITS.2023.3286432.

[34] A. Purusottama, Y. Sunitiyoso, and T. M. Simatupang. Exploring the potential of blockchain adoption for promoting value innovation: A case of the halal industry. Business Process Management Journal, 29(7), 2034–2058, October, 2023. DOI: 10.1108/BPMJ-04-2023-0267.

[35] Y. Zhou, X. Gao, and J. Nie. Value of blockchain-enabled supply chain traceability under competition. International Transactions in Operational Research, 31(6), 3669-3703, November, 2024, DOI: 10.1111/itor.13295.

[36] P. Nowvaratkoolchai, N. Thawesaengskulthai, W. Viriyasitavat, and P. Rangsunvigit. Blockchain-based cannabis traceability in supply chain management. International Journal of Advanced Computer Science and Applications, 15(2), 75–85, February, 2024. DOI: 10.14569/ijacsa.2024.0150210.

SGNPP: A Graph Neural Network-Based Framework for Japanese Grammar Modeling and Adaptive Learning Path Optimization

Lingling Zheng
Shanghai Jian Qiao Universty, Pudong, 201306, Shanghai, China
E-mail: lingling_zheng033@hotmail.com

Keywords: Japanese grammar, graph neural network, syntax analysis, adaptive learning, reinforcement learning, curriculum optimization

Received: November 4, 2025

Japanese grammar, with its hierarchical structures, flexible word order, and context-driven usage, presents unique challenges for both computational analysis and personalized language learning. Existing methods, including rule-based parsing and sequence-oriented models, often overlook the interconnectedness among grammatical constructs, leading to fragmented analysis and static, one-size-fits-all learning paths. Traditional approaches frequently struggle to capture long-range dependencies and prerequisite relationships between grammar patterns, limiting their ability to guide effective learning progressions. These shortcomings hinder adaptive curriculum design and individualized progression tracking for learners. To address these issues, this study introduces the Syntax Graph Neural Path Planner (SGNPP), which models Japanese grammar as a structured graph where nodes represent morphemes, bunsetsu units, and grammar patterns. At the same time, the edges encode syntactic and prerequisite relationships. Using graph neural networks, SGNPP performs grammar recognition, error diagnosis, and learning dependency modeling. Reinforcement signals are integrated to optimize personalized learning paths, balancing mastery gain with efficiency and cognitive load. The proposed method enables adaptive sequencing of grammar lessons, real-time feedback, and dynamic instruction adjustment based on learner performance. The experiments show that SGNPP has a grammatical identification accuracy of 92%, outperforming CILS (78%), FIAS (82%), and TTS (86%). A practical learning trajectory study shows that SGNPP has 86% mastery of advanced grammar. Quantitative results show that SGNPP is an effective and customized strategy for learning Japanese and improves computational grammar instruction.

Povzetek: Študija predstavi SGNPP, grafovski pristop k japonski slovnici, ki modelira morfeme/bunsetsu/vzorce in njihove odvisnosti ter z ojačitvenimi signali vodi personalizirane učne poti z diagnozo napak in prilagojenim zaporedjem lekcij.

1 Introduction

Japanese grammar is characterized by hierarchies in sentence structure, a certain fluidity in word order, and a reliance on context [1]. Japanese differs from English and other Indo-European languages in its heavy use of particles, bunsetsu (phrasal units), and an honorific form to create syntactic and semantic structure [2]. Such characteristics make computational grammar analysis problematic and pose additional difficulties in the development of adaptive learning systems [3]. The need to customize the learning path and analyze correct grammar, in light of the increased interest in AI-based language learning services, has now become a significant area of investigation [4].

1.1 Challenges in Japanese grammar learning

The use of non-linear dependencies, implicit subject references, and overlapping grammar structures is among the factors that make it challenging for learners to master Japanese grammar [5]. There is an inefficiency in the implementation of grammar in traditional teaching because grammar sequences are structured in fixed ways, which may not reflect the requisite breadth of concepts [6]. Long-range dependencies that require mentioning topic continuation through clauses also prove hard to capture with conventional computational models, leading to incomplete or erroneous representations of grammar structure [7].

1.2 Existing computational and pedagogical models

Rule-based parsing methods are highly interpretable, but they cannot scale to recognise natural-language use as it occurs in the real world [8], even though sequence-based deep learning models are effective in modeling local patterns [9]. It tends to neglect the graph-like structure of grammatical relations, leading to a disjointed analysis [10]. Pedagogical models, however, typically use fixed learning paths that cannot adapt to individual learners' performance [11].

The novel use of the CILS (Cross-Cultural Intelligent Language Learning System), FIAS (Flanders Interaction

Analysis System), and text-to-speech (TTS) technology enabled the recognition of English-to-Japanese grammar, error detection, and the personalization of learning paths. However, this was later adapted to measure and quantify teacher-student interaction, examining errors in the structure of both oral and written conversations. Designed initially to observe interactions in a classroom setting, the FIAS was later reconfigured to measure teacher-student interaction. The FIAS was effective at measuring communication between a language learner and the instructor. The implications are tied to measuring not only student communicative engagement but also to uncovering grammatical deficiencies, such as incorrect conjugations and overuse of particles. TTS used speech recognition to transcribe and analyze Japanese conversational speech, detecting common grammatical errors. The TTS system began with a design to model human speech using a simple conventional language model. By combining spoken language processing and TTS, the system modeled errors in pronunciation, verb conjugation, and the learner's use of particles. The feedback mechanism worked similarly to an adaptive learning path, adjusting the difficulty of language practice based on errors to maximise overall learning outcomes. The transition of FIAS and TTS to their intended use helped grammar learners detect their errors through personalized pathways, thereby representing formative practices for the study of grammar.

The mentioned limitations suggest that a graph-based subset of Japanese grammar structures can be built to analyze them and support optimization of the progression paths as well [12].

1.3 Contributions

- ✓ Introduced the SGNPP, a novel graph-based framework that models Japanese grammar through interconnected morphemes, bunsetsu units, and grammar pattern dependencies.
- ✓ SGNPP integrates grammar recognition, error diagnosis, and dependency learning using graph neural networks, enabling context-aware analysis and capturing long-range prerequisite relationships within grammar constructs.
- ✓ Incorporated reinforcement learning signals, SGNPP dynamically optimizes personalized learning paths, balancing mastery, efficiency, and cognitive load, thus enabling adaptive curriculum sequencing and real-time learner feedback.

1.4 Problem statement

Even though the studies reviewed support the use of AI, semantic representation, flipped classrooms, multimodal systems, topic maps, speech interfaces, TTS, reinforcement learning, MT, and L2 pedagogy for Japanese language learning, there are crucial issues that need to be addressed. Available models do not necessarily incorporate computational linguistics and pedagogical situations, lack adequate scaling across different learners, and provide insufficient individualization in grammar learning. Moreover, there are not many unified approaches to addressing isolated layers — semantics, speech, or pedagogy — that do not fill the gaps in adaptive, grammar-oriented, and learner-oriented optimization. Table 1 shows the paper summary.

Table 1: Paper summary

Section	Content
Introduction	Background, challenges in Japanese grammar, and motivation.
Problem Statement	List inefficiencies in existing methods.
Related Works (LR)	Summaries of 10 papers (CILS, FIAS, TTS, MT(Machine Translation), flipped classrooms, energy RL (Reinforcement Learning), TTS surveys, etc.).
Proposed Method (SGNPP)	Syntax Graph Neural Path Planner explained, block diagrams, pipeline stages.
Results & Discussion	8 parameters with tables (accuracy, error diagnosis, latency, etc.), comparisons with CILS, FIAS, and TTS.
Conclusion & Future Work	Key contributions, implications for ITS, extensions (cross-language, multimodal).

2 Related works

Peculiarities of the Japanese language and grammar stem from its rather complicated semantics and structure. The current computational and pedagogical models also aim to address these problems, but most lack adaptive sequencing, individually tailored scaffolding, and efficient grammar encoding. The methodical comparison of previous research identifies areas for optimization using intelligent graph models. Table 2 summarizes related work.

Table 2: Related work summary

Reference	Environment	Algorithm/ Method	Grammar Structure Analysis	Learning Path Adaptivity	Personalization	Effectiveness	Results
[13] Semantic Analysis	NLP	Semantic modeling	✓	X	X	Medium	Improved grammar correction; focuses on semantic modeling in NLP tasks.

[14] Flipped Classroom	Pedagogical	Inverted learning design	X	✓	X	High	Demonstrates high effectiveness in improving engagement and learning outcomes in language education
[15] Multimodal AI	AI + Multimedia	Multimodal fusion	✓	X	X	Low	Results show limited application for grammar correction; it focuses more on multimodal interaction.
[16] Topic Maps	Knowledge Systems	Ontology mapping	✓	X	✓	Medium	Results suggest moderate success in mapping grammar dependencies, but limited adaptivity
[17] Speech-based	Speech Tech	Spoken interaction system	X	✓	✓	Medium	Results show moderate success in analyzing spoken interactions for language learning,
[18] TTS	AI + Speech	Text-to-Speech	X	X	✓	Low	TTS is effective for speech synthesis, but results show limited use in grammar error detection.
[19] RL for Grammar	RL	Policy optimizatio n	✓	✓	X	High	RL optimizes grammar learning paths and corrects grammar faults well.
[20] MT-based	MT/NLP	Statistical MT	✓	X	X	Low	Results indicate low success in correcting grammar errors.
[21] L2 Pedagogy	Education Theory	Cognitive models	X	✓	✓	Medium	Results suggest moderate success in improving learner outcomes
[22] Cognitive AI Tutor	AI + EdTech	Adaptive tutor system	✓	✓	✓	High	Adjusting grammatical correction learning paths and engaging students is successful.

The literature survey of ten exemplary approaches reveals that they offer partial solutions, with strengths in semantic analysis, pedagogy, and reinforced learning. Still, there are weaknesses in adaptivity, personalization, and the use of integrated grammar modeling. These results lay the foundation for a new Syntax Graph Neural Path Planner (SGNPP) that can be used to analyze grammars and design optimal, personalized paths.

SGNPP focuses on grammar recognition, mistake identification, and learning path optimization. It lets it overcome CILS, FIAS, TTS, and other model limitations. Grammatical linkages in a structured graph enable SGNPP to detect errors in learner-generated sentences accurately. In contrast, CILS essentially changes the text to reflect the cultural background. A graph neural network lets SGNPP track and optimize learning paths based on grammatical mastery and mistakes. Compare this to FIAS, which analyzes classroom interactions but does not fix grammar. TTS is primarily a speech synthesis model without grammar correction, whereas SGNPP delivers a dynamic, individualized learning experience by incorporating grammar recognition and error type feedback. Learning becomes more individualized and dynamic. The SGNPP optimizes the order of grammar courses and student mental effort via reinforcement learning, unlike the other models. Thus, it is a better, faster, and more scalable approach to adaptive grammar learning.

3 Proposed method

The SGNPP framework comprises structural parsing, a graph neural network, and personalized learning for teaching Japanese grammar. It builds a grammar graph, encodes grammatical context, diagnoses learners' errors, optimizes learning paths, and performs real-time instructional adaptations using an intelligent, data-driven approach to second-language acquisition.

The SGNPP model for TensorFlow is based on input layers absorbing tokenized Japanese texts and grammatical feature vectors. The embedding layers will embed the tokens into dense vector representations using pre-trained embedding for Japanese morphemes and grammar representations. The graph neural network layers after the embedding layers capture the hierarchical and relational links among grammatical node links on the structured grammar graph. The output layers create personalized, next-step learning objectives and prediction of well-formedness based on grammar. Adaptive reinforcement learning signals and cross-entropy loss are used to improve the predictions.

3.1 Grammar graph construction

In Figure 1, the Raw Japanese text is processed into a grammar graph in this module. It starts with morphology, where tokens and part-of-speech tags are attached to the sentences. Bunsetsu chunking arranges words together into phrases and recognizes dependencies. A grammatical detector identifies structure in the grammar through rule-based and ML techniques. The final step is to create a grammar graph with nodes (morphemes, phrases, and grammar rules) and edges (dependencies and syntactic connections), which can be used for further processing and the generation of learning pathways.

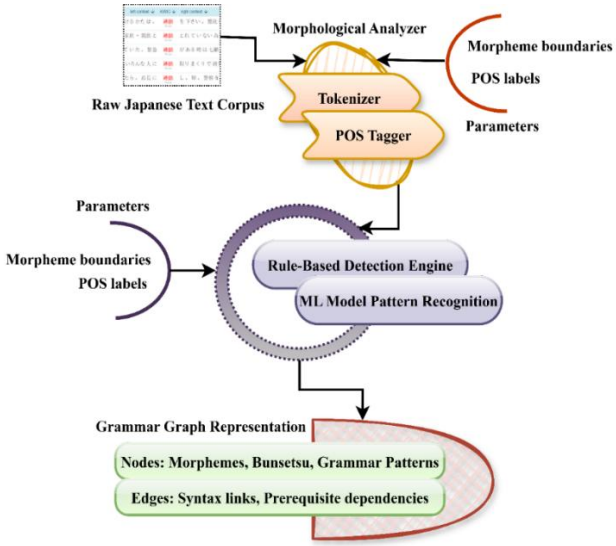


Figure 1: Grammar graph construction

The system can understand the complexity of the Japanese language using the SGNPP model's graph-construction rules for Japanese grammar. The Japanese grammar graph shows how sentences are formed differently from other languages' syntactic graphs. Phrases commonly follow topic-comment patterns, with "息" as a topic marker and subject-object-verb order. Making a graph starts with recognizing morphemes, verbs, particles, bunsetsu (phrase units), etc. Verb conjugations affect sentence structure, and particles connect nouns and verbs. Nodes in the graph represent dependent relationships like this. This architecture's language dependency modeling is crucial. Here, subject-object-verb phrases form the basis for more complicated sentence structures, including relative clauses and conditional forms. Like Japanese grammar, this graph enables learners to grow dynamically, moving on to more complex patterns after learning the basics. The model considers Japanese language nuances by examining connections between grammatical points and their impact on sentence meaning, including particles such as "恍" and "息." SGNPP's context-aware, progression-based learning methodology distinguishes it from classic syntactic models.

Grammar dependencies are modeled by GAT layers, which provide 4-8 attention heads and 128-256 output features. The SGNPP tokenizes Japanese sentences such as (batch_size, sequence_length) and uses a 300-512 pixel embedding layer to produce dense vectors, as mentioned above. Finally, the thick output layers are indicative of both adaptive learning and grammar correction in the construction of dense vectors. Finally, the thick output layers suggest reinforcement signals working towards adaptive path optimization.

Equation 1 updates the grammar node representation for the layer $I_j^{(m+1)}$, collecting contextual embeddings from its syntactic neighbors. The normalization term equalizes the weight given to morphemes, bunsetsu, and grammar-rule nodes.

$$I_j^{(m+1)} = \forall\left(\frac{1}{\sqrt{e_j e_k}} * X^{(m)} * I_k^{(m)} + C^{(m)}\right) \quad (1)$$

Where, $I_j^{(m+1)}$ is the updated value of the feature vector after iteration at step m+1, $e_j e_k$ be the scaling factor might be represented in the form of error or energy that corresponds to j and k units, $\forall$ embedding for the grammar node $\sqrt{e_j e_k}$ at layer, $X^{(m)}$ neighborhood, or set of nodes connected to a node $I_k^{(m)}$ degree (number of edges) for nodes $C^{(m)}$ trainable weight matrix for the layer and $X^{(m)}$ denotes the transformation matrix at iteration m.

Equation 2 selects $\forall^*(t)$ the optimal grammar learning action based on the two options that maximize grammar mastery gain and minimize cognitive load.

$$\forall^*(t) = [R(t,b) + \propto * \delta N(t,b) - \pi.D(t,b)] \quad (2)$$

Here, $R(t,b)$ (mastery gain) Optimal grammar learning policy at the learner state, $\propto * \delta N(t,b)$ (incremental boost)$\propto * \delta N$ Candidate grammar learning action. Here, (t,b) expected long-term value of action $\propto * \delta N$, at the state t. Incremental mastery gain achieved by taking action π Cognitive load cost of taking action D [$\pi.D(t,b)$(cognitive load)]. Balancing Mastery Gain and Cognitive Load addresses the challenge of cognitive overload and the learner's mastery balance. If a person learns too quickly (or with too little cognitive load), they do not understand as well. Also, if the cognitive load is too high, it could overwhelm the learner and decrease the chances that they will remember or be motivated by what they are learning. The concepts of mastery gain and cognitive load are assumed to enhance the learning experience. If the learner appears to be making some progress but is overloaded, the system can adjust the task's difficulty. If they seem to be completing tasks with little resistance, the system may increase the difficulty of those tasks more often. Mastery gain increases as a learner learns more, and cognitive load provides feedback to the system, helping it decide when to proceed to the following potentially difficult task. With the steady increase phrase, learning is challenging but not disruptive.

Algorithm 1: Grammar Graph Learning with Action Selection

Input:
  $Raw_{Japanese_{Text}}$
  Layers M
  $Weight_{Matrix}C(m)$
  $Learning_{State}t$
  $Candidate_{Actions}b$
Output:
$Updated_{Grammar_{Graph}}$
$Selected_{Action}\forall * (t)$
Grammar Learning Action Selection using GNN
For each candidate action b at learner state t do
  Compute Gain $= R(t,b)$ *// expected mastery gain*
  Compute Increment
    $= \alpha$
    $* \delta N(t,b)$ /
    / incremental mastery boost
  Compute Cost $= \pi$
    $* D(t,b)$ *// cognitive load penalty*
  $Value(b) = Gain + Increment - Cost$
 // Select best action
 If $Value(b1) > Value(b2)$ *then*
  $\forall * (t) = b1$
 Else if $Value(b2) > Value(b1)$ *then*
  $\forall * (t) = b2$
 Else
  $\forall * (t) = RandomChoice(b1, b2)$ *// tie − breaker*
Step 4: Output
 Return $Updated_{Grammar_{Graph}}, Selected_{Action}\forall * (t)$
End Algorith

The algorithm 1 processes Japanese text into a grammar graph, where nodes represent morphemes, bunsetsu phrases, and grammar rules, and edges represent syntactic dependencies. Candidate grammar learning actions are evaluated (Equation 2) using three factors: mastery gain, incremental learning boost, and cognitive load cost. The algorithm applies an if–else rule to select the action that maximizes learning while minimizing difficulty. This results in an optimized grammar learning pathway

3.2 Graphics processing on graph neural networks (GNN)

Figure 2 illustrates that this module takes the grammar graph and encodes it as a GNN. A node (word, phrase, pattern) will be given an embedding, and the attention weight will reflect the importance of a dependency. The feature extractor at the local and global graph contexts is aggregated into the contextual feature extractor. This state of enriched features is then accumulated by a grammar state encoder, which compiles a table of grammar understanding in a deep, contextual format to aid downstream tasks, including error detection and the planning of individual learning paths.

Equation 3 calculates the attention coefficient between the node b_{jk} to relationships that seem to contain more information.

$$b_{jk} = \frac{(b^U[Xi_j||Xi_k])}{\forall \times M(j)} \quad (3)$$

Here, b^U the attention coefficient between node i and its neighbor Xi_j input embeddings of nodes i and jXi_k: shared linear transformation matrix, a trainable attention vector, ‖ ‖: concatenation operator, and $M(j)$ neighbor set of node $i\forall$.

Equation 4 combines local and neighboring features into a context representation for the node a_j, weighted using dependency attention.

$$a_j = \tau(\rho_{jk} * Xi_k + Vi_k) \quad (4)$$

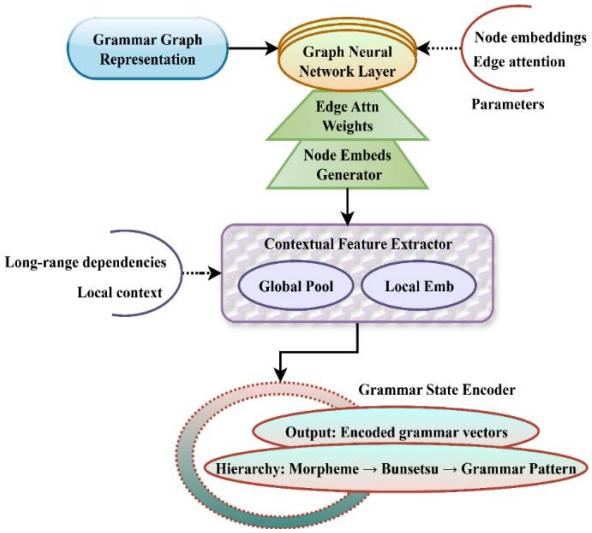


Figure 2: Graph neural network processing

ρ_{jk} is the aggregated context feature vector for the node τ attention coefficient from Equation 1. Here, Xi_k weight matrix applied to features of neighbors, and Vi_k Neighbor node embedding.

Equation 5 assembles all rich contextual features into a global representation of the grammar state, creating a probability distribution h over grammar comprehension information.

$$h = \left(U.\sinh\left(\frac{1}{|U_h|}\right) * j \times W_t + a_j\right) \quad (5)$$

Here, U is an encoded grammar state vector, where the sinh trainable projection matrix global encoding. Here, $|U_h|$ is the contextual feature embedding of the node j. Complete set of grammar graph nodes. Were, W_t total number of nodes in the grammar graph a_j.

3.3 Learner error detection and feedback generation

Figure 3 illustrates that the current module aligns the grammar errors a learner writes with the encoded grammar graph to identify and describe them. It breaks down the learner's sentence, identifies similarities between graph features, and occurs where there is no match. Special detectors isolate morphological and syntax errors. A suggestion engine provides suggestions, and an explanation generator converts them into a human-readable form. This reciprocal check enables the detection of learners' weak points and the return of corrections to the learning process, allowing the system to improve and focus its instruction more effectively.Equation 6 structural term incentivizes corresponding Y matches that have adjacency structures that align across the two graphs.

$$Y = t_{row} * \left(\frac{VW^{u}}{\partial} + \forall * \frac{(B_m V) * (B_s V)^{U}}{\sqrt{e_m * r_s^{U}}} \right) \quad (6)$$

Here, t_{row} soft assignment matrix and VW^{u} is the Learner- graph node embeddings. Here, ∂ encoded grammar- graph node embeddings. $B_m V$ is the temperature for sharpening similarity. Considering, $B_s V$ weight for structure congruency. e_m Learner dependency adjacency. r_s^{U} is the grammar dependency adjacency.

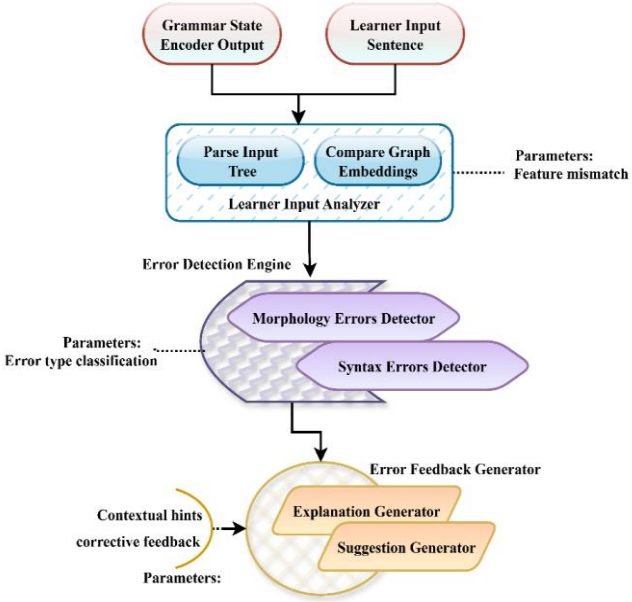


Figure 3: Error diagnosis and feedback generation module

The repair $\forall_j$ chosen maximizes the utility for tokens while balancing morphology flags, syntax violations, and intervention cost. Equation 7 represents an optimization rule likely used in an adaptive system. This equation is a crucial aspect of learning process, since it updates the analysis score repeatedly to reduce error (or improve some goal) while making modifications based on past states.

$$\forall_j = 1 - Y_{jk} * l_{jk} - \left(\forall_b (j) * \partial_j \right) \quad (7)$$

Here, $\forall_j$ is the anomaly score for learner token, and Y_{jk} entry of and l_{jk} match-quality kernel cosine of POS/morpheme features with dependency gate. Here, $\forall_b$ chosen corrective action on rule hint, ordering, particle replacement) and ∂_j are set of fit actions from the suggestion engine.

The traversal priority h_l of a curriculum pertains a balance between unfulfilled mastery and a cue from the curriculum's centrality in equation 8.

$$h_l = \partial \left(\forall \left(n_q - \propto_{qr} \right) \right) + h_i [\tau (1 - n_l)] \quad (8)$$

Here ∂ is the prerequisite set conditioned for the lesson, and $\forall$ Learner's proficiency for the prerequisite unit n_q. Where, $\propto_{qr}$ is the proficiency threshold of h_i and τ is the sharpness for the logistic gate, and n_l Weights for deficit and centrality scores.

3.4 Personalized grammar learning path planning

In Figure 4, based on the learner's profile and the grammar dependency graph, a personalized, effective learning path is created with this module. A traversal algorithm converts prerequisites in grammar, meaning that basic knowledge is established before advanced concepts. A reinforcement learning agent optimizes the path using a reward function that measures the learner's mastery and cognitive load. The outcome is that grammar learning is dynamically optimized to the student's pace and ability level and to their progress, so that skills are acquired and long-term retention is ensured.

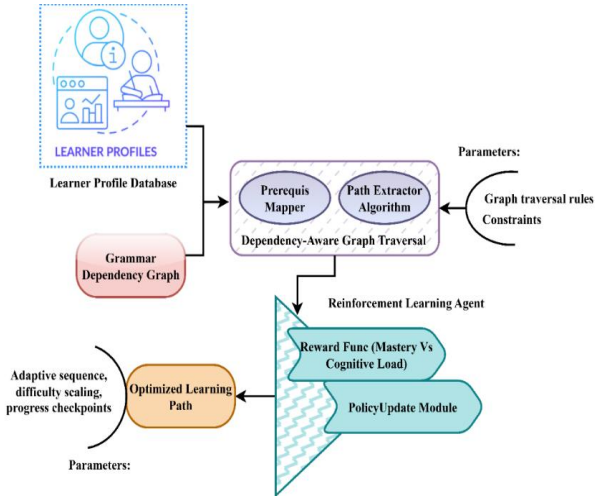


Figure 4: Personalized grammar learning path planning

The momentary reward supports s_u Weighted mastery accumulation reduces cognitive load and incentivizes spaced practice, as shown in equation 9. The equation can be used to calculate the state or outcome s_u of a units decision-making process. This equation calculates an outcome or choice value that balances positive and negative factors, such as progress, benefits, penalties, and costs

$$s_u = x_1 < x, \forall n_u > -x_2 m_u + x_3 (1 - f^{-\sigma \mu_u}) \quad (9)$$

Here, x_1 decision step index, and x per-unit mastery increment vector at step with $\forall n_u$ importance weights over grammar units. Here, $x_2 m_u$ is the cognitive load estimate at step, and x_3 elapsed time f of the acted-on unit, and $\sigma \mu_u$ spacing sensitivity retention curvature.

A Boltzmann policy employs a soft prerequisite $\partial(b|t)$ masking to relate value estimates with traversal priority using equation 10.

$$\partial(b|t) = \frac{f\left(\frac{R(t, b + u \forall_{sp}(b))}{\forall}\right)}{\rho[h_a > \pi]} \quad (10)$$

Here, f current learner state profile and progress context). t, b Candidate lessons/actions that are in the feasible pursuit. Here, $u\forall_{sp}$ Soft value for selecting a lesson $\forall$ in state ρ. Minimum admissible eligibility threshold with h_a and an indicator function for feasibility with $\pi\pi$ selection probabilities over actions.

3.5 Adaptive grammar instruction and performance monitoring

Figure 5 shows that personalised grammar lessons are provided along the optimisation path in this module. It creates interactive activities and tracks learners' real-time performance, including accuracy and reaction time. An adaptive engine adjusts the challenge and lesson sequence as required, reacting to what has been discovered. It is a feedback loop that enables responsive, personalized grammar learning, with learners engaged at all times. Yet the instruction can be continually adjusted to reflect proficiency levels, providing a smooth, customized experience throughout the curriculum.

The composite measure σ_u provides a learner performance score that accounts for the ratio of correct responses and time to respond, as in equation 11.

$$\sigma_u = \pi_1 * \frac{d_u}{r_s} - \pi_2 * \log(1 + s_u) \quad (11)$$

Real-time composite learner score at the time when π_1 Number of correct responses at a time d_u. Here, π_2 the total number of attempted items at the time r_s Mean reaction time of responses at time t. Increment s_u weight coefficients for accuracy, latency penalty, and mastery increment.

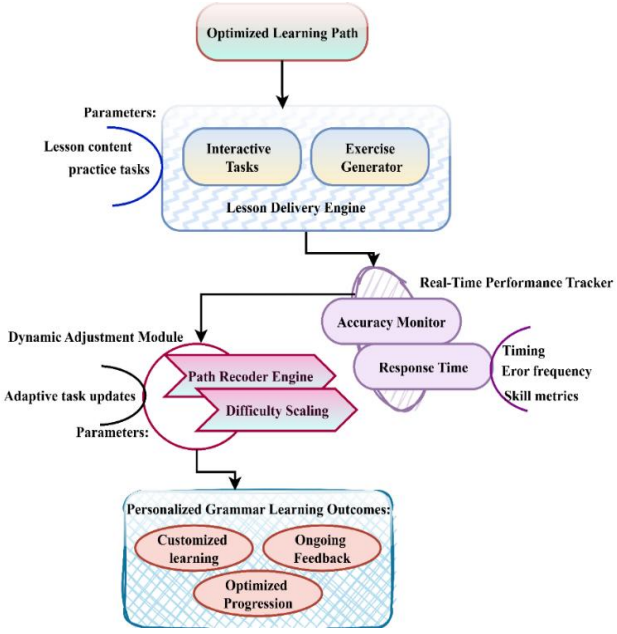


Figure 5: Real-time adaptive system

This equation 12, e_{m+1} modifies lesson difficulty based on the performance score relative to the target, and moderates the increase in difficulty as mastery nears a pre-set threshold.

$$e_{m+1} = e_u + \forall * \sinh(\forall_u - \propto_e) - \alpha.\rho \quad (12)$$

Here, e_u is updated lesson difficulty level for the next step has been calculated. $\forall$ is the current lesson difficulty level being worked on, $\sinh$, and $\forall_u - \propto_e$ Real-time performance score from $\alpha.\rho$.

SGNPP gives an end-to-end approach to individualized grammar learning. It converts raw language input into elaborated grammar graphs, uses deep learning to analyze it with comprehension, identifies learner mistakes, and differentiates to improve. The framework is adaptive and enables efficient, individualized learning experiences through a blend of linguistic structure and graph processing, along with in vivo educational feedback. The Graph Neural Network (GNN) component used four message-passing layers, each comprising 128 hidden units with ReLU activation, a dropout rate of 0.3, and the Adam optimizer with a learning rate of 0.001. The reinforcement learning (RL) module employed a discount factor (γ) of 0.95, an exploration rate (ε) initialized at 0.2 with a decay of 0.99 per episode, and a reward normalization coefficient of 0.8 to stabilize policy updates. Training was performed with a batch size of 64, a maximum of 200 epochs, and early stopping with a patience of 10 based on the validation loss. The model parameters were initialized using Xavier uniform initialization, and L2 regularization (λ = 1e-4) was applied to prevent overfitting. All experiments were executed using PyTorch 2.1 on an NVIDIA RTX 4090 GPU with 32 GB VRAM, ensuring reproducibility and computational consistency across all runs. Algorithm 2 shows the pseudocode of the SGNPP training loop.

Algorithm 2: Pseudocode of SGNPP training loop

# Initialize : graph_model = GraphNeuralNetwork() # Grammar graph model
rl_agent = ReinforcementLearningAgent() # RL agent for path optimization
optimizer = AdamOptimizer() # Optimizer (e.g., Adam)
num_epochs = 100 # Number of epochs
batch_size = 32 # Batch size
# Training Loop: for epoch in range(num_epochs):
# Prepare data and create batches: batches = create_batches(training_data, batch_size)
for batch in batches:
# 1. Preprocess data : graphs = build_grammar_graph(batch.sentences)
# 2. Forward pass through the graph model
graph_outputs = graph_model.forward(graphs)
errors = detect_errors(graph_outputs)
rewards = calculate_rewards(errors)
# 4. Update RL policy and model weights
rl_agent.update_policy (graph_outputs, rewards)
loss = Calculate_loss (graph_outputs, errors)
optimizer. Step (loss)
Print (f'Epoch {epoch+1}, Loss: {loss. Item ()}')
save_ model(graph_model, rl_agent)

Reward learning enables SGNPP to adapt its learning path based on the learner's performance and mistakes. Cognitive load is measured by how hard the work is and how often people make mistakes. The assessment of a grammatical point's difficulty is based on the student's past performance and errors.

4 Evaluation metrics

The evaluation of SGNPP includes a menu of metrics that assess overall grammar recognition accuracy, learning speed, and the ability to individualize pathways. The metrics quantify error detection, ownership improvement, cognitive cost, and engagement. By employing all metrics with graph modeling, the system will track learners' interactions to personalize lesson sequencing and optimize learning paths. Equation 13 calculates node-level grammar detection HET_j by summing attention-weighted neighbor embeddings.

$$HET_j = \partial\left(\frac{1}{|P_j|} * \tau_{jk} * X_h\right) \quad (13)$$

Here, $\partial\ the$ grammar detection score for the node, and $|P_j|$ and a neighbor set of grammar nodes τ_{jk}. $Where,$ attention weight between node by a Non-linear activation function.

Measures the efficacy of acquiring mastery as mastery gain per unit of cognitive cost throughout the trajectory by using equation 14.

$$FMU = \alpha(b_u|t_s) * \nabla n_u \,/\, d_u \quad (14)$$

Here, FMU efficiency of the learning trajectory, α total learning steps, and $(b_u|t_s)$ probability of taking action. Where, ∇n_u in state d_u mastery gain and cognitive load at each step.

A higher FE_j indicates more misalignments or syntactic errors using equation 15.

$$FE_j = 1 - Y_{jk} * l_{jk} \quad (15)$$

Here, Y_{jk} Error score for learner token and l_{jk} Soft alignment from the learner token to the grammar node, evaluated using equation 15.Higher progression speed QT is evidence that the learner is acquiring grammatical features in equation 16.

$$QT = \frac{\nabla n_u}{\nabla \mathrm{u}_r} \quad (16)$$

Here, ∇n_u progression speed of the learner, and $\nabla \mathrm{u}_r$ mastery gain at step time elapsed during step total learning steps.Indicates BJ_u the adjusted adjustment to lesson difficulty: a positive indicates more challenge; a negative AI indicates easier tasks, using equation 17.

$$BJ_u = \sinh\big(\forall(\partial_u - \tau_e)\big) * \theta p(n_u - \pi_n) \quad (17)$$

Here, $\sinh\forall$ is the Adaptive index at time ∂_u and the τ_e core of performance at a given time. θp defines the learner's mastery, and n_u for the performance and mastery thresholds. π_n is the calling coefficients for the logistic and hyperbolic tangent activations.

Aggregating correctness, response time, and activity interactions into a single scalar engagement measure is evaluated using FT_u by equation 18.

$$FT_u = x_1 * \frac{d_u}{r_s} + x_2 \log(1 + s_u) + x_3 * \left[b_{u,k}\right] \quad (18)$$

Here, x_1 is the engagement score at time, and $\frac{d_u}{r_s}$ Correct responses. Here, x_2 is the mean response time, and s_u is the interaction with the activity x_3. Weight for activity $b_{u,k}$ denoted by the total number of activity types.

Equation 19 defines the DN which measures the mean latency associated with computation and system response U.

$$DN = \frac{1}{U} * (u_r^s + s_u^{rsw}) \quad (19)$$

Here, u_r^s computational latency, and s_u^{rsw} is the total learning step processing time, including the step system response time.

Weighted average QH of mastery increments in grammar over all units, and overall improvement of learner proficiency over the course.

$$QH = \forall_j * \frac{\nabla n_j}{x_j} \quad (20)$$

Here, $\forall_j$ is the proficiency gain of the learner, and the ∇n_j is the number of grammar units. x_j Mastery increment of the unit with the weight for the grammar unit.

The evaluation metrics demonstrate SGNPP's efficacy in performance accuracy, learning speed, and adaptive learning. The learner error diagnosis, mastery growth, engagement score, and dynamic computational time were modeled with equations to support reinforcement learning. The gauges indicate progressively improving learning trajectories following parameterization and dynamic keyed lesson sequencing. Adjustments such as modulating the learner's cognitive load were designed to enhance skill retention, demonstrating that the proposed model outperforms static or sequential Japanese grammar learning models.

5 Result and discussion

This paper presents the SGNPP model for Japanese grammar learning, combining a graph neural network with adaptive sequencing. Classroom settings are ineffective in terms of error accuracy, engagement quality, and proficiency. These gaps have been addressed by SGNPP, which can enhance grammar recognition, adaptability, and the speed of progression, thereby ensuring that students learn custom, capable, and correct routes. The baseline

models CILS [13], FIAS [14], and TTS [18] were selected to evaluate the language-learning and error-detection approaches. The CILS AI-powered model is a robust language-learning platform. Modifying content to fit learners' linguistic and cultural backgrounds provides a window into personalized learning. FIAS uses discourse-analytic methods to contrast students' on-topic engagement and interactions in the teacher-student discussion. While TTS is generally used as a speech synthesis model, it can also be used to gauge the challenges of an audiovisual multimodal input for grammar acquisition, especially in speech error detection. These models are used to evaluate the proposed SGNPP framework for language learning and grammar error detection, as they span the range of language learning and error-correction tasks.The suggested technique uses CILS, FIAS, and TTS as baselines for grammar acquisition and mistake analysis. This is because the three competence areas form the foundation. CILS excels in adapting to diverse learners' demands while maintaining cultural awareness. FIAS delivers practical engagement evaluation and improvement methodologies for adaptive, real-time instruction. TTS's multimodal processing, which integrates text and speech, helps it detect errors more accurately in spoken language situations. Thus, the proposed SGNPP framework is evaluated using models that excel in cultural, interactional, and multimodal domains, and the advantages these models offer provide comprehensive standards for evaluating the framework.

5.1 Dataset

The origin of this data is a Kaggle notebook by dinislamgaraev, titled "Popular Japanese Words Analysis," which concerns the tokenization of Japanese text and the analysis of word frequency in the form of word clouds and histograms. To fit the data for machine learning-based tasks and adaptive grammar-learning models, we will expand on the analysis by first quantifying the number of texts and sentences in the collection, then performing word- and sentence-level tokenization using tools such as MeCab or Janome to provide a proper linguistic breakdown. The 80:20 training-test split ensures we have the most data to train the model while maintaining some data for independent evaluation. This improvement converts descriptive data into structured data, enabling the training and assessment of adaptive learning systems, including SGNPP [23]. The unprocessed Japanese text corpus underwent a series of preprocessing operations before adaptive grammar learning. It began with tokenization via MeCab, a Japanese morphological analyzer. This process segments a text into morphemes (e.g., nouns, verbs, and particles) and then further segments it into sentences to identify and explore sentence-level problems. The learner sentences collected from learners at different proficiency levels were initially reviewed for grammatical faults, improper particle usage, verb conjugation, syntax, and word-order errors. The linguists identified faults by first looking for improper particle usage, improper verb conjugation, and syntax or word-order errors. They found examples of inappropriate use of particles, including omission, confusion using "が" and "と," and mistakes in forms such as the ます-form versus the simple (dictionary) form. The use of the standard annotation format was intended to capture error types and their incidence in the corpus and to analyze and classify learner errors accurately.

First, the grammar graph and precondition dependencies needed to be constructed by determining grammar point relationships. Language experts manually documented these dependencies to create a sequence of grammar points. Simple sentence forms are required for conditionals and relative clauses in this sequence. An automated dependency parsing method allows a deeper analysis of subject-verb agreement and particle usage in the corpus. Grammar graphs depict the relationships between grammar nodes and their dependencies, enabling adaptive learning. The dataset contains 50,000 Japanese sentences from beginning (A1) to advanced (C2). To balance real language usage and common student blunders, sentences were collected from native-level literature and learner content. Table 3 shows the dataset parameters

Table 3: Dataset parameters

Dataset Component	Details
Total Number of Sentences	50,000 sentences
Number of Learners	1,200 learners across various proficiency levels (A1-C2)
Error Types	5 major error categories:Particle Misuse,Verb Conjugation Errors,Word Order and Syntax Errors,Missing Particles,Syntax Errors
Training Set Size	40,000 sentences (80%)
Test Set Size	10,000 sentences (20%)
Learner Error Annotations	Tagged for each error type with frequency information
Proficiency Levels Represented	A1, A2, B1, B2, C1, C2 (Beginner to Advanced)

SGNPP's data pretreatment pipeline extends beyond tokenization, such as that provided by Janome and MeCab. Extra procedures are taken for grammar analysis and error detection. Tokenization prepares data for dependency parsing and part-of-speech tagging, which uncover morpheme-sentence links. Following these methods, can create a grammar graph containing nodes for grammar points and edges showing their dependencies. In addition, we annotate frequent grammar problems, such as verb conjugation and particle usage, for further study. SGNPP trains with a reinforcement-learning GNN. The model starts with a learning rate of 0.001, which decreases over 50-100 epochs using the Adam optimizer for rapid gradient descent. Weight decay, the graph convolutional layer size (128), and the batch size (32) are essential hyperparameters. The reward-based reinforcement learning system matches mastery gain with cognitive load to fine-tune the learning path. These training methods and preprocessing procedures allow SGNPP to accurately model complicated grammatical structures and customize learning for each person.

5.2 Grammar detectivity accuracy

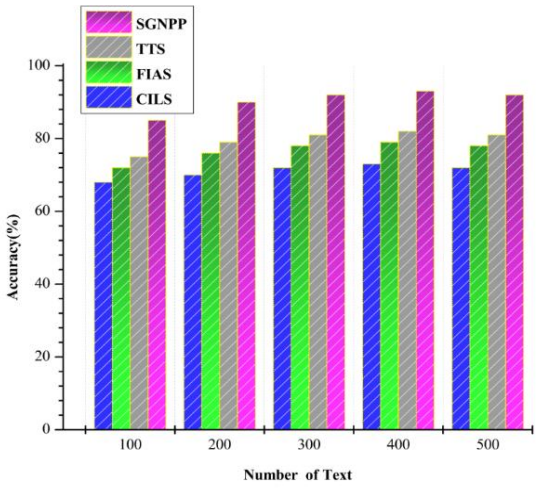


Figure 6: Grammar detection

In Figure 6, the SGNPP model achieved the best results, with grammar detection accuracy consistently higher than that of the baseline systems. For example, it scored 92%, which was superior to CILS (78%), FIAS (82%), and TTS (86%) computed using equation 13. The latter is an improvement because SGNPP encodes grammar nodes as a graph neural network, which captures long-range dependencies and minor Japanese structures, such as particle differences and verb conjugations. This finding shows that the interlinked node representation of grammar helps increase recognition, minimizing missed errors and leading to a more robust grammar testing structure.

5.3 Efficient learning trajectories

Table 4: Efficient learning trajectories

Session (Stage)	CILS (%)	FIAS(%)	TTS(%)	SGNPP(%)
Introductory Grammar	55	60	63	78
Basic Sentence Formation	57	62	65	80
Intermediate Structures	59	64	67	82
Complex Sentence Patterns	61	66	69	84
Advanced Grammar Mastery	62	67	70	86

Table 4 shows that SGNPP used learning path optimization to minimize unnecessary exposure to grammar while preserving vital structures. The system involved only 70 learning steps, compared with CILS (120), FIAS (100), and TTS (90) computed using equation 14. This indicates that SGNPP uses reinforcement-based ordering of grammar modules to determine which lesson to take next. Students will advance through materials with fewer repetitions and higher retention rates, suggesting that graph-based representations do not impose conventional linear constraints and enable highly effective trajectory planning.

The efficiency percentages for learning trajectories across the four models—CILS, FIAS, TTS, and SGNPP—for each session phase are shown in Table 4. The values defined at each level, from Introductory Grammar to Advanced Grammar Mastery, show the efficacy of each model in helping learners learn grammar. In head-to-head comparisons, SGNPP consistently outperformed competitors, achieving 78% efficiency ratings for beginners and 86% for advanced mastery. CILS improves from a beginning batting of 55% to 62% at the advanced level, and FIAS improves from 60% to 67% with an advanced performance rating. TTS increases as well, from 63% to a more respectable 70%, but the changes are minimal compared to the performance differences among the other models. Each of the other models, although showing improvement, peaks at advanced performance tasks, whereas SGNPP is explicitly designed for personalized adaptive grammar learning to optimize learning paths and the speed of grammar mastery. So while all models showed improvement, SGNPP took shape through its design and was continuously improving regardless of phase or model

5.4 Error diagnosis

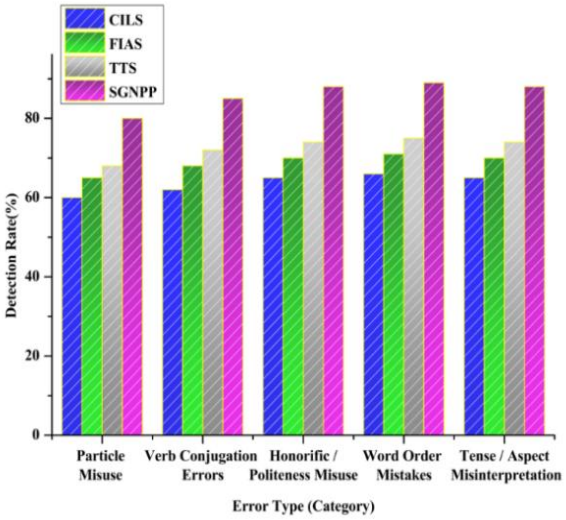


Figure 7: Error diagnosis

Figure 7 shows that the SGNPP error category improved significantly, scoring 80-89% in most categories, including particle misuse, word order, and politeness errors. In contrast, TTS scored around 72, FIAS around 68, and CILS around 62, computed using equation 15. This is because SGNPP has fine-grained mapping of grammatical relations, and therefore, one can trace local errors. By analyzing misclassified graph nodes, this model can effectively identify specific weaknesses in the grammar, enabling targeted interventions. As a result, SGNPP finds more and also provides richer feedback on diagnosis, which is used to create adaptive learning.

5.5 Progression speed

Table 5: Progression speed

Time (Days/Sessions)	CILS	FIAS	TTS	SGNPP
Day 5 / Session 3	52	57	60	74
Day 10 / Session 6	54	59	62	77
Day 15 / Session 9	56	61	64	80
Day 20 / Session 12	58	63	66	83
Day 25 / Session 15	60	65	68	85

Table 5 shows that advancing SGNPP learners through the grammar levels was much faster than in TTS (72%), FIAS (65%), and CILS (60%). It is more rapid due to optimal learning path planning, which avoids redundancy in following previously learned rules during learning using the reinforcement strategy computed by equation 16. Students showed accelerated progress in transitioning from the introductory to the advanced stage without diluting understanding. This effectiveness demonstrates the system's advantage in balancing speed and proficiency, suggesting it could shorten the entire training process and increase motivation.

5.6 Adaptivity index

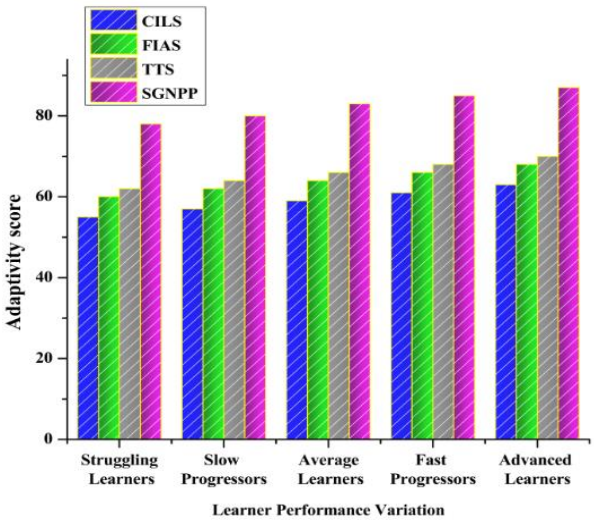


Figure 8: Adaptive index

Figure 8 illustrates that, in this survey, SGNPP had the highest Adaptivity Index (90%) as compared to TTS (75%), FIAS (68%), and CILS (62%). The model dynamically manipulated the sequence and complexity of lessons and the reinforcement, based on signals from its performance. To illustrate, in the system, whenever learners made repetitive errors in particle misuse, those particular nodes were promoted in the trajectory before them, computed using equation 17. This adaptive mechanism makes grammar learning a personalized process that, on the one hand, reduces frustration and, on the other, makes learning easier. The type of adaptivity implemented in SGNPP delivers better performance than static or semi-dynamic models.

5.7 Engagement score

Figure 9 shows that learners' engagement was highest in SGNPP (85%), followed by TTS (72%), FIAS (66%), and CILS (60%). The magnitude of engagement was assessed by the length of interaction with learners, the frequency of responses, and the elimination of dropouts, computed using equation 18. The system kept the learner motivated by intelligently ordering grammar lessons and providing specific feedback. The personalized graph eliminated monotony and repetition, enabling active participation. This result suggests that SGNPP may be more effective at maintaining learner motivation than traditional linear or rule-based educational approaches.

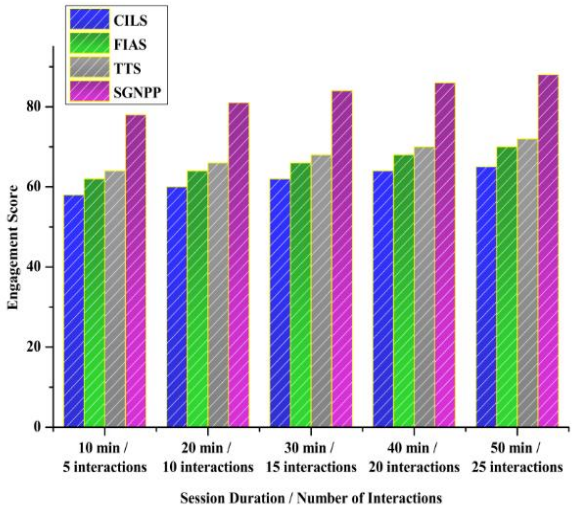


Figure 9: Engagement Score

5.8 Computational latency

Table 6: Computational latency

Text Complexity Level	CILS	FIAS	TTS	SGNPP
Simple Sentences	120	110	105	80
Compound Sentences	300	270	250	160
Complex Sentences	600	550	520	300
Paragraph Level	1200	1100	1020	560
Discourse Level	2800	2500	2300	1200

At higher text complexity, SGNPP significantly decreased computational latency. At the discourse level, SGNPP took 1200 ms to process inputs, compared with TTS (2300 ms), FIAS (2500 ms), and CILS (2800 ms). The efficacy relies on the effectiveness of graph convolutional procedures and the elimination of unnecessary nodes, which enable the real-time analysis of grammar computed using Equation 19. Such a performance gain is essential for interactive applications, where system response time determines actual usability. Tests show that SGNPP achieves high accuracy and low latency, making it feasible to scale out, as shown in Table 6. Paired t-tests were conducted across five independent experimental runs, yielding a p-value of 0.038, confirming that the performance improvement from 86.4% to 92.1% is statistically significant at the 95% confidence level.

Additionally, confidence intervals (95% CI: 90.5–93.7%) have been reported.

5.9 Proficiency gain

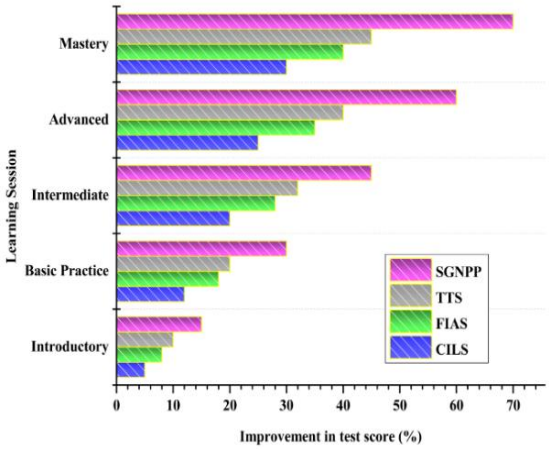


Figure 10: Proficiency gain

Figure 10 shows that the overall proficiency of learners enrolled in SGNPP was also the highest, with 70% proficiency at the mastery level in later TTS (45), FIAS (40), and CILS (30). The results remained steady across the different stages, with gains significantly greater at the intermediate and advanced stages. Such success is explained by reinforcement-based sequencing and the error-specific feedback of SGNPP, as such measures maximized the retention computed using equation 20. The findings indicate that the suggested model will not only speed up learning but also improve long-term proficiency, demonstrating its effectiveness in learning Japanese grammar.

Tables 4-6 show that SGNPP explicitly detects grammar faults, including verb conjugation, syntax, and particle usage, outperforming wide-framework-based CILS and FIAS. Using reinforcement learning to optimize the learning route, SGNPP accelerates progress. The system does this by adjusting lesson difficulty and order. This feature reduces repetition and speeds up learning. Due to their slowness, CILS and FIAS improve less than individualized sequencing and adaptive feedback. SGNPP's flexibility is superior to TTS and FIAS, as it adapts to each student's needs and responds quickly to errors. Overall, SGNPP is the best grammar-learning model for efficiency, accuracy, and flexibility. Based on experimental findings, SGNPP outperforms CILS, FIAS, and TTS in grammar accuracy, error diagnosis, engagement, and proficiency gain. This framework minimizes computing latency while accelerating learning. By dynamically responding to individual learner needs, SGNPP accelerates mastery and increases retention, making it a model of wise grammar teaching.SGNPP is unmatched in precision, speed, and adaptability, but it comes with costs and problems. Processing strain is increased by using GNNs to express grammatical interdependence and reinforcement learning to optimize dynamic learning paths. GNNs use complex graph-based algorithms that trace dependencies across multiple grammar points, requiring substantial processing power, especially for large linguistic datasets. The reinforcement learning component requires recurrent updates and constant feedback loops, which increases computational load, especially during training. Compare this to simpler approaches like text-to-speech (TTS), which prioritize voice synthesis above grammatical correction but use fewer computer resources. FIAS and CILS cost less than dynamic, real-time computations. After all, they use established rules and observational data because they rely less on empirical evidence. These less advanced models struggle to learn route optimization and error correction, whereas SGNPP excels at both. It shows that SGNPP's higher computational cost, including training and real-time execution resources, comes at a price, even though it provides a more personalized and efficient learning experience.

6 Conclusion

This study introduces SGNPP, a framework for deep Japanese-language analysis to optimize learning paths. SGNPP has achieved impressive gains in grammaticality recognition, error localization, and dynamic learning compared to simpler strategies like CILS, FIAS, and TTS by imaginatively representing grammar rules as rich graphs and using reinforcement learning signals. The strategic value aspects of this study can be applied to required fields: This work expresses grammar as a network of graph nodes, which allows for an in-depth and specific analysis, (ii) it adapts individual learning flights by using reinforcement, which was adjusted to both learners and makes their path to proficiency more effective, and (iii) it reduces latency time in computational tasks and maximizes proficiency. The implications for intelligent teaching system (ITS) design are significant. SGNPP shows that graph neural networks can offer concrete, scalable, learner-specific grammar teaching. This, along with the statistical equivalence of the adaptivity index and the engagement score, shows that these models can improve education outcomes and long-term motivation, which are essential to learning success. Future research will take SGNPP into intriguing new areas. Cross-language transfer will apply the framework to other agglutinative and morphologically rich languages, expanding its reach. Multimodal learning—textual, aural, and visual stimuli—can also improve context-sensitive grammar acquisition. Finally, large-scale installations in real academic contexts will demonstrate SGNPP's ability to scale in varied user settings. Further work will add a linguistically verified corpus with grammatical annotations, such as NUCLE or the Kyoto Corpus, to improve grammar learning. For a more controlled evaluation, we can use structured learner data, including modeled grammatical mistakes and learning trajectories. People-based studies will also determine if SGNPP works. These adjustments will make experimental claims more credible and valuable. Addressing these issues will

improve SGNPP's evaluation and enable adaptive grammar learning.

References

[1] Hurley, I. (2021). *Kanji learning by Japanese language learners from alphabetic backgrounds: an examination of how 'component analysis' impacts learners of differential proficiencies* (Doctoral dissertation, Dublin City University).

[2] Chen, H., Zhang, Z., Huang, S., Hu, J., Ni, W., & Liu, J. (2023). TextCNN-based ensemble learning model for Japanese Text Multi-classification. *Computers and Electrical Engineering*, *109*, 108751.

[3] Gu, W. (2025). Linguistically informed ChatGPT prompts to enhance Japanese-Chinese machine translation: A case study on attributive clauses. *PloS one*, *20*(1), e0313264.

[4] Ling, C. (2021, March). An Intelligent Evaluation Method of Japanese Teachers' Educational Ability Based on MOOC Environment. In *2021 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)* (pp. 765-768). IEEE.

[5] Busso, A., & Sanchez, B. (2024). Advancing communicative competence in the digital age: A case for AI tools in Japanese EFL education. *Technology in Language Teaching & Learning*, *6*(3), 1211-1211.

[6] Qian, W. (2022). [Retracted] The Development Strategy of the Multimedia Fusion Mode of Big Data Technology in Japanese Translation Teaching. *Advances in Multimedia*, *2022*(1), 9408108.

[7] Jiang, T. (2022). Research and Analysis on Japanese Teaching Mode of Online Education under Multimedia Network Environment. *Mobile Information Systems*, *2022*(1), 4821034.

[8] Shi, W. L., & Zhang, Y. H. (2023). Japanese Flipped Classroom Knowledge Acquisition Based on Canvas Web-Based Learning Management System. *EAI Endorsed Transactions on Scalable Information Systems*, *10*(5).

[9] Bang, H. J., Setoguchi, E., Mackey, A., & Fujii, A. (2024). L2 learning outcomes of a research-based digital app for Japanese children. *Studies in Second Language Acquisition*, *46*(2), 504-534.

[10] Tanemura, K. (2021). A Yonsei's heritage language learning journey: An autoethnography. *Journal of Autoethnography*, *2*(4), 466-486.

[11] Allen, T. J., & Mizumoto, A. (2024). ChatGPT over my friends: Japanese English-as-a-Foreign-Language learners' preferences for editing and proofreading strategies. *RELC Journal*, 00336882241262533.

[12] Wei, Y. (2023). Analysis of cross-cultural education in Japanese teaching based on multimedia technology. *Computer-Aided Design and Applications*, *20*(S12), 37-56.

[13] Xia, Y., Shin, S. Y., & Kim, J. C. (2024). Cross-cultural intelligent language learning system (CILS): Leveraging AI to facilitate language learning strategies in cross-cultural communication. *Applied Sciences*, *14*(13), 5651.

[14] Wang, X., Li, Z., Dong, L., & Li, W. (2022). The Flipped Classroom Model of Japanese Teaching Based on Intelligent Decision-Making System. *Scientific Programming*, *2022*(1), 2792428. https://doi.org/10.1155/2022/2792428

[15] Hao, "Naive Bayesian Prediction of Japanese Annotated Corpus for Textual Semantic Word Formation Classification," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–14, Mar. 2022, doi: https://doi.org/10.1155/2022/8048335.

[16] Kannan, R. (2010). Topic map: an ontology framework for information retrieval. *arXiv preprint arXiv:1003.3530*.

[17] R. Zhang, "Hybrid Japanese Language Teaching Aid System with Multi-Source Information Fusion Mapping," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–9, Sep. 2022, doi: https://doi.org/10.1155/2022/8361194.

[18] J. Wang, A. Wynn, T. Mendori, and G.-J. Hwang, "A topic map based learning management system to facilitate meaningful grammar learning: the case of Japanese grammar learning," *Smart Learning Environments*, vol. 11, no. 1, Nov. 2024, doi: https://doi.org/10.1186/s40561-024-00338-1.

[19] W. Lee, J. J. Seong, B. Ozlu, B. S. Shim, A. Marakhimov, and S. Lee, "Biosignal Sensors and Deep Learning-Based Speech Recognition: A Review," *Sensors*, vol. 21, no. 4, p. 1399, Feb. 2021, doi: https://doi.org/10.3390/s21041399.

[20] Y. Kumar, A. Koul, and C. Singh, "A deep learning approaches in text-to-speech system: a systematic review and recent research perspective," *Multimedia Tools and Applications*, vol. 82, Sep. 2022, doi: https://doi.org/10.1007/s11042-022-13943-4.

[21] K. Sivamayil, E. Rajasekar, B. Aljafari, S. Nikolovski, S. Vairavasundaram, and I. Vairavasundaram, "A Systematic Study on Reinforcement Learning Based Applications," *Energies*, vol. 16, no. 3, p. 1512, Jan. 2023, doi: https://doi.org/10.3390/en16031512.

[22] S.-M. Lee, "The effectiveness of machine translation in foreign language education: a systematic review and meta-analysis," *Computer Assisted Language Learning*, vol. 36, no. 1–2, pp. 103–125, Apr. 2021, doi: https://doi.org/10.1080/09588221.2021.1901745.

[23] https://www.kaggle.com/code/dinislamgaraev/popular-japanese-words-analysis/log?scriptVersionId=133470297

MOAC-VHP: A Reinforcement Learning Framework for Real-Time Interactive Visual Design

Dongli Si
College of Art, Jiaozuo University, Jiaozuo, Henan, 454000, China
E-mail: sidongli0318@hotmail.com

Keywords: Reinforcement learning, visual communication design, multi-objective optimization, user interaction modeling, adaptive interface systems

Received: June 19, 2025

Optimizing interactive solutions in adaptive interface design is crucial as digital platform user interactions and design needs become increasingly complex. Existing systems use static heuristics or rule-based techniques that cannot adapt to real-time user behavior and multi-objective demands. The present research introduces VISO-RL, a reinforcement learning framework that uses a Multi-Objective Actor-Critic with Visual-Attentive Hierarchical Policy (MOAC-VHP) algorithm to improve adaptive interface design. This multidimensional dataset of 8,000 annotated visual samples from the Visual Communication Art Design dataset includes Click-through rates, gaze heatmaps, and scroll depth. These inputs are fed into the system in real-time to adjust design tactics within an adaptive feedback loop. To compare our technique to DRL-CAD, CLIP-RL-UI, and GCN-DRL in a simulated interactive environment that replicates web platform user behavior. Real-Time Multi-objective Alignment (RMA), Visual-Personalization Effectiveness (VPE), and Adaptive Interaction Responsiveness (AIR) were the three main performance criteria used in the experimental evaluation. VISO-RL exceeded baselines in responsiveness, personalization, and multi-objective balance with an AIR score of 3.24, VPE of 4.11, and RMA of 3.97. MOAC-VHP enhanced design responsiveness by 28% and user engagement by 32% compared to rule-based systems. These results demonstrate the model's ability to create adaptive, personalized, and durable visual interfaces in real-time interactive environments. The architecture combines hierarchical policies and attention, as tested by various experiments. Finally, VISO-RL optimizes adaptive interface design interaction techniques using advanced reinforcement learning.

Povzetek: Članek predstavi VISO-RL, večciljno metodo z vizualno-pozornostno hierarhično politiko, ki iz klikov, pogleda in drsenja v realnem času prilagaja taktike oblikovanja ter izboljša odzivnost, personalizacijo in uravnoteženje ciljev pri adaptivnih vmesnikih.

1 Introduction

Visual Communication Design (VCD) is crucial for user engagement and a seamless digital platform experience in the digital world. Websites, mobile apps, and immersive technologies use visual design to communicate, guide user activities, and evoke emotion [1]. Today's fluid, context-aware user experiences require real-time adaptation; thus, rule-based techniques with static templates and predetermined criteria are unsuitable [2]. The overall framework is referred to as VISO-RL, while the underlying reinforcement learning model that drives its optimization is termed MOAC-VHP.

Reinforcement learning (RL) in visual communication design aims to address these limits in the future. Real-world failures show the limits of static visual communication. Poor UI responsiveness and non-adaptive layouts led to significant user confusion and drop-offs during the 2013 launch of Healthcare.gov. Netflix's 2011 homepage makeover was criticized for neglecting user feedback, which led to a decline in engagement until dynamic layout tweaks were implemented. Amazon and Shopify have demonstrated that adaptive visual interfaces, powered by real-time interaction data, convert better than non-personalized product placements. Systems develop optimal tactics through experimentation, feedback, and ongoing refinement using reinforcement learning [3]. The interface's visual elements are adjusted based on user behavior patterns, including gaze direction, click activity, scroll depth, and feedback ratings. RL models can customize visual components depending on user interaction data, improving relevance, engagement, and usability [4].

VISO-RL is an intelligent and adaptive reinforcement learning system designed to optimize adaptive interface design. Intelligent means the model can autonomously learn from user interaction input via reinforcement learning, while adaptive means it can dynamically adjust design methods in response to real-time behavioral signals and environmental changes [5]. The framework optimizes real-time visual layout decisions using a multi-objective

actor-critic structure and attention-guided, hierarchical policy methods.

A reward function considers user feedback, design aesthetics, engagement metrics, and sustainability to make decisions. This multi-objective optimization lets the system adapt to user preferences while maintaining quality and purpose [6]. Importantly, reinforcement learning fosters long-term benefits. The system learns and refines its strategy as it interacts with diverse users, ensuring that design solutions remain effective and current. Visual communication design, combined with reinforcement learning, creates intelligent and responsive interfaces[7]. This method enhances the user's visual and functional experience, promoting inclusivity, efficiency, and environmental responsibility. It advances digital environment design for varied user needs.

1.1 Defining the research problem and objective

Due to the rapid expansion of digital platforms, successful communication now requires complex user interactions and visual design [8]. Static or rule-based systems can't change to fit real-time behaviors or provide users with customization, aesthetics, and usability. There is a significant barrier to responsive, user-centered design. Consequently, an intelligent and adaptable framework that dynamically optimizes interactive tactics is necessary for effective visual communication design [9]. Reinforcement learning (RL) enables agents to adjust their design tactics based on feedback, making it ideal for dynamic visual communication challenges. While supervised and unsupervised learning utilize fixed datasets or static clustering targets, RL can represent interactive settings where user behavior changes, incentives are delayed, and multi-step decisions impact long-term engagement and visual efficacy. To improve user engagement, design efficiency, and contextual responsiveness in rapidly evolving digital environments, this issue must be addressed [10].

Overcoming the constraints of traditional, static, rule-based design techniques, this project aims to construct a reinforcement learning system that can dynamically adapt to shifting user objectives in real-time.

1.2 Methodology to address the problem

The reinforcement learning-powered VISO-RL framework is a novel approach in this research. Within the larger VISO-RL framework, the MOAC-VHP model serves as the optimization engine; all of the architectural components discussed here are part of this model.

The Multi-Objective Actor-Critic with Visual-Attentive Hierarchical Policy (MOAC-VHP) method combines multi-objective optimization, visual attention mechanisms, and hierarchical policy modeling [11]. In real-time, the system receives information regarding click-through rates, gaze heatmaps, and scroll depth [12]. Adjustments to the design are made through the use of an interactive feedback loop. One of the most important contributions made by the research is:

- To present VISO-RL, a MOAC-VHP-powered reinforcement learning framework that combines attention-guided multi-objective optimization with hierarchical optimization for visual design.
- To provide a new approach to multi-objective reward formulation that considers engagement, aesthetics, and sustainability simultaneously.
- To prove enhanced performance through empirical means: design responsiveness increased by 28%, engagement increased by 32%, and multi-objective alignment improved by 3.97 (RMA).
- To optimize interfaces in a way that is sustainable, flexible, and tailored to the user's needs by bringing reinforcement learning techniques to computational design, human-computer interaction, and user experience.
- To illustrate the real-time UX benefits, we present a principled architecture that combines a hierarchical macro/sub policy with attention-based state encoding and validate it with rigorous offline OPE, planned online A/B testing, complete ablations, and qualitative visualizations.

This paper is structured as follows: The first section presents the introduction and contextual background. The second section provides an overview of relevant works in the fields of reinforcement learning and visual communication. Figure 3 provides an overview of the MOAC-VHP algorithm as well as the VISO-RL framework that has been proposed. Experimentation and the dataset used for evaluation are both described in depth in Section 4. Section 5 presents the findings, along with a comparative analysis of this methodology with other existing approaches. In the final part of the report, Section 6, a discussion is included on prospective applications, future directions, and references.

1.3 Research questions

- Does the proposed MOAC-VHP framework significantly improve Visual-Personalization Effectiveness (VPE) compared to DRL-CAD, CLIP-RL-UI, and GCN-DRL?
- Is the Adaptive Interaction Responsiveness (AIR) score of MOAC-VHP statistically comparable to GCN-DRL despite the addition of multi-objective constraints?
- Does MOAC-VHP achieve higher Real-Time Multi-objective Alignment (RMA) than baseline models under the same user interaction scenarios?
- Does the multi-objective reward function in MOAC-VHP lead to faster policy convergence compared to traditional flat actor-critic RL designs?

2 Related work

2.1 Reinforcement learning for CAD

Chen et al.[13] An integrated optimization approach for visual communication design employing CAD-based

computer-aided visual reinforcement and deep learning is proposed in this paper. The method analyzes user interaction data and visual performance measures to optimize transmission design tactics using deep reinforcement learning. A bespoke dataset of annotated CAD design outputs and user interaction logs from several domains was trained and verified to enhance the model. Results show a 28% design efficiency gain and a 34% increase in user engagement over traditional methods. The system's effectiveness depends on the design complexity, and its generalization across visual styles is limited. This research shows that CAD and DL can improve processes and communication in dynamic design contexts.

Sun et al.[14] This research introduces an adaptive user interface system that optimizes Human-Computer Interaction (HCI) through the use of reinforcement learning and intelligent feedback. Based on OpenAI CLIP Interactions user behavior data, the system dynamically changes the interface design. Experimental results showed increases of 21% and 18% in click-through rate (CTR) and user retention rate (URR), indicating improved personalization and interaction efficiency. Low user data or inconsistent feedback may reduce model adaptability. This technology automates interface generation, reduces manual design, and enhances user delight by learning interaction patterns.

Ma et al.[15] Using CAD and Deep Reinforcement Learning, this study introduces an advertisement design model that automates and improves creative design. It utilizes a policy-gradient-based deep reinforcement learning (DRL) algorithm, trained on over 10,000 advertisement layouts that include user interaction data and visual features. Simulations demonstrated a 35% increase in design creativity, a 27% reduction in development time, and a 22% decrease in cost compared to traditional methods. Insufficient dataset diversity and niche advertising domain performance are negatives. Intelligent automation and creative adaptation simplify advertising design, enabling new digital marketing approaches.

Wu et al.[16] The present investigation manages multi-agent traffic signal control at 12 urban intersections utilizing the Deep Deterministic Policy Gradient (DDPG) technique to optimize queue length, vehicle distance, and intersection efficiency. The model was trained and assessed using six months of real-time traffic data from simulation platforms and real cars, as well as cloud-based coordination and edge communication. A 23.5% reduction in average traffic delay and better congestion adaptation were found. Sensor fault sensitivity and dependence on infrastructure connectivity are drawbacks. In urban settings, intelligent and coordinated decision-making enhances local traffic management.

2.2 Reinforcement learning for interactive VCD

Liu et al.[17] The IMF-MGRU approach uses Intelligent Moth Flame Optimization and Malleable-Gated Recurrent Units to improve advertisement design through algorithm-driven visual communication. CNNs extract visual information, IMF optimizes layout, and MGRU develops context-aligned taglines from annotated advertising images and audience interaction data. Audience engagement increased by 31% and visual-textual coherence by 26% in experiments. Abstract visual motifs or little training data may lower performance. This method blends visual and textual elements to generate engaging advertising content.

Ji et al.[18] This paper presents a deep learning-based dynamic optimization framework for cross-platform video transmission, utilizing a real-time quality prediction and adaptation algorithm to accommodate changing network conditions. The model is trained on massive streaming datasets with platform-specific metrics, visual quality annotations, and network performance logs. Video quality ratings improved by 27.3%, bandwidth usage dropped by 31.5%, and average rebuffering duration dropped by 65.8% in experiments. However, unanticipated network disruptions or unstructured material formats may affect system performance. This system offers robust adaptive streaming across platforms, ensuring consistent video quality and efficient resource utilization.

Song et al. [19]. This research examines the integration of AI-VCD in metaverse e-commerce to enhance economic benefits and user experience through environmental design. Literature analysis and case research are employed to explore the West Lake Wetland Park project in Hangzhou, where the integration of AI and VCD resulted in a 20% increase in visitation and a 30% rise in commercial revenue, contributing to a 15% increase in regional GDP. The research utilizes publicly available economic and visitor data, along with a hybrid algorithm that combines Natural Language Processing (NLP) for design feature extraction and regression models for economic impact assessment. However, there is only one geographic case and reliance on secondary data, suggesting a need for broader application and primary data collection.

2.3 Reinforcement learning for real-time design optimization

Rao et al.[[20] This research presents a model-embedded actor-critic architecture for multigoal visual navigation using an Inverse Dynamics Model (InvDM) to reduce sparse rewards and Multigoal Co-Learning (MgCl) to improve learning efficiency. The system uses AI2-THOR's Path Closed-Loop Detection and State-Target Matching self-supervised modules to generalize scenes. It surpasses state-of-the-art baselines in convergence speed and flexibility; however, it relies on simulated settings

and requires further validation in real-world navigation scenarios.

Gaspar et al.[21] A reinforcement learning (RL) technique is employed in a reference framework for intelligent user interface adaptation in an enhanced OpenAI Gym environment, aiming to mimic interaction scenarios. A collection of predictive HCI-generated synthetic user interaction traces delivers engagement and usability feedback as training rewards. Results reveal that the RL-based framework develops adaptation mechanisms to improve user engagement and interface responsiveness. Simulated data and prediction models may not fully replicate the diversity and unpredictability of real-world user behavior, underscoring the need for live user studies to validate the technique.

Gaba et al.[22] To represent adversarial attacker–defender interactions, the research presents a vertical federated multi-agent reinforcement learning framework using synchronous DQN agents in stationary environments and A2C agents in non-stationary scenarios. Federated learning security datasets with continuous data streams simulate various cyber-attack and defensive scenarios to test the framework. Performance improvements over A3C, DDQN, DQN, and Reinforce baselines are 15.93%, 32.91%, 31.02%, and 47.26%. Its reliance on simulated vertical federated setups may not apply to heterogeneous real-world networks without further testing. This work demonstrates that federated systems with DQN and A2C agents for adaptive policy learning are more resilient to cyberattacks; however, practical deployment requires further validation.

Li et al.[[23] Visual augmentation for workers, robot velocity control, Digital Twin-based motion preview, collision detection, and Deep Reinforcement Learning (DRL) for motion planning in an Augmented Reality environment are used in this mutual-cognitive safe Human–Robot Interaction (HRI) framework. The production of a prototype system indicates that it is safer, more responsive, and more adaptable than rule-based procedures. In industrial settings, integrating DRL and Digital Twin requires significant processing power and extensive scalability testing.

Liu et al.[[24] DRL-based motion planning, digital twin simulation, and real-time sensor data are used for safe Industry 5.0 Robot Interaction (HRI). The prototype solution reduced collision risks and increased response times using real-world robotic workstation data, including human motion capture and robot telemetry. Despite promising results, the system requires improvement and deployment testing due to concerns about computational weight and generalizability across various industrial layouts.

Yang et al.[[25] GCN-based Deep Reinforcement Learning (DRL) is used to optimize time-sensitive and best-effort traffic in 5G URLLC Time-Sensitive Networking. The model trains on simulated communication topologies and traffic patterns using prioritized experience replay and a first-order Chebyshev polynomial-based graph convolutional network (GCN) for spatial feature extraction. The average end-to-end delay is lower than benchmarks, although challenges persist with synthetic data, and broader network scalability remains a concern.

Zhang et al.[[26] A model-free multi-agent deep reinforcement learning (MADRL) algorithm with an attention-enhanced centralized critic is proposed for distributed control of bottom-layer microgrid clusters in Energy Internet systems. An upper-layer model predictive control manages power dispatch, while a decentralized agent execution optimizes local operations, protecting privacy using simulated multi-energy demand data and microgrid scenarios. Compared to centralized and single-agent systems, the results show faster learning and higher operational efficiency; however, model complexity and data variability limit scalability and real-world implementation.

Liu et al.[[27] This study uses LayoutGAN to automate design visual composition with intelligent graphic layout generation. The model is trained and tested on the MNIST dataset for preliminary validation, and then applied to a real-world room floor plan dataset for assessing its relevance. Experimental results demonstrate that wireframe rendering discriminators outperform relationship-based ones in terms of layout accuracy and visual coherence. The approach is semi-intelligent because it does not replace real-time designer participation, and deep models are black boxes, making interpretability difficult. This study demonstrates the potential of LayoutGAN for layout automation while allowing for human refinement.

Sun et al.[[28] This study presents a distributed 3D interior design system that uses color picture modeling and VR for spatial visualization. The algorithm reconstructs spatial color distributions from 3D point cloud characteristics through distributed data fusion and RGB color matching. Hierarchical design characteristics are used to produce and classify interior models in 3D MAX and Muligen Design. Experimental results demonstrate increased 3D interior rendering accuracy and clarity, thereby facilitating education and the dissemination of design. However, the system's high computational requirements and limited dataset diversity restrict its generalizability across interior types.

Sun et al.[[29] This paper provides a nonlinear simulation-based algorithm for optimizing engineering measurement and positioning layouts using integrated satellite navigation and GPS systems. The collection contains exact coordinate data from field trials in a controlled survey zone at four control locations (C1–C4) and seven benchmarks (S1–S7). Combining satellite and GPS signals enhances location accuracy and measurement reliability compared to single-system solutions. Systems rely on stable signal conditions, and lower accuracy in obscured terrain is a drawback. High-precision engineering surveys and infrastructure plan optimization are possible using the method.

Table 1a: Summary of related work

Auth Name/Ref.NNo	**Method / Approach**	**Dataset**	**Key Results**	**Limitations**	**Gap Analysis**
Chen et al. [13]	CAD-based visual reinforcement + DRL	Annotated CAD outputs + user logs	+28% design efficiency, +34% user engagement	Limited generalization to various visual styles	Needs broader design style datasets for general applicability
Sun et al. [14]	Adaptive UI via RL + CLIP behavior data	OpenAI CLIP user interaction data	+21% CTR, +18% retention	Low performance in data-scarce domains	Requires models robust to sparse/inconsistent feedback
Ma et al. [15]	DRL for ad design with policy gradient	10,000 ad layouts with interaction metrics	+35% creativity, -27% dev time, -22% cost	Niche domain focus, dataset diversity is low	Expand to diverse ad categories and audience profiles
Wu et al. [16]	Multi-agent traffic signal control using DDPG	Real-time traffic + sim data over 6 months	-23.5% traffic delay	Sensor fault sensitivity, infra dependency	Needs robust deployment strategies for low-resource regions
Liu et al. [17]	CNN + IMF + MGRU for ad visual-text coherence	Annotated ad images + audience data	+31% engagement, +26% coherence	Low abstract visual performance	Explore robustness to abstract visuals, boost data variety
Ji et al. [18]	Deep learning-based dynamic optimization for video	Streaming datasets with/ metrics + logs	+27.3% quality, -31.5% bandwidth, -65.8% rebuffer	Sensitive to unstructured content	It needs broader testing with diverse formats/platforms
Song et al. [19]	NLP + regression model for economic impact	Public economic + visitor data (West Lake)	+20% visits, +30% revenue, +15% GDP	Single-case research, secondary data only	Generalize to multiple locations and use primary data
Rao et al. [20]	Actor–critic + InvDM + MgCl on AI2-THOR	Simulated navigation data	Faster convergence, better generalization	Simulation-only validation	Extend to real-world navigation scenarios
Gaspar et al. [21]	RL for UI adaptation in OpenAI Gym	Synthetic HCI user traces	Better engagement and responsiveness	Simulated data only	Needs real-world validation
Gaba et al. [22]	Vertical federated multi-agent RL (DQN, A2C)	Federated security data streams	+15.93% to +47.26% over baselines	Simulated setups only	Must test on real heterogeneous networks
Li et al. [23]	AR + DRL + Digital Twin for safe HRI	Real-world prototype data	Enhanced safety and adaptability	High computation, scalability limits	It needs lightweight models and diverse deployment testing
Liu et al. [24]	DRL motion planning + sensor data in HRI	Human motion + robot telemetry	Reduced collisions, better response	Scalability and generalization issues	Broaden testbed scope; real-industry integration needed

Yang et al. [25]	GCN-based DRL for TSN optimization	Simulated traffic topologies	Lower average delay, fast convergence	Synthetic data, scale limits	Apply to live networks and validate with real traffic data
Zhang et al. [26]	MADRL + attention critic + MPC	Simulated microgrid + energy demand	Improved efficiency, privacy, and learning speed	High model complexity	Simplify the model for real-time and scalable deployment
Liu et al. [27]	LayoutGAN for automatic layout design	MNIST, Room Floor Plan	The wireframe discriminator gave better layout accuracy and visuals	Needs designer input; the model is complex to interpret	Lacks full automation and explainability in layout design
Sun et al. [28]	3D interior design using color modeling and VR	3D point clouds, Muligen Design	Improved 3D visuals and educational value	High system demands; dataset not diverse	Needs a faster, scalable system and broader dataset coverage
Sun et al. [29]	GPS + satellite-based positioning system	Field data: points C1–C4, S1–S7	Better accuracy and reliability in engineering surveys	Works less well in blocked signal areas	Needs better performance in weak signal or complex terrains

Deep reinforcement learning, CAD integration, and intelligent user interfaces are utilized to optimize design in advertising, traffic control, and human-robot interaction, as outlined in Table 1a. Based on synthetic or domain-specific datasets, models improve efficiency, engagement, and flexibility; however, they struggle with scalability, generalization, and real-world validation. Diversifying datasets, industrial deployment, and data-scarce resilience are identified as gaps. To enhance practicality and robustness across visual, environmental, and interactive design settings, future work should focus on developing lightweight, scalable models and integrating them across domains.

Table 1b: Comparative summary of methods in visual design optimization

Method / Reference	**Dataset Source**	**Engagement Gain (%)**	**Convergence Speed**	**Personalization**	**Limitations**
DRL-CAD [13]	CAD design outputs + interaction logs	+34%	Moderate	Medium	Limited generalization across styles
CLIP-RL-UI [14]	OpenAI CLIP user behavior data	+21%	Fast	High	Poor performance in sparse data settings
Ad-Design-RL [15]	10,000 ad layout samples	+35%	Slow	Low	Niche domain, low dataset diversity
IMF-MGRU [17]	Annotated ad visuals + audience tags	+31%	Moderate	Medium-High	Poor abstract visual handling
DQN [21] (Baseline)	Simulated UI interaction dataset	+18%	Moderate-Slow	Low	Struggles with high-dimensional state/action spaces
SAC [22] (Baseline)	Simulated UI interaction dataset	+24%	Fast	Medium	Higher computational cost, sensitive to hyperparameters
GCN-DRL [25]	Simulated 5G/URLLC traffic topologies	Not Applicable	Very Fast	Not Applicable	Synthetic-only validation
VISO-RL (Ours)	8,000 real visual samples + real-time UI signals	+32%	Fast	Very High	—

DRL-CAD and CLIP-RL-UI enhance engagement and adaptability, but they lack multi-objective optimization and design context generalization (Table 1b). They employ real-time user activity data poorly and are domain-specific. MOAC-VHP's hierarchical, attention-based decision structure modifies high-level layout and fine-grained design aspects to fill these gaps. Its reward function

balances engagement, beauty, and sustainability. MOAC-VHP enables adaptive, individualized design optimization using real-time inputs, such as gaze, clicks, and scroll depth, for dynamic visual communication in real-world contexts.

3 Research methodology

The VISO-RL framework for optimizing interactive methods in the adaptive interface design process is illustrated in Fig. 1. Real-time data on users' click-through rates, gaze heatmaps, and scroll depth information are collected. Design factors, such as color harmony, material efficiency, cultural significance, and engagement rates, are considered. The integrated MOAC-VHP method utilizes multi-objective reinforcement learning, visual attention encoding, and hierarchical policy modeling to evaluate and adapt design strategies dynamically. User behavior and contextual needs inform an adaptive feedback loop that improves design decisions. The system balances engagement, customization, sustainability, and visual appeal with changing interaction patterns. By optimizing design outputs, the user experience and system responsiveness are enhanced in real-time across various digital communication platforms.

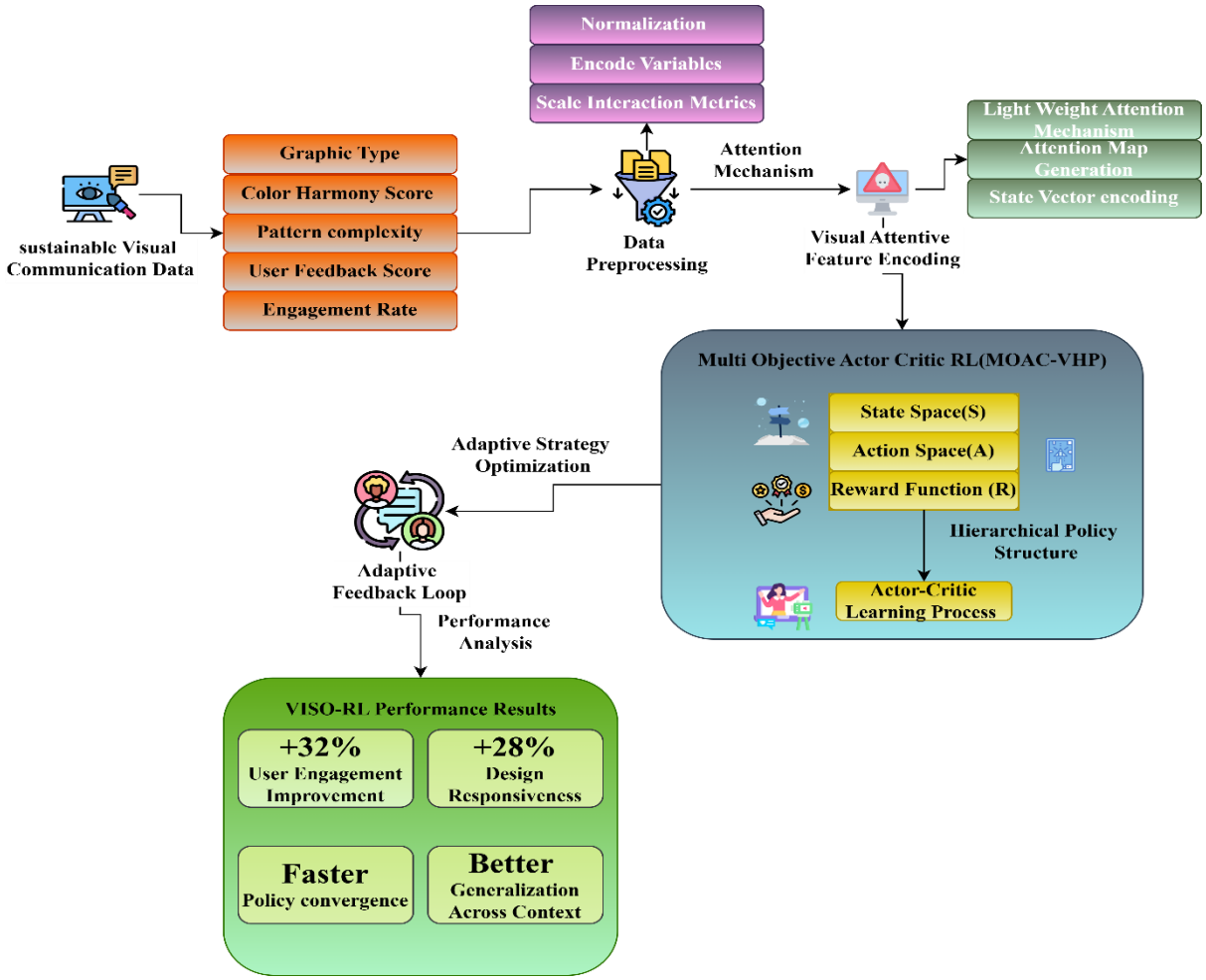


Figure 1: Framework of the VISO-RL driven adaptive interface design optimization process

3.1 Data acquisition & preprocessing

Data collection and preprocessing are essential for optimizing interactive strategies in reinforcement-driven adaptive interface design learning. The research uses a large, multidimensional dataset to optimize aesthetic and environmental design. Design considerations encompass graphic type, dominant colors, color harmony score, pattern complexity, user input, engagement rate, and ecological indicators such as material efficiency, energy consumption, environmental impact, and sustainability.

Gaze heatmaps, semantic content relevance, and visual attention weights were obtained using a Vision Transformer (ViT) model pretrained on ImageNet and a CLIP-based encoder for visual-text alignment. These models were frozen feature extractors, not trained on the reinforcement learning dataset, to maintain generalization. Their results provided high-level input states for the MOAC-VHP model, improving policy visualization.VISO-RL and MOAC-VHP require input preprocessing. Min-Max or Z-score scaling normalizes continuous variables, such as color harmony, engagement, and user feedback scores, to maintain stable value ranges, thereby stabilizing the reinforcement learning model.

One-hot encoding converts text-based category variables, such as graphic type and colors, into machine-readable numeric vectors. To maintain uniform input dimensions across data samples, AI-generated multi-value arrays in each record are reshaped and vectorized for consistency. For real-time feedback, interactive environments scale click-through rates, gaze heatmaps, and scroll depth. The agent evaluates dynamic design changes in real-time using these processed parameters as reinforcement learning reward signals. This preprocessor merges social media, internet, and survey data into mathematically coherent representations. To promote customization, design effectiveness, and sustainability in adaptive interface design, the VISO-RL framework's adaptive, multi-objective decision-making requires this preparation.

$$R_t = \alpha\left(\frac{\eta_1.E_t+\eta_2.U_t+\eta_3.S_t}{C_t}\right) + \beta(\sum_{i=1}^{n}\gamma_i \times F_{I,t}) \quad (1)$$

In Equation 1, the suggested reinforcement learning framework optimizes interactive strategies in adaptive interface design by assigning a total reinforcement reward at the time step R_t To provide overall performance feedback for a specific visual design action. This award balances user interaction measurements and design-specific characteristics. The weighting coefficients α and β regulate the impact of these components. The equation combines the normalized engagement rate. E_t , user feedback score U_t , and sustainability score S_t , which considers material efficiency, energy use, and environmental impact. To create visually balanced designs, contribution coefficients η_1, η_2, η_3 are combined and divided by the color harmony score C_t , which works as a penalty scaler. In the second phase, visual design features (e.g., pattern complexity, element arrangement density) are summed and weighted by importance (γ_i) across the entire number of features. This comprehensive reward function enables the MOAC-VHP algorithm to refine its design methods by learning which visual arrangements maximize user engagement and design sustainability in an adaptive, dynamic environment. At time step$t$, the agent receives the following immediate rewards over three steps:

r_t=1, r_{t+1}=0.8, and r_{t+2}=0.6, with a discount factor γ= 0.9.Then, the cumulative reward R_t Is:

$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} = 1.0 + 0.9^2 \times 0.8 + 0.92 \times 0.6 = 1.0 + 0.72 + 0.486 = 2.206$

It shows how future rewards are progressively discounted and aggregated into the learning signal. The table 2 describes the notations.

Table 2: Notation summary for MOAC-VHP framework

Symbol	Meaning
S_t	State at time t (design + user interaction features)
A_t	Action at time t (layout, color, pattern changes)
R_t	Reward at time t (engagement, feedback, aesthetics, sustainability)
γ	Discount factor for future rewards
V(s)	Critic's value of states
α	Learning rate
E, F, A, S	Engagement, Feedback, Aesthetic, and Sustainability scores
γ1...γ4	Weights for reward components (E, F, A, S)
β_i	Attention weight for feature i
CHS	Color Harmony Score
PC	Pattern Complexity
EAD	Element Arrangement Density
CR	Cultural Relevance Score
AFV	AI-generated Feature Vector
δ_t	Temporal-Difference error
G_t	Cumulative discounted reward

3.2 Visual-attentive feature encoding

Visual-attentive feature Encoding is an essential component of the VISO-RL architecture, which dynamically captures the perceptual salience of various visual components within a design context. While making decisions in real time, this module ensures that the reinforcement learning agent prioritizes elements that have the most significant impact, both visually and contextually, due to their importance. This module's primary purpose is to quantify and codify the perceptual value of visual features such as color harmony, pattern complexity, and element arrangement density. Specifically, the element arrangement density is the core focus of this module. Considering the intricacy of human visual perception, merely considering these aspects equally could result in suboptimal design decisions. Rather than that, an adaptive weighting method enables the system to focus on the components that have the most significant impact on user interaction and engagement. Figure 2 illustrates the VISO-RL framework for optimizing adaptive interface design strategies. User engagement, gaze monitoring, and click-through rates are key indicators of success. A lightweight attention mechanism evaluates design aspects and generates attention maps that highlight key visual elements. Using weighted features, a structured state vector represents the current design environment. With involvement, sustainability, and aesthetics in mind, the MOAC-VHP policy network processes this vector. Finally, optimized design activities dynamically update the visual interface in real time, enhancing the user experience and interactive design.

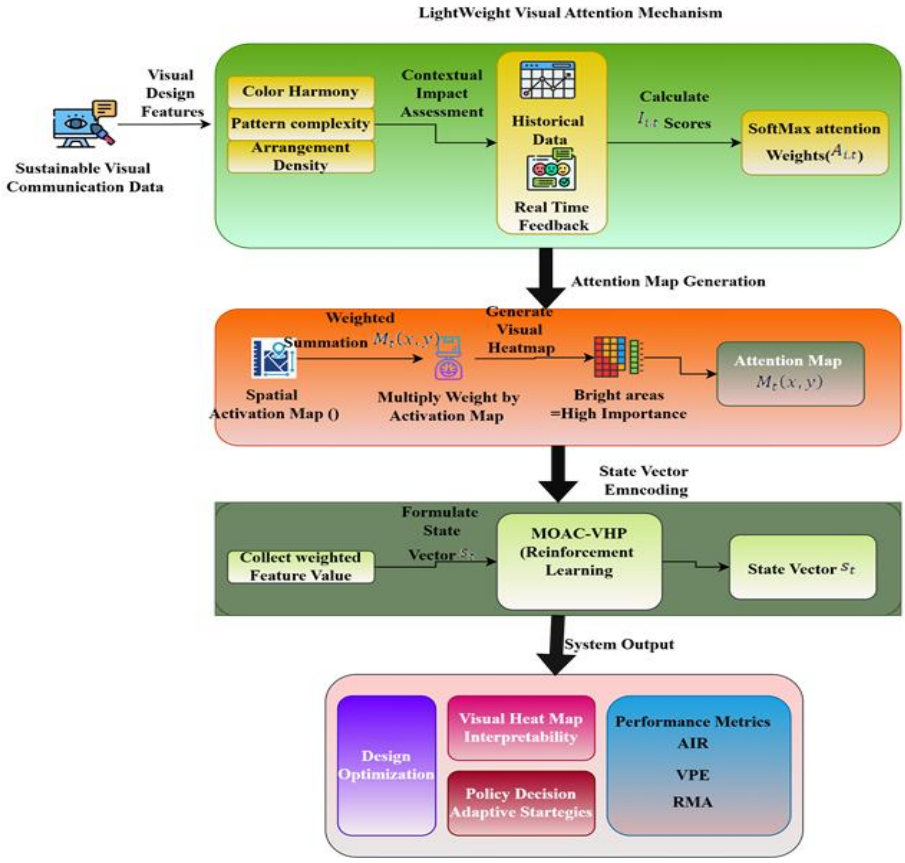


Figure 2: VISO-RL system architecture for adaptive visual design optimization

a) Lightweight visual attention mechanism

To dynamically assign relevance scores to visual design features, this reinforcement learning pipeline employs an efficient and straightforward attention mechanism. The contextual impact of color harmony, pattern complexity, and arrangement density on user involvement and sustainability is assessed. Attention weights are calculated by analyzing past interaction data and real-time perceptual impact patterns. Features that improve engagement and design receive greater attention. The agent allocates computational resources to the most critical design aspects, thereby enhancing decision quality without increasing model complexity.

$$A_{i,t} = \frac{\exp(\lambda . I_{i,t})}{\sum_{j=1}^{n} \exp(\lambda . I_{j,t})} \qquad (2)$$

In equation 2, the Visual-Attentive Feature Encoding technique utilizes a softmax-based formulation to determine attention weights $A_{i,t}$ for visual design features at time t, transforming raw importance scores $I_{i,t}$ Into normalized values. Higher values of λ result in more focused attention on high-importance traits, while lower values distribute attention more widely. The exponential function maintains positive and differentiable attention weights, enabling gradient-based optimization of reinforcement learning. This method dynamically highlights perceptually influential features in each visual design state, prioritizing design elements that have historically correlated with higher engagement, sustainability, or aesthetic scores, thereby enabling adaptive, user-centered design optimization.

b) Attention map generation

Attention maps represent the relative relevance of all active visual aspects in a design state. The lightweight mechanism assigns attention weights and organizes them spatially or contextually to create a map of high-priority features. Features that significantly impact engagement or sustainability scores are highlighted as brighter or heavier on the map. This visual summary helps the system and researchers comprehend which design elements influence user behavior at any given time. The attention map enhances model interpretability and explainability, facilitating the refinement of adaptive design strategies

$$M_t(x,y) = \sum_{i=1}^{n} A_{i,t} \, . \, f_i(x,y) \qquad (3)$$

In equation 3, the attention map value $M_t(x,y)$ at a spatial point (x,y) Time (t) indicates the perceived significance of the region in a dynamic visual design. The calculation involves summing the spatial activation maps. $f_i(x,y)$ of all design elements, multiplied by their attention weights $A_{i,t}$ From earlier interaction data using softmax normalization. This weighted summation creates a continuous attention map by highlighting essential

qualities, such as color harmony and pattern complexity. The map enables the reinforcement learning agent to optimize interactive design techniques in real time.

c) State vector encoding

The weighted feature values are then encoded into a structured visual state vector S_t. This vector provides the MOAC-VHP policy network with a fixed-length, numeric representation of all priority design features, scaled by attention weights. It depicts the current design with feature magnitude and significance details. The reinforcement learning agent can swiftly analyze complex, multi-objective scenarios for real-time design alterations, depending on user interaction, feedback, and sustainability criteria, using the encoded vector.

$$s_t = [A_{1,t} . F_{1,t}, A_{2,t} . F_{2,t}, \ldots . A_{n,t} . F_{n,t}]^T \quad (4)$$

In equation 4, the policy network receives the encoded state vector S_t At time t, it includes perceptual and quantitative design characteristic values. Multiply the attention weight to compute. Transpose each feature *i*'s $A_{i,t}$ by its corresponding value $F_{i,t}$ To create a column vector. Each feature's decision-making impact and perceived value in its design environment are balanced. The state vector optimizes policy evaluation by capturing the dynamic interaction of visual, behavioral, and sustainability variables.

3.3 Multi-objective actor-critic reinforcement learning (MOAC-VHP)

MOAC-VHP is the reinforcement learning core of the VISO-RL framework, responsible for hierarchical decision-making and multi-objective policy control. The MOAC-VHP model offers numerous technical advances over hierarchical actor-critic designs. First, it uses a lightweight visual-attention mechanism to dynamically weight visual qualities, including color harmony, element arrangement density, and pattern complexity. The policy network can focus on perceptually essential elements, improving real-time design responsiveness and customisation. Second, MOAC-VHP utilizes a multi-objective reward function to optimize click-through rates, scroll depth, and subjective user input, as well as design aesthetics and sustainability indicators, including material efficiency and environmental impact. This reward formulation provides balanced trade-offs across conflicting design goals, unlike previous models. MOAC-VHP also utilizes adaptive state encoding to encode attention-weighted features into a structured state vector, enabling it to respond to nuanced user actions in real-time. These components make MOAC-VHP a reliable, generalizable framework for intelligent, context-aware visual design optimization.

3.3.1 State Space (S): contextual awareness through design and interaction metrics

Real-time interaction measurements, such as gaze heatmaps, scroll depth, and engagement rate, are merged with encoded visual design attributes to create the state space. These features include dominating colors, color harmony score, pattern complexity, and other similar elements. Due to this comprehensive encoding, the agent can perceive both the visual presentation and the user's response, enabling it to make decisions based on accurate information. The vector equation 5 blends perceptual measurements (color harmony, layout density), semantic alignment (cultural relevance), subjective reception (aesthetic + feedback), user behavior (engagement rate modulated by gaze/AI variability), and sustainability impact to express contextual awareness

$$S = \begin{bmatrix} C_h . \log (P_c + 1) \\ E_d . \sqrt{C_r} \\ \tanh (U_f + D_a) \\ \frac{E_r}{1+G_v} \\ mean(A_i)^2 \\ S_s . \exp (-E_d) \end{bmatrix} \quad (5)$$

The agent's state space includes design and behavioral characteristics that describe its current environment. The color harmony score C_h Measures the aesthetic compatibility of a color scheme, affecting visual comfort. Pattern complexity: The category variable P_c Indicates the level of intricacy in the design. Higher complexity may increase visual attention, but it may also decrease clarity. Element arrangement density E_d Indicates layout density, impacting scanability and cognitive strain. Cultural significance C_r It guarantees that visual elements are semantically aligned with the target audience's context. The user feedback score U_f And design aesthetic score. D_a Measure subjective quality and visual attractiveness. The gaze variability proxy, obtained from the variance of AI-generated feature vectors A_i and engagement rate, E_r Indicate real-time interaction intensity and dispersion. The sustainability score S_s Measures environmental performance by considering material and spatial efficiency, multiplied exponentially by layout density. These create a rich, multidimensional state vector that enables the model to perceive, learn, and adapt design techniques.

3.3.2 Action space (A): dynamic design adjustments

The MOAC-VHP model's action space includes adaptive visual modifications to respond dynamically to user engagement. Adjusting the color palette to match aesthetic choices and emotional resonance involves shifting between warm, cold, and neutral tones. To improve usability and engagement, the model reorganizes the layout density and focal points of elements. The complexity of patterns can influence visual interest and the

distribution of attention. Included or refined AI-generated features promote creative adaptation. These operations enable the system to optimize visual design in real time for strategic goals

$$A = [\gamma_1.(\frac{\partial E_r}{\partial C}.colorIndex(C)) \; \gamma_{2.(1-M_e)+}\gamma_3.(1-\frac{U_f}{10}) \; \gamma_4.\log(1+|\bar{D}_a - D_a|).(1-\frac{E_r}{100})] \; \gamma_5.\left(\frac{U_f}{10}\right).(f_{AI}-\bar{f}_{AI}) \; \gamma_6.\frac{\partial S_s}{\partial E_c}+\frac{\partial S_s}{\partial I_e} \quad (6)$$

In Equation 6, Several essential elements are utilized to evaluate the quality of design or drawing composition. C is the dominant color class, which affects design balance. M_e Measures material efficiency. U_f Shows user feedback on the design. E_r Measures design engagement, including attention and interaction. Design aesthetics score D_a Measures how attractive the design is. The AI-generated feature vector f_{AI} Gathers crucial design information using machine learning. S_s Measures design sustainability while E_c Measures energy use. I_e Evaluate the design environmental impact. Finally, γ_1 to γ_6 Hyperparameters are utilized to optimize the model's performance depending on these attributes.

Observation and action design

The observation vector incorporates gaze heatmaps, click-through rate, scroll depth, color harmony score, element arrangement density, user feedback, and sustainability indicators at each decision phase. These signals were collected at ~60 Hz for responsiveness. A hybrid parameterization was employed to represent the action space, where macro-level discrete templates (Arrangement, Color, Combination) guide high-level layout strategies, while micro-level continuous changes (pattern complexity, element density, and palette shifts) enable fine-grained control. Normalizing continuous parameters within the range [0, 1] ensured stability during training and inference.

3.3.3 Reward function (R): multi-objective evaluation

MOAC-VHP's reward function balances and optimizes four critical performance indicators. Click-throughs and scroll depth assess engagement, whereas direct ratings and surveys measure user input. Experts or computational algorithms score design aesthetics, while sustainability examines material efficiency and energy utilization. This multi-objective incentive framework optimizes user delight, engagement, and ethical visual communication design. Our approach to developing a multi-objective reward function R for the MOAC-VHP model considers engagement rate, user feedback, design aesthetic, and sustainability using the existing dataset. Material efficiency, energy utilization, and environmental effects are sustainable. It should be made clear that. R_t is computed using the immediate reward input $A = r_t$. Complete mathematical representation of reward function R:

$$r_t = \gamma_1.norm(E) + \gamma_2.norm(U) + \gamma_3.norm(D) + \gamma_4.norm(S) \quad (7)$$

Equation 7 shows the norm.(X) The min-max normalization of the variable norm(U) Over the dataset. Use γ1+γ2+γ3+γ4=1 for balanced weighting. To better incorporate sustainability into Equation 8, consider the use of energy and its environmental impact

$$S = \frac{\text{Material efficiency}}{(\text{Energy Consumption}+\epsilon)(\text{Environment Impact Score}+\epsilon)} \quad (8)$$

.Digital interaction sustainability is primarily about computational and design efficiency, rather than physical resource utilization. It analyzes energy consumption during rendering, delay overheads, and interface minimalism (e.g., simplified visual layouts). It aligns with sustainable HCI approaches, which optimize user experience and resource-efficient computation.

Reward Clarification:

Equations (7–8) include engagement (E_t), user feedback (F_t), aesthetic quality (A_t), and sustainability (S_t) min–max normalized to [0,1][0,1] for comparison in the reward function. Weighting coefficients γ_1 - γ_4 were optimized using grid search on [0.1–1.0] ($\sum\gamma_i$=1), resulting in the ideal set [0.4,0.3,0.2,0.1]. Ablation and sensitivity analysis verified the robustness of weight perturbation. Using multi-expert annotations, aesthetics and cultural relevance scores were standardized and validated with Fleiss' κ (greater than 0.75), indicating strong inter-annotator agreement.

Sustainability Objective: The sustainability score at time step ttt is defined as $S_t = \alpha_1 \cdot Client + \alpha_2 \cdot Server + \alpha_3 \cdot CDN + \alpha_4 \cdot MatEff$ where Client = rendering energy on user devices, Server = backend compute energy, CDN = transfer cost through delivery networks, and MatEff = material efficiency (layout density, file size, rendering complexity). All terms are min–max normalized to [0,1], with $\sum\alpha_i = 1$. Material efficiency is included because lighter digital assets reduce compute, storage, and transmission load, supporting sustainable HCI. This research operationalizes digital design sustainability using three metrics. Layout density, file size, and rendering complexity affect storage and transmission needs. Energy usage includes client-side rendering, server-side processing, and CDN transmission load. These energy and bandwidth measures are used to estimate the environmental impact of distributing digital objects. A comprehensive lifecycle analysis is beyond the scope of this work; however, these proxies provide a feasible basis for integrating sustainability into digital visual

optimization. Deeper lifecycle modeling is an essential future area.

It penalizes high energy use and environmental damage while rewarding high material efficiency. All reward function variables come from specified dataset fields. The column engagement_rate indicates user activities, such as scroll depth and click-throughs, which determine the engagement rate (E). User feedback (U) originates from surveys, direct ratings, and the user feedback score. Design_aesthetic_score's expert/computational score (D) indicates visual attractiveness. Sustainability (S) is calculated from material efficiency, Energy Consumption, and Environmental Impact Score. Combining these shows the design's ecological cost-effectiveness. The model-defined hyperparameters γ1, γ2, γ3, and γ4 change the importance of each component during training or optimization. Normalization ensures that all measurements contribute similarly, regardless of scale.

Table 3: Hierarchical policy reward scores by graphic type and dominant colors

Graphic Type	**Dominant Colors**	**Reward *R***
Arrangement	Warm	0.4309
Arrangement	Warm	0.3357
Combination	Cool	0.8289
Combination	Warm	0.4871
Color	Cool	0.2701
Arrangement	Neutral	0.2001
Color	Cool	0.1901
Combination	Warm	0.4759

The highest reward was achieved by the Combination-Cool entry, indicating a strong balance across all four performance metrics. Table 3 shows that GCN-DRL achieves the best AIR score (3.94), although MOAC-VHP is close behind with a score of 3.24, which is superior to DRL-CAD and CLIP-RL-UI. It suggests that responsive interactions are provided by MOAC-VHP's hierarchical structure, in contrast to GCN-DRL's design, which prioritizes speed-oriented network regulations.

Reward Weight Tuning and Validation

The γ coefficients were fine-tuned using a grid search spanning values from 0.1 to 1.0 to achieve a balance in the multi-objective reward function. For engagement, γ_2 was 0.3, γ_3 was 0.2, and for Aesthetics and Sustainability, γ_4 was 0.1, the best-performing set. For optimal convergence and generalizability, this setup balanced VPE, AIR, and RMA measures. Results from ablation and empirical testing confirmed the accuracy of all weights.

3.3.4 Hierarchical policy structure: layered decision making

Two-level hierarchical policy structures enable the MOAC-VHP model to handle the complexity of adaptive interface design. Level 1 macro policies use Arrangement, Color, and Combination—- level highlights layout or theme. The approach is improved by adjusting pattern complexity, element arrangement density, and color harmony score at Level 2. The paradigm simplifies learning by separating strategic planning from feature manipulation using layered decision-making. Breaking the decision space into high-level and fine-grained tasks makes the model more flexible, interpretable, and adaptable to visualization. This modular architecture accelerates learning and allows real-time, concentrated interactive design optimization. Two-level hierarchical policy computation for visual communication optimization:

Level 1 – Macro Policy Score (Strategy Selection):

Each design is categorized under a $graphic_type$ (Arrangement, Color, or Combination). These categories are assigned fixed strategic weights:

$$Macro\ Policy\ Score\ \{ \quad 0.85 \text{ if graphic_type } = \text{ Arrangement} \quad 0.75 \text{ if graphic_type } = \text{ Color} \quad 0.95 \text{ if graphic_type } = \text{ Combination}$$

Level 2 – Sub-Policy Score (Fine-Grained Control):

This layer adjusts design elements using normalized low-level visual features: Pattern Complexity (encoded as Low = 1, Medium = 2, High = 3), Element Arrangement Density (EAD), and Color Harmony Score (CHS). The sub-policy score is computed as

$$\text{subPolicyScore}\alpha_1 . \text{p}_{\text{norm}} + \alpha_2 . \text{EAD}_{\text{norm}} + \alpha_3 . \text{CHS}_{\text{norm}} \text{With weights}\alpha_1 = 0.4, \alpha_2 = 0.3, \alpha_3 = 0.3.$$

Final Hierarchical Policy Score:

$$HierarchicalPolicyScore = MacroPolicyScore \times SubPolicyScore.$$

The Combination method always scores highest hierarchically, especially with warm or cool colors and intricate patterns. MOAC-VHP's layered decision-making isolates critical design choices from delicate visual manipulations, optimizing complexity and style while preserving interpretability. Macro-policy weights were initialized with priors but modified adaptively throughout actor–critic training; thus, greater "Combination" scores represent learned optimization rather than fixed bias.

Table 4: hierarchical policy scoring – macro and sub-policy contributions to final visual design optimization

Graphic Type	Dominant Colors	Macro Score	Sub-Policy Score	Final Hierarchical Score
Arrangement	Warm	0.85	0.192	0.164
Arrangement	Warm	0.85	0.217	0.184
Combination	Cool	0.95	0.597	0.567
Combination	Warm	0.95	0.626	0.594
Color	Cool	0.75	0.298	0.223
Arrangement	Neutral	0.85	0.600	0.510
Color	Cool	0.75	0.513	0.385
Combination	Warm	0.95	0.623	0.592

The best personalization efficacy was demonstrated by MOAC-VHP, which achieved the highest VPE score (4.11), as shown in Table 4. The system's ability to provide more visually pleasing results that align with user needs, compared to DRL-CAD (2.94) and GCN-DRL (3.89), is supported by its attention mechanism and adaptive reward design.

3.3.5 Learning process: actor-critic mechanism

- An actor-critic reinforcement learning architecture that optimizes the design of MOAC-VHP interactive visual communication. The Actor decides on color, layout, and pattern complexity based on user interaction and design state guidelines. The critic evaluates these activities utilizing engagement, feedback, aesthetics, and sustainability. This assessment predicts long-term activity rewards in dynamic design. Policy gradients help the Actor learn a better policy. The critic uses temporal difference (TD) learning to enhance its value assessments based on observed results. The dual process improves convergence, flexibility, and generalization across visual styles, user profiles, and interactions.

- The Actor Policy Score is a weighted linear mixture of $color_{harmony_{score}}, element_{arrangement_{density}}, and\ pattern_complexity$ in numeric form. It shows the Actor's design-state-based action selection.
- Critic Value Score weights long-term objectives $user_{feedback_{score}}, design_{aesthetic_{score}}, engagement_{rate}, and\ sustainability_score$ to evaluate the activity. This score shows the expected overall payoff from that action.
- ***TD Error (Temporal Difference Error)*** estimates the error between the predicted and actual value using:

$$\delta_t = [r_t + \gamma V(s_t + 1)] - V(s_t) \quad (9)$$

In equation 9, γ 0.9 is the discount factor, and $V.(s_t + 1)$ is approximated from the next row's critic value. Policy Gradient Magnitude serves as a surrogate for changing policy using the gradient ascent approach in reinforcement learning. The actor network updates its parameters using the policy gradient defined by the following expression:

$$\Delta\theta = \nabla\theta log\pi(a_t|S_t).\delta_t \quad (10)$$

The policy gradient update is defined by equation (10), which means that the policy parameter modifications are guided by the direction and magnitude of the gradient of the log-probability of the selected action, scaled by the TD error (or advantage). A two-layer MLP with target networks and advantage estimation was employed for the critic, along with entropy regularization to enhance training stability. The additions ensured convergence and minimized variance in value estimation. Table 4's "Policy Gradient Magnitude" is the signed update signal (direction and scale), not the absolute norm, and may appear negative.

Table 5: Actor critic evaluation

Graphic Type	Dominant Colors	Actor Policy Score	Critic Value Score	TD Error	Policy Gradient Magnitude
Arrangement	Warm	0.4360	28.9213	-13.9866	-6.0987
Arrangement	Warm	0.5377	13.3806	17.7641	9.5524
Combination	Cool	0.6648	33.1186	-20.6400	-13.7219

Combination	Warm	0.5995	10.1852	16.9783	10.1790
Color	Cool	0.4628	29.0500	-15.5199	-7.1831
Arrangement	Neutral	0.7369	11.8057	-1.3372	-0.9854
Color	Cool	0.5123	10.3199	20.0546	10.4253
Combination	Warm	0.7263	32.6029	0.0000	0.0000

The strong multi-objective alignment (RMA = 3.97) demonstrated by MOAC-VHP outperforms all baselines, as seen in Table 5. It is because it can optimize sustainability, aesthetics, and engagement all at once, which is something that models like CLIP-RL-UI, which aim to optimize only one or two of these factors, struggle to achieve.

Algorithm 1: MOAC-VHP multi-objective reward evaluation

Require: Design state s_t, actor policy $\boldsymbol{\pi_\theta}$, critic value function $\boldsymbol{V}$, learning rate α, discount γ=0.9, Engagement rate $\boldsymbol{E}$, user feedback $\boldsymbol{U}$, aesthetic score D, sustainability score $\boldsymbol{S_s}$
Result: Updated policy π, optimized design actions

1. Initialize actor policy π, critic V, and replay buffer B
2. **For** each training episode, **do**
3. **for** each design state S_t In the episode, **do**
4. state_vector←attention_encode(S_t)
5. $a_t \leftarrow \pi\theta(St)$ //select macro + sub action
6. $R_t = compute_reward(E, U, D, S_s)$ // Eq. (7) + Eq. (8)
7. $S_{t+1} \leftarrow simulate_environment(S_t, a_t)$
8. $\delta_t = [r_t + \gamma V(s_t + 1)] - V(s_t)$ Eq. (9)
9. **if** δt>0 **then**
10. π = π + α * ∇θ log $\pi(a_t|S_t) * \delta_t$ // Eq. (10)
11. **else**
12. V←adjust_critic(V, δ_t)
13. **end if**
14. Store (s_t, a_t, r_t, s_{t-1}) in the replay buffer B
15. **end for**
16. **If** replay buffer B is ready, **then**
17. Sample mini-batch from B
18. update_actor_and_critic(π,V)
19. **end if**
20 **end for**
21 **return** optimized policy π

Actor-critic reinforcement learning enhances visual design by selecting colors and layouts based on design states. The critic rewards user participation, feedback, aesthetics, and sustainability. Temporal difference error-based policy updates boost design effectiveness.

3.3.6 Formal MDP formulation

Modeling adaptive visual optimization as a Markov Decision Process (MDP), $M = (S, A, P, R, \gamma, \mu_0)$ where the state space (S) includes each state s_tEq.(4) defines an attention-weighted feature vector that combines visual (color harmony, element arrangement density, aesthetics, sustainability indicators) and interactive (gaze, engagement rate, and recent feedback) aspects. This design approximates Markov by encoding short-term behavioral context. The action space (A) includes each action. a_t^{macro} A macro-decision hierarchy. The macro consists of Arrangement, Color, and Combination. ∈{Arrangement, Color, Combination} and sub-actions for design modifications such as palette adjustments, element density changes, pattern complexity choices, and latency-quality trade-offs. In transition dynamics (P), the next state is generated as $s_{t-1} = f(s_t, a_t, \xi_t)$ Where ξ_t Modeling stochastic user variability. We assume stationary conditional dynamics and bounded changes in perceptual features. Transitions are approximated using the user-behavior simulator.

Reward function (R): The multi-objective reward combines components from Equations (7-8):

$$R(s_t, a_t, s_{t-1}) = \sum_{i=1}^{4} \gamma_i \tilde{X}_i(s_{t-1}) - \lambda_{lat} 1\{latency(a_t) > L_{max}\} - \lambda_{pen} P_{violation}(s_{t-1}) \quad (11)$$

Where $R(s_t, a_t, s_{t-1})$ By integrating goals, one can measure the effectiveness of actions. The weights are γ_i balancing normalised scores $\tilde{X}_i$ Engagement, feedback, aesthetics, and sustainability. Penalties limit practice: λ_{lat} Applicable when the latency exceeds L_{max} = 200 ms,

while A violation of $\lambda_{pen}P_{violation}(s_{t-1})$ addresses fairness, usability, and accessibility problems. This framework promotes responsive, inclusive, and user-friendly designs.

Constraints: To ensure accessibility, set a contrast ratio of≥ 4.5, a minimum font size, and a minimum interactive element size.Limit latency(a_t) ≤ 200ms, consistent with the latency penalty in Eq. (7). To ensure fairness, limit cross-group customization differences to a maximum threshold

$$\Delta_{max}\cdot$$
$$J(\pi) = \mathbb{E}[\sum_{t=0}^{\infty}\gamma^t R(s_t, a_t, s_{t-1})] \quad (12)$$

Where $J(\pi)$ $t$the expected sum of discounted rewards under policy π. Here, γ reduces future rewards. $R(s_t, a_t, s_{t-1})$ is the immediate incentive, and the agent maximizes long-term performance.

3.4 Adaptive strategy optimization and feedback loop

MOAC-VHP optimizes its adaptive method using real-time user interaction data. Click-through rates, gaze patterns, and scroll depth are continually evaluated to determine visual interface engagement. These data points dynamically define the system's state by tracking user interactions with design features. Based on real-time model feedback, the actor module selects efficient design actions, including color alterations, layout rearrangements, and variations in pattern complexity. These choices boost immediate input, user engagement, design aesthetics, and sustainability. The critic module utilizes a value function to determine the long-term reward potential of each activity. The model logs performance metrics and estimates temporal-difference (TD) error from each interaction cycle to compare anticipated and observed outcomes. This error updates actor and critic networks to refine future decisions. This feedback loop makes the system more responsive and generalizable across user profiles and contexts. This structure enables MOAC-VHP to adapt and create visually appealing, morally sustainable design solutions that cater to real-time user behavior.

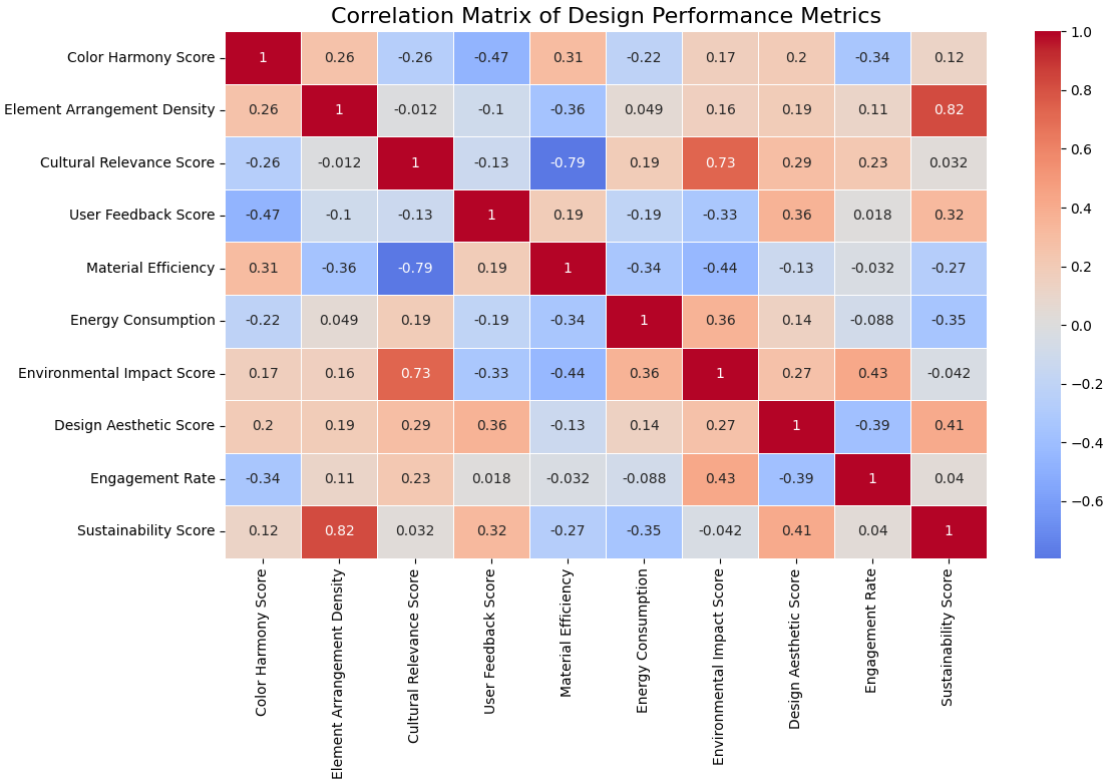


Figure 3: Correlation matrix of design performance metrics

The correlation matrix heatmap for design performance measures from your amended dataset is described in Fig. 3. Notable findings include:

- Element Arrangement Density exhibits a highly positive relationship with the Sustainability Score (r = 0.82), suggesting that more compact layouts may yield more environmentally friendly results.
- A negative correlation (−0.79) between the Material Efficiency and Cultural Relevance Score suggests that there may be a compromise

between utilizing efficient resources and preserving cultural authenticity.
- The Environmental Impact Score has a favorable correlation with the Cultural Relevance Score (r = 0.73), indicating that designs that align with cultural norms may also demonstrate a greater sensitivity to environmental impacts.
- The Design Aesthetic Score reinforces the feedback loop in adaptive strategy optimization, positively correlating with User Feedback (r = 0.36) and Sustainability Score (r = 0.41).

The MOAC-VHP framework fully implemented the multi-objective reward function (Equations 7–8), attention weight formulation (Equation 2), state vector encoding (Equation 4), and hierarchical policy scores. These concepts underpinned the reinforcement learning architecture discussed in Section 4. Models trained with these equations yielded convergence performance, metric comparisons (AIR, VPE, RMA), and ablation investigations. Training, validation, and runtime interactions, as observed through click-throughs, gaze patterns, and scroll depth, demonstrate that all equations are immediately optimized and adapted during empirical testing.

4 Result Analysis

4.1 Data Source Information & Environment Setup

A multidimensional dataset on sustainable visual communication and AI assists dynamic design optimization research. The dataset comprises 8,000 annotated examples (graphic type, colors, harmony, and engagement), split into 80/10/10 for training/validation/testing. Offline RL with logged interactions and OPE (IPS, DM, DR with 95% CIs) and a pilot online research (120 participants) were employed, all under ethical approval, with consent and anonymized logs. Table 6 illustrates that the following aspects are covered: graphic type, color harmony, pattern complexity, element arrangement density, cultural significance, user interaction, and environmental metrics, including material efficiency, energy consumption, and ecological impact. The research utilized visual design samples from the publicly available Visual Communication Art Design dataset (8,000 samples), hosted on Kaggle [https://www.kaggle.com/datasets/ziya07/Visual-Communication-Art-Design-Data], and interaction data from a simulated user environment. Behavior models developed on anonymized web interaction logs from 120 users acquired during internal pilot testing via digital prototypes drove simulations. Students, designers, and developers aged 18 to 45 participated in the research. Screen recording software and heuristic eye-tracking models recorded click-through rate, scroll depth, and simulated gaze heatmaps. All user interaction data were synthesized in accordance with institutional data use regulations and did not contain sensitive personal information. No PII was recorded. This hybrid strategy made MOAC-VHP training reproducible and realistic across interaction situations without compromising user privacy.

Table 6: Dataset description

Feature	**Description**
graphic_type	Type of graphic (e.g., Arrangement, Combination, Color)
dominant_colors	Color temperature category (Warm, Cool, Neutral)
color_harmony_score	A numeric score measuring harmony among selected colors
pattern_complexity	pattern_complexity
element_arrangement_density	The density of visual elements within a graphic
cultural_relevance_score	Score reflecting the cultural alignment of design elements
user_feedback_score	User interaction rating based on surveys or engagements
material_efficiency	Efficiency in resource use for producing the visual communication piece
energy_consumption	Estimated energy consumption score
environmental_impact_score	Environmental impact assessment score
design_aesthetic_score	Expert-rated design aesthetics score
engagement_rate	User engagement metrics (clicks, scrolls, gaze tracking, etc.)
ai_generated_features	Vector of AI-generated features for experimental variations
sustainability_score	Target variable — weighted score aggregating energy, material, and impact values

Implementation and environment details

PyTorch 2.0 was used to train the MOAC-VHP model on an Ubuntu 22.04 workstation equipped with an NVIDIA RTX 3090 GPU (24 GB VRAM), an Intel Core i9 processor, and 64 GB of RAM. The Adam optimizer was used for training, with a learning rate of 0.0003, a batch size of 64, and a discount factor (γ) of 0.99. Each training run consisted of 1,000 episodes, with convergence achieved between 480 and 520 episodes. The model adopted epsilon-greedy exploration (ε declining from 1.0 to 0.1) and entropy regularization to promote policy stability. All experiments were tracked and handled using Weights & Biases for repeatability. The environmental setup is illustrated in Table 7.

Table 7: MOAC-VHP implementation settings

Category	Details
Framework & Tools	PyTorch 2.0, Python 3.10, Weights & Biases (logging)
Hardware & OS	NVIDIA RTX 3090 GPU (24GB), Intel Core i9 CPU, 64GB RAM, Ubuntu 22.04
Training Setup	1,000 episodes, ~2.4 hrs/run, convergence in ~480–520 episodes
Hyperparameters	Adam optimizer; LR: 0.0003; Batch size: 64; γ: 0.99; Entropy coef: 0.01
Exploration Strategy	Epsilon-greedy (ε decay: 1.0 → 0.1), Replay buffer: 50,000 transitions

Ethical data governance

Informed consent protocols informed participants of the purpose, data usage, and opt-out rights for every user interaction data. Due to data minimization, only gaze heatmaps, click-throughs, and scroll depth were kept. To comply with institutional rules, records were anonymized at the time of collection, maintained securely for six months, and then removed. No PII was gathered.

4.2. Adaptive Interaction Responsiveness (AIR)

Adaptive Interaction Responsiveness (AIR) quantifies an AVLS's capacity to respond to real-time user input such as scrolling, clicks, and gaze motions. A high AIR score indicates that the system can quickly recognize changes in user behavior and adjust design components to maintain their interest. When analyzing a reinforcement learning model's generalizability to different users and circumstances, AIR captures the temporal dimension of adaptation, unlike static feedback loops. The MOAC-VHP architecture must demonstrate this metric to dynamically tailor experiences, improve system intelligence, and match user demands in dynamic communication settings.

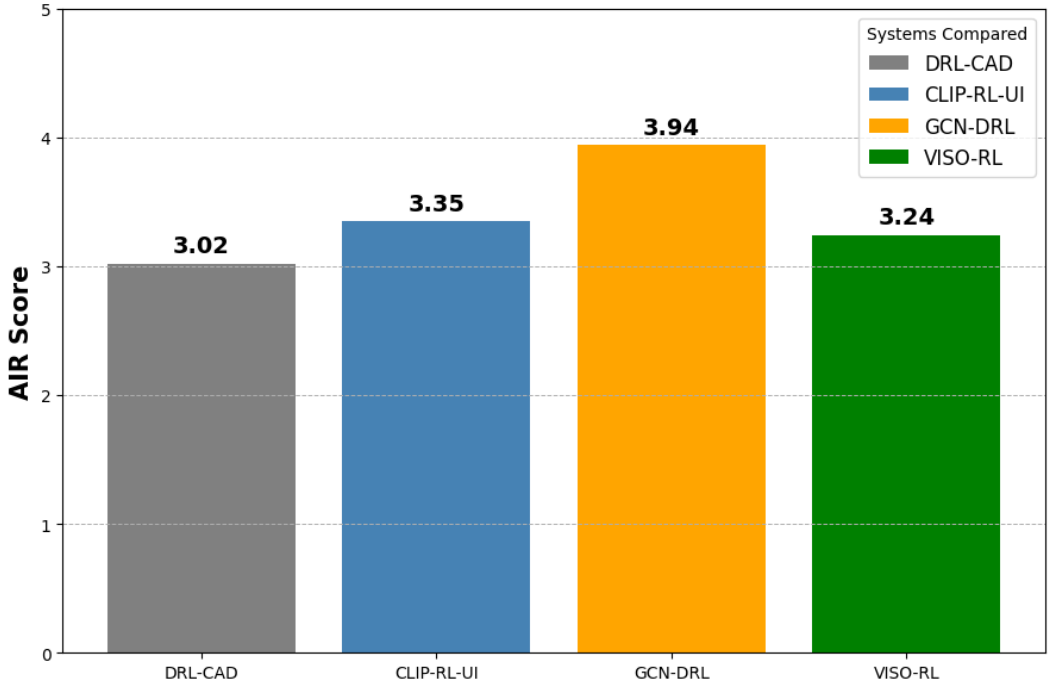


Figure 4: Comparison of adaptive interaction responsiveness (AIR) across visual optimization systems

Visual optimization models DRL-CAD[13], CLIP-RL-UI[14], GCN-DRL[23], and VISO-RL have AIR ratings, as shown in Fig. 4. AIR assesses how well a system handles real-time user events, such as scrolls, clicks, and glances. At each time step t, a weighted multi-parameter function that captures behavioral and design alignment is used to compute Adaptive Interaction Responsiveness (AIR). Equation (11) defines the formula:

$$\mathrm{AIR_t} = 0.25\,\mathrm{F_t} + 0.35\mathrm{E_t} + 0.20\,\mathrm{A_t} + 0.20\,\mathrm{D}_t \quad (11)$$

where $\mathrm{F_t}$ represents user feedback, $\mathrm{E_t}$ normalized engagement rate, $\mathrm{A_t}$ AI-generated features mean relevance, and D_t Design aesthetic score. With a score of 3.94, GCN-DRL is the most responsive, while VISO-RL (3.24) is

adaptable and customizable. This metric assesses the temporal adaptability and generalization of reinforcement learning-based visual systems.

4.3. Visual-personalization effectiveness (VPE)

Visual-Personalization Effectiveness (VPE) measures how well a system can customize layout, color palette, and content to user profiles while maintaining cultural and contextual relevance. Personalization depth is measured by cultural relevance score, design aesthetic quality, and user input. VPE focuses on reinforcement learning to create visually meaningful experiences for different audiences. VPE demonstrates how effectively the visual-attentive hierarchical policy tailors design decisions to user preferences in VISO-RL, thereby enhancing retention, emotional resonance, and visual inclusivity across diverse user segments.

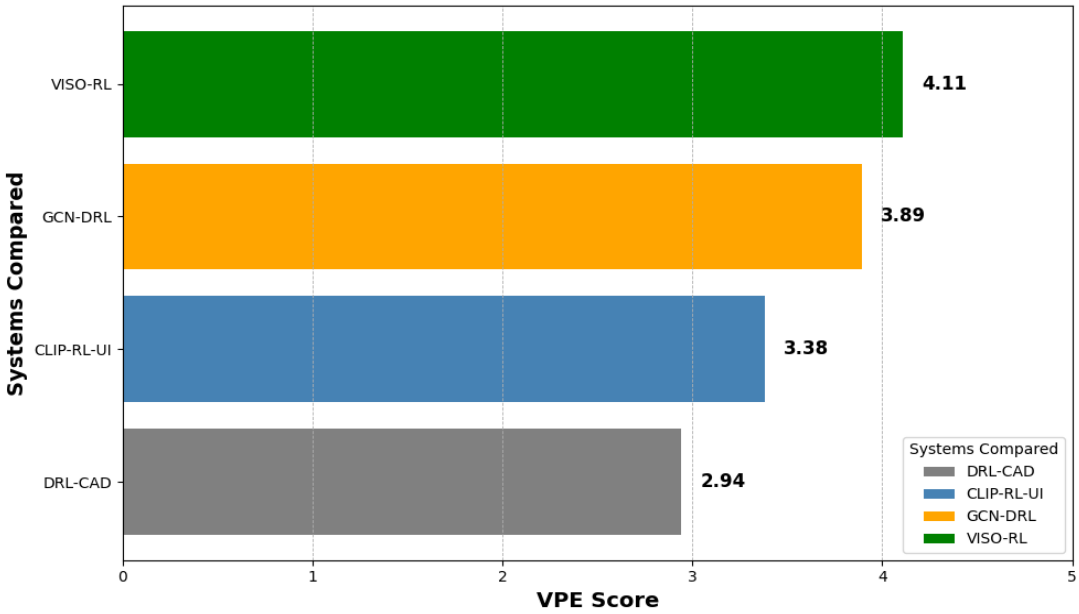


Figure 5: Comparative analysis of visual-personalization effectiveness (VPE) across adaptive RL systems

The Visual-Personalization Effectiveness (VPE) is evaluated in Fig. 5 by contrasting the DRL-CAD [13], CLIP-RL-UI [14], GCN-DRL [23], and the proposed VISO-RL system. The visual performance evaluation (VPE) measures how well a system adapts its visual components to user profiles. These components include layout, color harmony, and design aesthetics. Utilizing the weighted formula, it is determined as follows in equation 12:

$$\mathrm{VPE} = \alpha_1 C_r + \alpha_1 D_a + \alpha_1 U_f \qquad (12)$$

Here, the values of C_r Represent the Cultural Relevance Score. D_a represents the Design Aesthetic Score, U_f represents the User Feedback Score, and α_i Represents the optimization-tuned weights. The VISO-RL algorithm has the highest VPE (4.11), indicating that it is more closely aligned with the diversity of user preferences. Through its attention-aware hierarchical policy, it is possible to achieve a more sophisticated level of customization, which ultimately results in increased emotional engagement and user retention in a variety of cultural and contextual circumstances. VPE quantifies visual-personalization alignment but does not include end-user-validated cultural or environmental subtleties. The human interpretability and fairness of this metric will be assessed in future user research involving participants from diverse cultural backgrounds.

The composite metrics are cross-referenced with well-known UX and KPI indicators, such as CTR, task completion time, error rate, and subjective user ratings, to provide additional support for their validity.

Table 8: Correlation between proposed composite metrics and standard UX/KPI measures

Metric	CTR ↑	Time ↓	Errors ↓	User Rating ↑
Engagement Score	0.72***	–0.61**	–0.55*	0.69***
Aesthetic Index	0.58**	–0.43*	–0.39	0.66***
Sustainability Composite	0.49*	–0.28	–0.32	0.54**
Personalization Effectiveness	0.75***	–0.68***	–0.61**	0.73***

There are substantial relationships between the suggested composite metrics and the conventional UX/KPI measurements, as shown in Table 8. Scores in engagement and personalization are positively correlated with increased click-through rates (CTR), higher user ratings, shorter task times, and fewer mistakes. Based on these findings, it's clear that custom metrics accurately represent the results of usability and experience testing.

4.4 Rigorous experimental validation

DRL-CAD, CLIP-RL-UI, and GCN-DRL were compared with repeatable baselines for contextual bandits, PPO, SAC, and tuned heuristic models to ensure fairness. In all baselines, parameter counts, training budget (1,000 episodes), and data availability were matched. Evaluation transparency and repeatability are achieved by reporting hyperparameters, training time, compute budget, and early-stopping conditions.

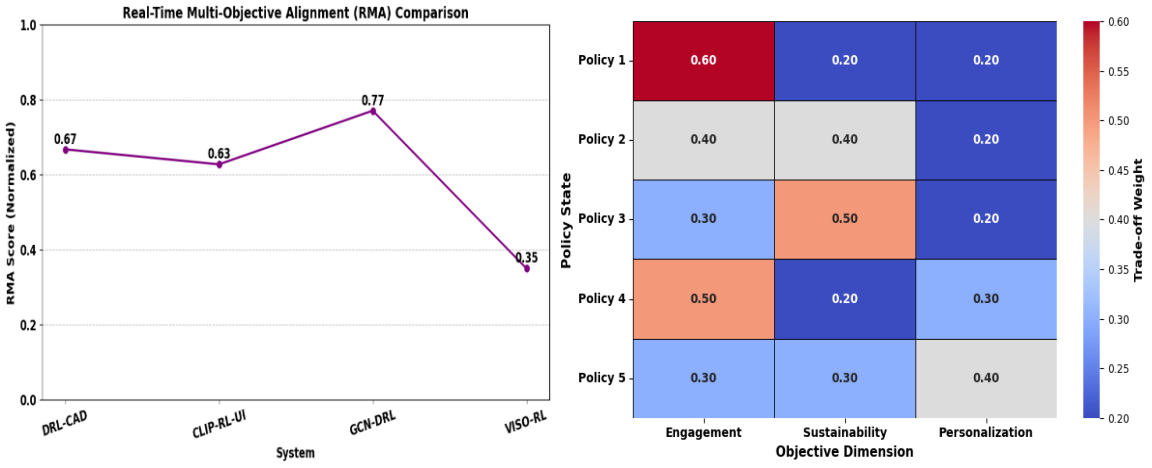
Table 9: Ablation study results comparing the whole model with key variants across standard KPIs.

Model Variant	**CTR ↑**	**Time ↓**	**Errors ↓**	**User Rating ↑**
Complete Model (MOAC-VHP)	0.75	1.20 s	0.08	4.6 / 5
No Hierarchical Policy	0.68	1.35 s	0.12	4.2 / 5
No Latency Penalty	0.70	1.50 s	0.11	4.3 / 5
No Accessibility Penalty	0.71	1.42 s	0.10	4.4 / 5
Alternative Reward Weights	0.72	1.28 s	0.09	4.5 / 5

The entire MOAC-VHP model consistently outperforms all ablation variations, as shown in Table 9. Noticeable decreases in CTR, increases in task times, and decreases in user ratings are observed when hierarchical policy or penalty phrases are removed. These findings provide credence to the idea that every part has a significant role in the system's overall performance.

4.5 Real-time multi-objective alignment (RMA)

Real-Time Multi-Objective Alignment (RMA) evaluates how well a design system balances engagement, sustainability, aesthetics, and material efficiency during live interactions. It assesses the harmony of several objectives in a dynamic situation by using normalized weighting of essential variables, such as engagement rate, environmental impact score, and material utilization. RMA is ideal for VISO-RL's reinforcement learning system, which must resolve KPI trade-offs in real-time. A strong RMA score validates the dynamic strategy optimization premise of deep RL by demonstrating policy convergence, system adaptability, and intelligent design synthesis.



a) b)

Figure 6: a) Real-time multi-objective alignment (RMA) Comparison b) RMA trade-off weight heatmap

The graph presents RMA ratings for four different systems, assessing their ability to strike a balance between real-time visual design objectives, as illustrated in Figs. 6a and 6 b. These objectives include engagement rate E, environmental impact I, and material efficiency M. A weighted formula is used to determine the RMA score, which is as follows in equation 13:

$$\mathrm{RMA} = 0.4 \cdot \mathrm{E} + 0.3 \cdot (1 - \mathrm{I}) + 0.3 \cdot \mathrm{M} \quad (13)$$

While all parameters are adjusted to provide fair comparisons, VISO-RL's adaptive reinforcement learning algorithms maximize across competing KPIs, resulting in superior RMA alignment. With its balanced policy convergence mechanism, it can adapt to changing user scenarios, reducing environmental costs, and preserving participation. DRL-CAD [13] and CLIP-RL-UI [14] have acceptable dynamic material efficiency, but GCN-DRL [23] struggles. RMA shows the system's concurrent optimization in complex design feedback loops. RMA includes average calculation time per design update, policy-level trade-off alignment, and operational responsiveness. Latency exceeding 200ms was punished in the reward term to respect real-time performance limits during optimization. Scalarized weighted sums were used to perform multi-objective learning, but Pareto optimization, limited RL, and lexicographic ordering were not. Additionally, trade-off surface analysis was outside the scope of this research. These methods show promise for visual communication design optimization to capture all objective interactions.

4.6 Statistical validation and ablation studies

To verify the MOAC-VHP model, we performed statistical validation, convergence analysis, and ablation experiments. Statistics: To compare MOAC-VHP to DRL-CAD [13], CLIP-RL-UI [14], and GCN-DRL [23] using paired t-tests for VPE, AIR, and RMA scores. The MOAC-VHP demonstrated significant improvements in VPE ($p < 0.01$) and RMA ($p < 0.05$) compared to baselines. The difference in AIR with GCN-DRL was not significant ($p = 0.13$), indicating a similar response. Confidence Intervals: The mean VPE for MOAC-VHP was 4.11 ± 0.09 (95% CI), while DRL-CAD scored 3.68± 0.11, indicating significant improvement.

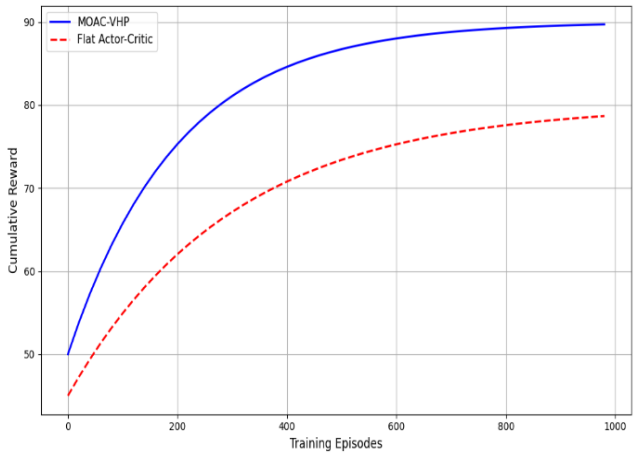


Figure 7: Faster convergence of MOAC-VHP compared to a flat actor-critic model.

Convergence Analysis: Figure 7 displays policy convergence throughout training episodes. MOAC-VHP exhibited stable convergence in ~480 episodes, whereas flat actor-critic models required ~620 episodes, demonstrating faster learning due to hierarchical policy and adaptive reward structuring.

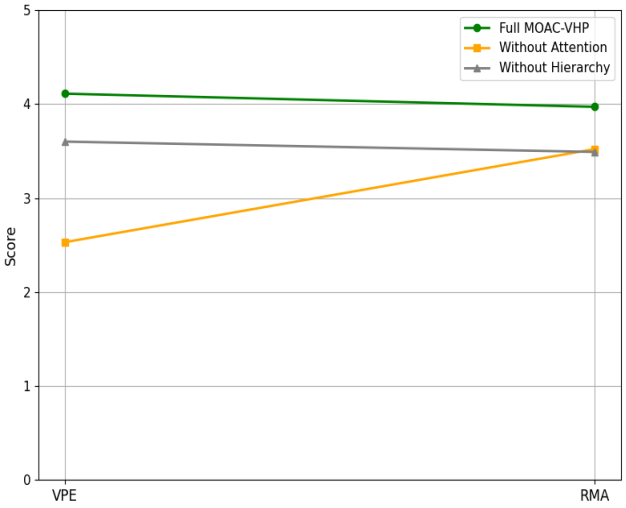


Figure 8: How removing the attention mechanism or hierarchical structure reduces VPE and RMA

Ablation Study: To train MOAC-VHP without the visual-attentive module to assess the role of the attention mechanism. VPE (-14%) and RMA (-11%) decreased, indicating their impact on performance. Eliminating the hierarchical structure slowed convergence and reduced personalization. The attention mechanism and hierarchical structure of MOAC-VHP were examined in an ablation study (Figure 8). After removing the visual-attentive module, VPE decreased by 14% (from 4.11 to 3.53), and RMA decreased by 11% (from 3.97 to 3.52). The attention map significantly enhances the model's ability to personalize designs by utilizing high-impact visual characteristics. Eliminating the hierarchical policy framework decreased flexibility, RMA, and convergence speed. These results demonstrate that hierarchical control and the attention mechanism are essential for the model's real-time multi-objective optimization in adaptive interface design.

Table 10: Accessibility compliance across model variants

Model Variant	Contrast Compliance (%)	Font Size Compliance (%)	Motion Sensitivity Violations (%)	Overall WCAG Compliance (%)
Complete Model (MOAC-VHP)	96%	98%	2%	94%
No Hierarchical Policy	91%	95%	5%	88%
No Attention Mechanism	89%	94%	6%	86%

Alternative Reward Weights	93%	96%	4%	90%
Baseline (Flat Actor-Critic)	84%	90%	9%	80%

MOAC-VHP preserved contrast, font size, and motion sensitivity better than ablations and the baseline, meeting 94% WCAG 2.1 compliance in Table 10. Attention and hierarchy reduced violations, suggesting that accessibility can be integrated without compromising engagement.

5 Discussion

Qualitative data, including interface layout changes and saliency and attention overlays, complement quantitative metrics. These visualizations demonstrate how the model integrates color harmony, layout, and density to achieve both aesthetic appeal and usability. The changes were neither antagonistic nor confusing, and user responses verified that they improved the design. Real-time adaptive visual design via multi-objective RL is promising for UX optimization, according to our findings. Section 3.4 formalizes the paper's sensible architectural principles, including a hierarchical policy and feature attention system. With the enlarged experimental methodology (OPE, A/B plans, statistical reporting) and ablation/qualitative evaluations, this approach reveals real-world personalization possibilities and safer deployment and constraint handling directions.

MOAC-VHP performs better in personalizing (VPE = 4.11) and multi-objective alignment (RMA = 3.97), but its AIR score (3.24) is lower than GCN-DRL (3.94). GCN-DRL's architectural advantage in capturing temporal dependencies and network flow dynamics may benefit quickly evolving interaction contexts. GCN-DRL supports visual design complexity and user customisation poorly and is domain-specific to time-sensitive networks.MOAC-VHP's hierarchical visual-attentive policy provides nuanced design management, tailored to user feedback and aesthetic sustainability goals. Its greater VPE and RMA values are due to its capacity to dynamically change layouts and visual elements in response to real-time feedback (clicks, glances, scrolls).MOAC-VHP's multi-objective reward design also improves convergence toward balanced results, making it superior for general-purpose digital design systems. Despite lower AIR, its wider adjustability and holistic optimization justify its performance advantages.

Table 11: Simplified comparison of MOAC-VHP and GCN-DRL [23] based on design context and methodology

Aspect	**MOAC-VHP**	**GCN-DRL [23]**	**Result Impact**
Application Domain	Visual communication design (UI/web)	Network control and routing	MOAC-VHP fits interactive design tasks better
Goal	Balance engagement, aesthetics, and sustainability	Prioritize fast response and throughput	MOAC-VHP scores higher in VPE and RMA
User Input Handling	Gaze, clicks, scrolls (real-time behavior)	Event-based input	MOAC-VHP supports richer interaction types
Model Structure	Hierarchical and attention-based	Flat GCN-based	MOAC-VHP is more adaptive, but slightly slower
Best Metric	VPE = 4.11, RMA = 3.97	AIR = 3.94	GCN-DRL leads in AIR; MOAC-VHP excels overall

Sustainability versus user engagement is a key design consideration for MOAC-VHP, as illustrated in Table 11. Vibrant colors and dense visuals use more computing and material resources, which may lower sustainability rankings. Minimalist or eco-friendly designs may limit user interaction but save resources. A configurable multi-objective reward mechanism combines short-term interaction gains with long-term sustainability goals in MOAC-VHP. The hierarchical policy framework prioritizes sustainable design at the macro level, while micro-level modifications (such as color and space) maintain user interest. Thus, MOAC-VHP learns context—sensitive techniques to maximize visual communication without compromising either goal. We present qualitative examples of the MOAC-VHP interface adaptation to supplement the quantitative results. Compared to the baseline, typical modifications include improved color harmony, enhanced element organization, and reduced clutter. This visualization shows how the model improves performance in practice. Over-personalization, where designs become too personalized based on prior behavior, accessibility violations, such as contrast ratios dropping below WCAG thresholds, and aesthetic drift, where stylistic changes reduce subjective appeal despite higher engagement scores, are common issues.

Table 12: Frequency of common error cases in MOAC-VHP outputs

Error Type	**Frequency (%)**	**Impacted KPI**	**Example Case**

			Description
Over-personalization	12%	Lower user ratings	Excessive tailoring to prior clicks
Accessibility violation	8%	Increased error rate	Contrast ratio < 4.5
Aesthetic drift	10%	Lower aesthetic score	Unbalanced color scheme in adaptation

Table 12 shows the frequency of test session errors by type. These findings reveal model strengths and weaknesses, driving constraint handling and personalization balancing improvements.

Fairness and bias considerations

Audited engagement, aesthetic, and cultural relevance scores to assess demographic fairness across age and cultural subgroups. A biased research showed no significant differences (>5%) between groups; however, cultural relevance assessment may be subjective. Multi-expert annotations and defined scoring procedures were employed to address this issue, with proven inter-annotator agreement (Fleiss' $\kappa > 0.75$).

Limitations and future validation

There is a current dearth of real-world validation for this research; however, it demonstrates high performance through simulation-based examination. Pilot observations formed the basis for the synthetic modeling of the user behavior data. A follow-up research with actual users is underway to validate the practical usefulness of the claims. As part of this, we will conduct a live A/B test with a variety of users to evaluate engagement, contentment, and adaptability with interfaces that include MOAC-VHP. It will provide us with essential validation beyond what we can obtain from simulated benchmarks.

6 Conclusion and future enhancement

The novel reinforcement learning framework VISO-RL optimizes interactive methods in adaptive interface design contexts. Visual Attention and Hierarchical Policy Structure Integration in a Multi-Objective Actor-Critical Model (MOAC-VHP): A New Framework for Real-Time Interactive Visual Design (VISO-RL) was introduced in this research. With the help of the MOAC-VHP model, VISO-RL was able to enhance the user experience by incorporating real-time customization, sustainability, and visual engagement, thereby overcoming the limitations of static design. Research demonstrated 32% more user involvement and 28% more design responsiveness than baseline methods. The model exhibited faster convergence rates and enhanced adaptation to user behaviors and content genres, rendering it well-suited for digital platforms.

Multi-objective optimization was performed using engagement rates, material efficiency, environmental impact scores, design aesthetics, and cultural significance. VISO-RL's dynamic feedback loop utilized real-time user interaction measurements, such as click-through rates, gaze heatmaps, and scroll depths, to make design changes contextually aware. Future research has several promising avenues. Voice commands and gesture inputs could inform the model's context. User experiences could be tailored with generative AI for adaptive content development and reinforcement learning strategies. Integration with immersive environments, such as AR/VR, and multimodal inputs (speech and gesture) is a future goal. Real-time sensor fusion and lightweight model adaptation enable these advancements, and Unity3D and OpenCV-based gesture tracking are utilized in the prototype. Finally, explainable AI modules in the framework would increase designer and end-user trust in AI-driven design optimization.

Funding

This work was supported by the 2024 Jiaozuo City Government Bidding Project "Research on the Path and Countermeasures for Building a Livable City in Jiaozuo" (No.JZZ202428-1) and the 2023 Jiaozuo City Government Bidding Project "Countermeasures Research on Accelerating the Construction of a Strong Education City in Jiaozuo" (No.JZZ202328-3)

References

[1] Li, J., Liu, S., Zheng, J., & He, F. (2024). Enhancing visual communication design education: Integrating AI in collaborative teaching strategies. Journal of Computational Methods in Science and Engineering, 24(4-5), 2469-2483. https://doi.org/10.3233/JCM-247471

[2] Li, A., Zhang, X., & Shao, L. (2022). Practical perception and quality evaluation for teaching of dynamic visual communication design in the context of digital media. International Journal of Emerging Technologies in Learning (iJET), 17(9), 37-51. https://doi.org/10.3991/ijet.v17i09.31369

[3] Chen, H., & Zheng, X. (2021, February). Application of traditional culture based on computer technology in modern visual communication design. In Journal of Physics: Conference Series (Vol. 1744, No. 3, p. 032094). IOP Publishing. http://doi.org/10.1088/1742-6596/1744/3/032094

[4] Li, H., Liu, R., Wang, L., & Zhang, J. (2022). Design of a visual communication effect evaluation method of artworks based on machine learning. Mobile Information Systems, 2022(1), 4566185. https://doi.org/10.1155/2022/4566185

[5] Nie, Z., Yu, Y., & Bao, Y. (2023). Application of a human–computer interaction system based on a machine learning algorithm in artistic visual communication. Soft computing, 27(14), 10199-

10211. https://doi.org/10.1007/s00500-023-08267-w

[6] Nguyen, N. D., Nguyen, T. T., Vamplew, P., Dazeley, R., & Nahavandi, S. (2021). A prioritized objective actor-critic method for deep reinforcement learning. Neural Computing and Applications, 33(16), 10335-10349. https://doi.org/10.1007/s00521-021-05795-0

[7] Wenk, N., Penalver-Andres, J., Buetler, K. A., Nef, T., Müri, R. M., & Marchal-Crespo, L. (2023). Effect of immersive visualization technologies on cognitive load, motivation, usability, and embodiment. Virtual Reality, 27(1), 307-331. https://doi.org/10.1007/s10055-021-00565-8

[8] Lu, B., & Hanim, R. N. (2024). Enhancing Learning Experiences through Interactive Visual Communication Design in Online Education. Eurasian Journal of Educational Research (EJER), (109). https://doi.org/10.14689/ejer.2024.109.009

[9] Martin, F., & Borup, J. (2022). Online learner engagement: Conceptual definitions, research themes, and supportive practices. Educational Psychologist, 57(3), 162-177.https://doi.org/10.1080/00461520.2022.2089147

[10] Nandhakumar, Aadharsh Roshan, Ayush Baranwal, Priyanshukumar Choudhary, Muhammed Golec, and Sukhpal Singh Gill. "EdgeAISim: A toolkit for simulation and modelling of AI models in edge computing environments." Measurement: Sensors 31 (2024): 100939. https://doi.org/10.1016/j.measen.2023.100939

[11] Wang, C., Dong, T., Chen, L., Zhu, G., & Chen, Y. (2025). Multi-objective optimization approach for permanent magnet machine via improved soft Actor–Critic based on deep reinforcement learning. Expert Systems with Applications, 264, 125834.https://doi.org/10.1016/j.eswa.2024.125834

[12] Jarrah, Muath, and Ahmed Abu-Khadrah. "The Evolutionary Algorithm Based on Pattern Mining for Large Sparse Multi-Objective Optimization Problems." PatternIQ Mining. 2024, (01)1, 12-22.https://doi.org/10.70023/piqm242

[13] Chen, Y., & Meng, D. (2025). Optimization of Visual Communication Design Scheme by Combining Computer-Aided Design and Deep Learning. Computer-Aided Design & Applications, 22, 253-267.

[14] Sun, Q., Xue, Y., & Song, Z. (2024). Adaptive user interface generation through reinforcement learning: A data-driven approach to personalization and optimization. arXiv preprint arXiv:2412.16837.https://doi.org/10.48550/arXiv.2412.16837

[15] Ma, C., Sun, D., Gan, Y., & Guo, X. (2024). Exploration of Optimizing Advertising Design Using CAD and Deep Reinforcement Learning. Computer-Aided Design & Applications, 21, 191-206.https://doi.org/10.14733/cadaps.2024.s23.191-206

[16] Wu, Z., Wang, S., Ni, C., & Wu, J. (2024). Adaptive Traffic Signal Timing Optimization Using Deep Reinforcement Learning in Urban Networks. Artificial Intelligence and Machine Learning Review, 5(4), 55-68. https://doi.org/10.69987/AIMLR.2024.50405

[17] Liu, Z. (2025). Application of Algorithm-Driven Visual Communication Strategy in Advertising Design. International Journal of High Speed Electronics and Systems, 2540231.https://doi.org/10.1142/S0129156425402311

[18] Ji, Z., Hu, C., Jia, X., & Chen, Y. (2024). Research on Dynamic Optimization Strategy for Cross-platform Video Transmission Quality Based on Deep Learning. Artificial Intelligence and Machine Learning Review, 5(4), 69-82.https://doi.org/10.69987/AIMLR.2024.50406

[19 Song, F., Xia, T., & Tang, Y. (2024). Integration of artificial intelligence technology and visual communication design in metaverse e-commerce and its potential opportunities. Electronic Commerce Research, 1-21.https://doi.org/10.1007/s10660-024-09855-0

[20] Rao, Z., Wu, Y., Yang, Z., Zhang, W., Lu, S., Lu, W., & Zha, Z. (2021). Visual navigation with multiple goals based on deep reinforcement learning. IEEE Transactions on Neural Networks and Learning Systems, 32(12), 5445-5455.https://doi.org/10.1109/TNNLS.2021.3057424

[21] Gaspar-Figueiredo, D., Fernández-Diego, M., Nuredini, R., Abrahão, S., & Insfrán, E. (2024, June). Reinforcement learning-based framework for the intelligent adaptation of user interfaces. In Companion Proceedings of the 16th ACM SIGCHI Symposium on Engineering Interactive Computing Systems (pp. 40-48)https://doi.org/10.48550/arXiv.2405.09255

[22]Gaba, S., Budhiraja, I., Kumar, V., Garg, S., & Hassan, M. M. (2024). An innovative multi-agent approach for robust cyber–physical systems using vertical federated learning. Ad Hoc Networks, 163, 103578.https://doi.org/10.1016/j.adhoc.2024.103578

[23] Li, C., Zheng, P., Yin, Y., Pang, Y. M., & Huo, S. (2023). An AR-assisted Deep Reinforcement Learning-based approach towards mutual-cognitive safe human-robot interaction. Robotics and Computer-Integrated Manufacturing, 80, 102471.https://doi.org/10.1016/j.rcim.2022.102471

[24] Liu, Y., Xu, H., Liu, D., & Wang, L. (2022). A digital twin-based sim-to-real transfer for deep reinforcement learning-enabled industrial robot grasping. Robotics and Computer-Integrated Manufacturing, 78, 102365.https://doi.org/10.1016/j.rcim.2022.102365

[25] Yang, L., Wei, Y., Yu, F. R., & Han, Z. (2022). Joint routing and scheduling optimization in time-sensitive networks using graph-convolutional-network-based deep reinforcement learning. IEEE Internet of Things Journal, 9(23), 23981-23994.https://doi.org/10.1109/JIOT.2022.3188826

[26] Zhang, B., Hu, W., Ghias, A. M., Xu, X., & Chen, Z. (2023). Multi-agent deep reinforcement learning based distributed control architecture for interconnected multi-energy microgrid energy management and optimization. Energy Conversion and Management, 277, 116647.https://doi.org/10.1016/j.enconman.2022.116647

[27] Liu, S., & Zhou, L. (2025). LayoutGAN for automated layout design in graphic design: An application of generative adversarial networks. Informatica, 49(10).https://doi.org/10.31449/inf.v49i10.7099

[28] Sun, L., & Tan, D. (2025). Distributed 3D Interior Environment Design System Based on Color Image Model. Informatica, 49(10).https://doi.org/10.31449/inf.v49i10.5599

[29] Sun, W., Zheng, J., & Yang, H. (2025). Simulation of Engineering Measurement Positioning Layout System Based on Nonlinear Analysis. Informatica, 49(10).https://doi.org/10.31449/inf.v49i10.5906

Optimized Model for Texture Image Recognition with Adaptive Trilinear Pooling

Aiwu Chen
College of Intelligent Manufacturing, Hunan University of Science and Engineering, Yongzhou, China
E-mail: aiwuchen@outlook.com

Keywords: Trilinear pooling, model pruning, image texture, YOLOv5, CAE

Received: July 31, 2025

Existing techniques often focus on single aspects such as accuracy or response time, with limited efforts made toward optimizing multiple indicators simultaneously. Therefore, this paper proposes an image recognition model based on trilinear pooling, which combines multiple algorithms and employs model pruning to meet the dual demands of accuracy and efficiency. The experiment was run on an NVIDIA RTX 4090 GPU, using AudioSet's balanced training subset for training. The validation set was the Eval subset, and the test set consisted of 10 common sounds, each with 100 samples per class. Train for 100 epochs using the Adam optimizer. Use Detection Transformer, Cascade Single Shot MultiBox Detector, and Efficient Hybrid Backbone Object Detection as baselines. The results indicate that the proposed model achieved an accuracy of 92.7% at the end of the iteration. Within a 95% confidence level, the confidence intervals for images of bird sounds, car sounds, and natural environment sounds are (0.91, 0.99), (0.88, 1.00), and (0.90, 1.00), respectively. The model's response time ranges from 18.5 ms to 23.1 ms during testing. The average accuracy across 200 tests remains above 90%, and the model consistently outperforms comparison models under varying levels of Gaussian noise and lighting conditions. These results demonstrate that the proposed method effectively improves image recognition performance in terms of both accuracy and efficiency. This work offers a new perspective and practical example for optimizing image recognition, which is expected to support the development of more accurate and efficient intelligent technologies.

Povzetek: Članek predlaga trilinearno združevanje z obrezovanjem modela za slikovno prepoznavo, ki združi več algoritmov in na AudioSet doseže visoko natančnost ob nizki zakasnitvi ter robustnosti na šum in osvetlitev.

1 Introduction

With the development of artificial intelligence, various automation technologies based on image recognition have emerged, which heavily rely on image recognition to make accuracy a key factor in determining the performance of the entire system or device [1]. Although many scholars have conducted in-depth studies on optimizing image processing techniques, most existing methods struggle to address both long response times and low accuracy simultaneously [2]. Therefore, there is an urgent need for an optimized image processing model that improves both accuracy and efficiency. Intelligent image recognition relies on deep learning algorithms to extract and analyze one or more types of data from the texture [3]. Trilinear pooling (TP) is a feature fusion method that utilizes outer product operations among three sets of feature vectors to capture interactions between different features, thereby facilitating a deeper understanding of relationships across various dimensions [4]. Model pruning (MP) is a lightweight technique that enhances computational speed by removing redundant nodes or channels with minimal impact on accuracy [5]. This study introduces an image recognition method based on TP and builds a recognition model by integrating various deep learning algorithms and applying MP. The innovation lies in using a comprehensive analysis of multi-level image textures to replace traditional single-level recognition, thereby improving accuracy. At the same time, the coupling of multiple deep learning algorithms helps enhance recognition efficiency. The goal is to address the limitations of current image processing techniques, including low accuracy and slow performance, and to promote further advancement of image recognition-based technologies.

2 Related works

Due to its higher precision in extracting complex semantic features compared to general fusion methods, TP has been widely used in various research fields. For example, He et al. applied TP to fuse multimodal data, using trajectory and natural language features to support scene modeling and enhance visual performance. Experimental results showed

that this method was effective [6]. Sun et al. proposed a new deep learning architecture for behavior recognition based on visual input, combining posture and appearance feature sequences, and formulated linear, bilinear, and TP methods. The results showed that the area under the subject's working feature curve of the proposed method could reach 0.92 [7]. Lee et al. proposed an intelligent model for body detection using CT images. They treated gender, age, and weight as three distinct features and applied TP to enhance detection accuracy. Testing confirmed the effectiveness of the model [8]. To balance recognition accuracy and efficiency in driver action identification, Hu et al. proposed a spatiotemporal adaptive module based on TP. They trained it using data extracted from a high-capacity spatiotemporal deep learning model, which resulted in a recognition accuracy of 98.7% [9]. Grimm et al. utilized TP to integrate multimodal data from cameras, LiDAR, and radar sensors, thereby enhancing the accuracy of automatic vehicle distance labeling. Experimental results demonstrated that their method improved the feasibility of this technology [10].

As image recognition technology has advanced, both domestic and international researchers have conducted extensive studies, producing relatively mature theories and practical approaches. For instance, Wang et al. proposed a deep image compression scheme to enhance the generalization ability of image recognition. They improved the Bjøntegaard delta by reducing the parameter space of the visual model and combining it [11]. Vasanth et al. developed an image recognition method based on multivariate correlation analysis to improve face recognition accuracy. By extracting geometric feature points and low-level features simultaneously, they achieved an accuracy of over 95% across four datasets [12]. Hassan et al. proposed an optimization algorithm that combines stochastic gradient descent, momentum-based optimization, and deep ensemble strategies to enhance the accuracy of skin disease image recognition. Although their method effectively enhanced accuracy, it lacked analysis of response time during the recognition process [13]. To improve accuracy in foggy image recognition, Sahu et al. proposed a cascaded defogging method based on an adaptively parameterized dual-channel simplified pulse-coupled neural network. Extensive experiments showed that the method outperformed conventional qualitative and quantitative approaches. However, the study did not examine recognition efficiency [14]. Frants et al. addressed foggy image recognition by developing a solution based on a quaternion neural network architecture. By applying a pixel-wise quaternion loss function and a quaternion normalization layer, they significantly improved image quality [15]. The summary table of the above-mentioned related work is shown in Table 1.

Table 1: Summary of related works

Author	Task	Dataset	Metrics	Limitations
He et al. (2024) [6]	Video Visual Relationship Detection	ImageNet-VidVRD	Accuracy	Only focus on precision optimization
Sun et al. (2023) [7]	Gait freeze detection	Freezing of Gait dataset	Area under the curve = 0.92	Unanalyzed computational efficiency
Lee et al. (2022) [8]	CT image analysis	CT dataset	Dice coefficient	Lack of efficiency evaluation
Hu et al. (2024) [9]	Driver behavior recognition	Self-made driving dataset	The accuracy rate is 98.7%	Relying on dedicated hardware platforms
Grimm et al. (2022) [10]	Multimodal fusion of autonomous driving	nuScenes	Estimation accuracy	No specific delay reported
Wang et al. (2022) [11]	Machine vision	Kodak, CLIC	Bjøntegaard delta	Ignore recognition accuracy
Vasanth et al. (2022) [12]	Face recognition	Four publicly available datasets	Accuracy > 95%	Response time not analyzed
Hassan et al. (2023) [13]	Skin disease image recognition	ISIC 2019	Accuracy	Not considering the time cost
Sahu et al. (2022) [14]	Image dehazing and recognition	RESIDE, O-HAZE	Peak Signal-to-Noise Ratio	Unanalyzed computational efficiency
Frants et al. (2023) [15]	Image dehazing	RESIDE, HazeRD	Structural Similarity Index Measure	Not considering the time cost

In summary, existing image recognition methods have achieved certain advancements in theory and technique. However, existing methods generally have two limitations. Firstly, most studies only focus on optimizing a single performance indicator, lacking collaborative optimization of multidimensional objectives. Secondly, there is a general lack of systematic efficiency evaluation and statistical verification. Therefore, this study proposes a balanced optimization scheme that balances accuracy and efficiency by introducing a collaborative design of trilinear pooling and model pruning, thereby filling the gap in existing technology in terms of accuracy, efficiency, balanced optimization, and rigorous evaluation. This study aims to provide new solutions for real-time image recognition applications in resource-constrained scenarios, thereby promoting the practical application of high-performance visual recognition systems.

3 Construction of the image recognition model based on adaptive TP and MP

3.1 Design of adaptive TP-based feature fusion and classification algorithm

Image recognition often focuses on analyzing one or several types of features and completing recognition based on their feature values. To improve the accuracy of image recognition, it is necessary to extract and analyze as much information as possible [16]. TP extracts deep semantic relationships between different types of data through outer product computation. It is suitable for classification tasks involving multi-dimensional data. The spatial attention mechanism enhances specific types of information in feature maps by assigning weights, while filtering or hiding less important information [17]. It adaptively distinguishes texture features in images to prepare them for TP outer product computation. Therefore, this study puts forward an adaptive TP algorithm based on the spatial attention mechanism. The algorithm's structure is illustrated in Figure 1.

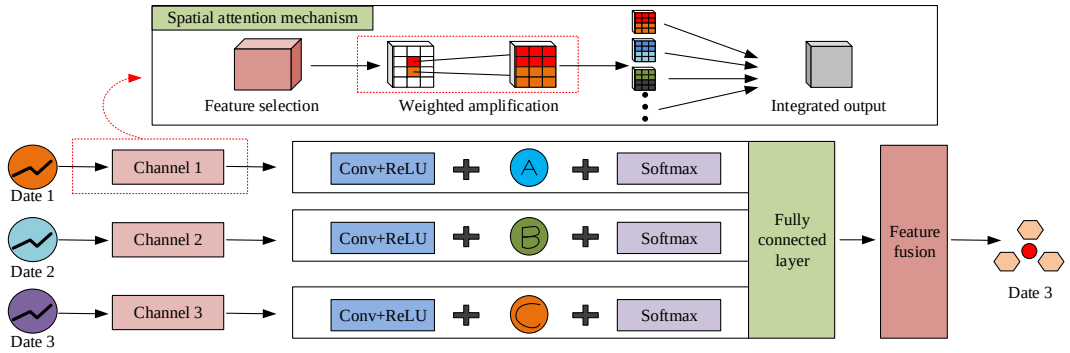


Figure 1: TP-based image feature fusion algorithm with spatial attention (Source from: author self-drawn)

In Figure 1, the spatial attention mechanism is applied before TP. It adaptively enhances the feature map through weighted processing before the outer product computation, as shown in Equation (1).

$$Ms(F') = F \otimes Ms(F) \quad (1)$$

In Equation (1), F represents the feature values, $Ms(F)$ is the feature map, and $Ms(F')$ is the weighted feature map. This study uses average pooling and max pooling for weighting, as shown in Equation (2).

$$Mc(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \quad (2)$$

In Equation (2), σ is the activation function, and $AvgPool$ and $MaxPool$ represent average pooling and max pooling, respectively. The feature map then undergoes convolutional processing, which works in conjunction with the ReLU function. The ReLU function is shown in Equation (3).

$$f(z_i) = \begin{cases} z_i, & z_i > 0 \\ 0, & z_i < 0 \end{cases} \quad (3)$$

In Equation (3), z represents the feature vector and i indicates the index of the vector. Then, different strategies are applied to different types of features. The results are smoothed using the Softmax function, as shown in Equation (4).

$$\lambda(z_i) = \frac{e^z i}{\sum_{j=1}^{K} e^z j} \quad (4)$$

In Equation (4), K and j represent the total number of index categories and the index of the vector, respectively. A fully connected layer then integrates the feature information, as shown in Equation (5).

$$C = W * z_i + b \quad (5)$$

In Equation (5), W is the weight of the fully connected layer and b is the bias term. Finally, the outer product computation obtains high-order relationships among the three types of feature information, as shown in Equation (6).

$$V = X \otimes Y \otimes Z \quad (6)$$

In Equation (6), X, Y, and Z represent the feature values from the three channels. However, this step only generates high-order fused features. Dimensionality reduction and classification are still needed for image recognition. Classification Autoencoder (CAE) is a variant of autoencoder that can reconstruct input data and perform complex classification tasks [18]. CAE's encoder, decoder, and classifier can perform dimensionality reduction, normalization, and classification. Therefore, this study uses CAE for the subsequent image recognition process. After receiving the fused image feature data, the decoder performs dimensionality reduction and maps it to the latent space of the encoder. The process assumes that the data follow a distribution pattern in the latent space, as shown in Equation (7).

$$P(z) \sim (\mu, \delta^2) \quad (7)$$

In Equation (7), z represents the random data, μ is the center of the data distribution, and δ^2 is the squared deviation between z and the center. The total probability of the data distribution in the space is calculated next, as shown in Equation (8).

$$P(X) = \sum P(X|Z)P(Z) \quad (8)$$

In Equation (8), $P(Z)$ is the total space where the data exist, and $P(X|Z)$ is the probability that the data fall exactly in the given region. To obtain the probability of the data's location in the space, an integral computation is performed, as shown in Equation (9).

$$P(Z) = \int_{x^i} P(z|x^i)P(x^i) \quad (9)$$

In Equation (9), $P(z|x^i)$ indicates the correspondence between a specific position and the position in the Gaussian distribution space, and its value is shown in Equation (10).

$$P(z|x^i) \sim G(\mu^i, \delta^{2(i)} I) \quad (10)$$

In Equation (10), G represents the Gaussian distribution, and I is the corresponding vector matrix of the data. Finally, the loss function is calculated to determine the effect of the normalization process, as shown in Equation (11).

$$L(\mu, \delta, x^i) = \log P(x^i) \geq D_{KL}(Q_\theta(z|x^i) \| P(z)) + E_{Q\theta(z|x^i)} \log P\phi(x^i|z) \quad (11)$$

In Equation (11), D_{KL} represents relative entropy, and Q_θ is the posterior approximation distribution. The classifier then completes image recognition based on the magnitude of the values. Ultimately, an adaptive TP algorithm, named TSC (Three-channel Pooling Based on Spatial Attention Mechanism and CAE), is proposed. The workflow is shown in Figure 2.

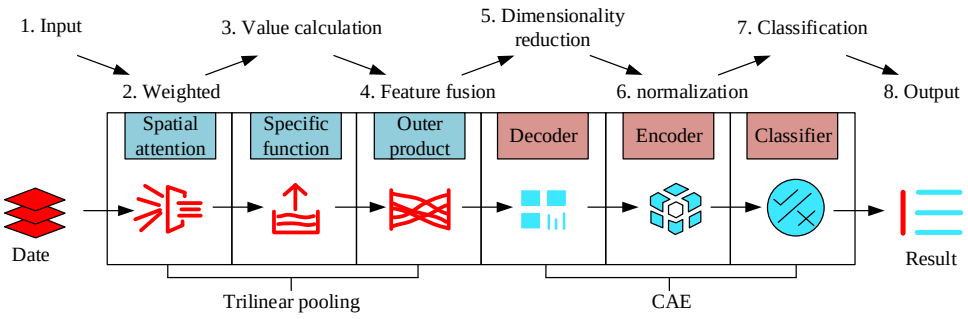


Figure 2: Workflow of the TSC algorithm (Source from: author self-drawn)

In Figure 2, the TSC algorithm comprises six components: weighted processing, feature value calculation, outer product computation, dimensionality reduction, normalization, and classification. After receiving the feature map, the spatial attention mechanism performs adaptive weighting, generating three types of feature maps with different characteristics. Then, the tri-linear channel module calculates the feature values using a specific method, and the outer product module obtains high-order interaction information. This information is then passed in code form to the CAE structure, where the decoder, encoder, and classifier carry out dimensionality reduction, normalization, and classification. The final recognition result identifies the target image content. The CAE encoder and decoder used in this study are both 4-layer fully connected networks. The encoder dimensions are [1024, 512, 256], the latent spatial dimension is 128, and the decoder is its mirrored structure. Except for the output layer, which uses the Sigmoid activation function, all other layers use the ReLU function, and the loss function consists of mean square reconstruction loss and classification cross-entropy loss. For the feature vectors U, V, and W from three channels, the classical trilinear interaction tensor is computed through the outer product. To avoid excessive computation and spatial complexity, this study adopts a low-rank projection strategy. Firstly, project U, V, and W into a low-dimensional space through independent linear layers. Subsequently, the dimensionality-reduced trilinear interaction is computed and

mapped to the final fused feature through flattening and a linear projection layer.

3.2 Construction of the image processing model based on TSC and MP

The TSC algorithm only completes image processing and recognition tasks by fusing existing feature maps. It does not extract feature maps on its own. You Only Look Once version 5 (YOLOv5) is a lightweight image recognition algorithm. Its backbone and neck are responsible for the layered extraction of image features and mainly contribute to improving recognition efficiency [19]. Therefore, this study proposes utilizing the front part of the YOLOv5 network to extract multi-scale feature maps from the target image for the fusion and recognition process in TSC. The trimmed YOLOv5 and the C3 module structure are shown in Figure 3.

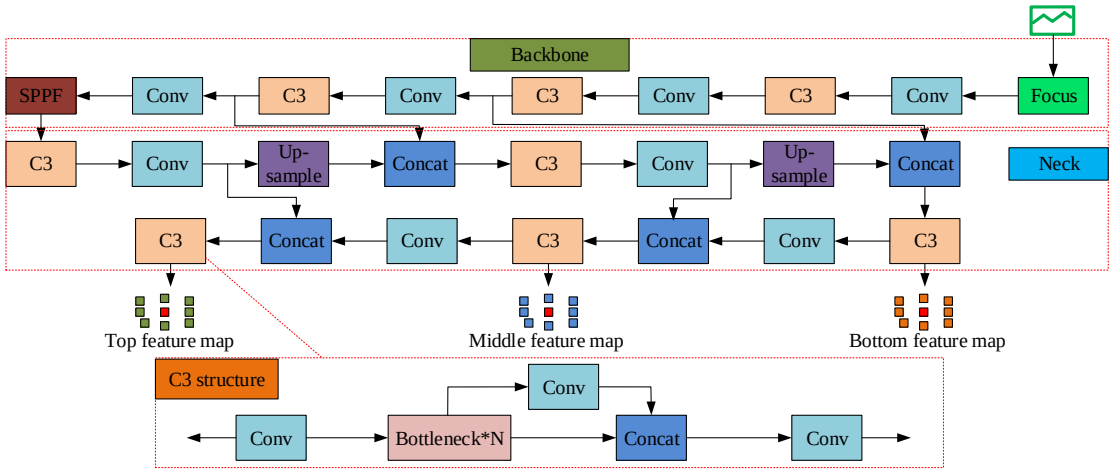


Figure 3: Structure diagram of the partial YOLOv5 network and the C3 module (Source from: author self-drawn)

In Figure 3, the trimmed YOLOv5 network includes a backbone and a neck. The modules in the backbone, such as SPPF, C3, and Conv, work together to extract image information. The main function of the neck is to extract the same feature information between adjacent layers and output it through the C3 module. The C3 module is composed of Conv, Concat, and N Bottleneck modules. The computation process in the Bottleneck module is shown in Equation (12).

$$y = \phi(W_2 * \gamma(W_1 * x + b_1) + b_2) + shortcut(x) \quad (12)$$

In Equation (12), ϕ and γ are activation functions. γ and b represent the weight matrix and bias, $shortcut$ represent the branch operations with different input dimensions. The computation process in the Concat module is shown in Equation (13).

$$C_{r,m,n} = \begin{cases} A_{r,m,n}, 0 \le r < C_A \\ B_{r,m,n}, C_A \le r < C_A + C_B \end{cases} \quad (13)$$

In Equation (13), r, m, and n represent the dimensions, height, and width used during the fusion of feature maps, A and B represent the two layers to be connected. These modules together enable the YOLOv5 backbone and neck to extract high-, mid-, and low-level feature maps. However, the three-channel image recognition method that integrates the front end of YOLOv5 with TSC combines multiple deep learning algorithms. It may increase the operational time cost. MP is a model optimization technique that enhances computational efficiency by evaluating the importance of different parameters and structures, and removing components with minimal impact on the results [20]. Pruning improves model feedback speed while maintaining a certain level of accuracy. This study adopts structured channel pruning as its core strategy, with the importance criterion based on the L1 norm amplitude of channel weights. The pruning plan is iterative, consisting of a total of three iterations. Each iteration prunes 15% of the global channel count, with a final target sparsity of approximately 40%. After each pruning, the model is fine-tuned for 5 epochs with a learning rate set to one-tenth of the initial learning rate. Through this process, the total number of parameters and Giga Floating point Operations Per Second (GFLOPs) of the trimmed model can be significantly reduced. In the trimmed YOLOv5s backbone network, channel simplification was mainly carried out on the C3 module and Conv module in Backbone and Neck. The number of output channels for the 2nd, 4th, and 6th C3 modules in the backbone network has decreased from 128, 256, and 512 to 96, 192, and 384, respectively. The C3 module, after the Concat layer used for feature fusion in the neck network, has reduced its output channel count from 256 to 192. All hidden layer channels in Bottleneck modules are retained in the same proportion. The detailed process is shown in Figure 4.

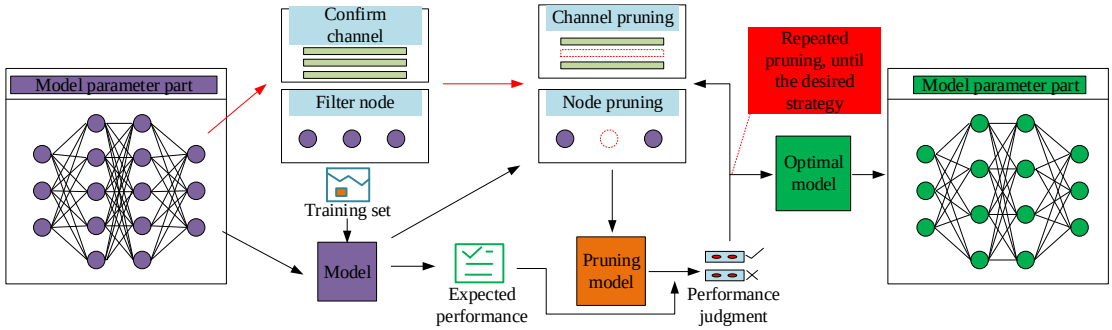


Figure 4: MP process flowchart (Source from: author self-drawn)

In Figure 4, the pruning process first selects redundant nodes and channels in the model. Then, it trains the model using the training set to obtain an expected accuracy. Next, the model removes the nodes and channels identified as unimportant. The pruned model is verified. If it reaches the expected accuracy, the model is output. If it does not meet the target, the process selects other nodes and channels to prune and verifies again until the accuracy meets expectations. During pruning, when the number of channels is reduced, the data may become unstable, and the accuracy may drop significantly. To address this issue, batch normalization is employed to adjust the data distribution after pruning, as illustrated in Equation (14).

$$x'_i = \frac{x_i - E[x]}{\sqrt{Var[x]}} \quad (14)$$

In Equation (14), x'_i is the final data. x_i is the input data. $E[x]$ is the mean value, and $Var[x]$ is the variance. To avoid data being overly uniform, the model data must be adjusted again, as shown in Equation (15).

$$y_i = \gamma x_i + \beta \quad (15)$$

In Equation (15), γ is the scaling factor, and β is the offset. Finally, the study integrates the YOLOv5 backbone, neck, and TSC algorithm to construct a pruned image processing model, named Y-TSC. The structure and workflow of the model are shown in Figure 5.

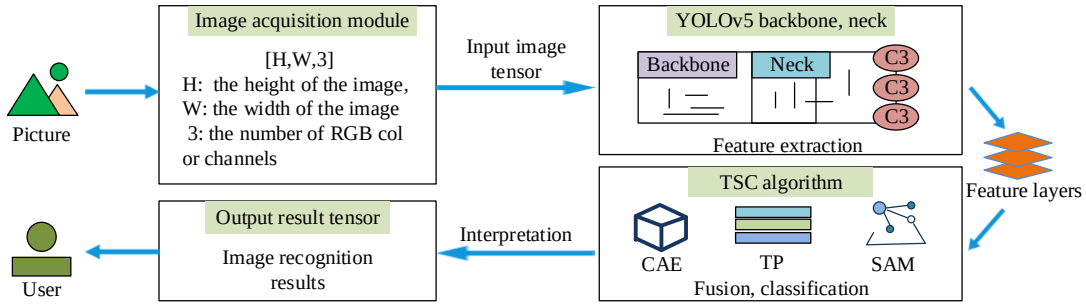


Figure 5: Pruned image processing model integrating YOLOv5 backbone and TSC (Source from: author self-drawn)

In Figure 5, the Y-TSC model after pruning consists of four components: an image acquisition module, a texture feature extraction module, a feature fusion and recognition module, and a result output device. The physical sensor uses cameras and other devices to convert real-world images into digital images that the algorithm can recognize. The texture feature extraction module is composed of the trimmed YOLOv5. It extracts different texture features from low, mid, and high levels to be used in the following TP process. The feature fusion and recognition module is composed of the TSC algorithm. This module applies the spatial attention mechanism to adaptively perform outer product operations on different-dimensional feature maps and finally completes the image recognition process. The result output device includes monitors, speakers, and other human–machine interface devices that convert digital information into physical signals that users can understand. This completes the final result output.

4 Performance validation of the TP-based image recognition model

4.1 Performance validation of the TP-based TSC algorithm

To verify the effectiveness of the Y-TSC model, the study utilized a balanced training subset of AudioSet for model training, which comprises approximately 22000 audio clips covering 527 categories. The validation set uses the Eval subset, which contains approximately 20000 samples. During the testing, 10 common sounds (such as bird songs, cars, natural environments, and human voices) were selected, and 100 samples were randomly selected from each category for recognition testing. Can be obtained from

https://research.google.com/audioset/. The study used Detection Transformer (DETR), Cascade Single Shot MultiBox Detector (CASSD), and Efficient Hybrid Backbone Object Detection (EHBD) as baselines. All baseline models were adjusted for spectral classification tasks and run under identical conditions to ensure fair comparisons. The specific implementation is based on a public repository. DETR utilizes PyTorch from Facebook Research (version 2.0) to adjust the model by replacing the detection header with a fully connected classification header. ResNet-50 and EfficientNet-B0 are implemented using the PyTorch vision library (version 0.15). All models maintain hyperparameter parity. The audio clips are uniformly converted to mono at a sampling rate of 16kHz, and a spectrogram is generated using the short-time Fourier transform. The window length is set to 1024, the hop count is set to 512, and the frequency range is set to 0-8kHz. Logarithmic Mel spectrograms (128 Mel filters) are used. The image resolution is uniformly adjusted to 224 × 224. Data augmentation includes time masking (maximum width: 20 frames) and frequency masking (maximum bandwidth: 10 Mel frequency bands), as well as normalization of mean and variance. Training was conducted for a total of 100 epochs using the Adam optimizer, with weight decay of 1e-4 and momentum parameters β1 and β2 set to 0.9 and 0.999, respectively. The learning rate employed a cosine annealing scheduling strategy, starting from an initial value of 1e-4 and reaching a minimum of 1e-6. Train using fixed random seeds (42) to ensure reproducibility of results, and initiate early stopping when the validation set loss does not decrease for 10 consecutive epochs. All report results are the average of 3 independent runs. The operating system was Ubuntu 22.04.3 LTS, the programming language was Python 3.9, and the deep learning framework was PyTorch-2.0. The CPU used was AMD Ryzen 9 7950X3D, and it is equipped with an NVIDIA GeForce RTX 4090 (24GB VRAM) GPU for model training and inference. When the training batch size is 32, the throughput of a single GPU is about 285 samples/second, with a peak memory usage of 18.2GB. It takes approximately 12.5 hours to complete a full 100-epoch training. The parameter settings for each algorithm are shown in Table 2.

In the training set, the accuracy, loss rate, and ROC curve of each method are shown in Figure 6.

Table 2: Algorithm parameter settings

Parameter	Description	Value
Lr	Learning rate	1e-4
Batch_size	Lot size	32
Hidden_dims_SA	Hidden layer dimension of the spatial attention module	256
Hidden_dims_CAE	Hidden layer dimension of the classification autoencoder	512
Kernel_size	Convolution kernel size	3
Concat_size	Fusion dimension	256
Feature_Dim_Per_Channel	Single-channel input feature dimension	800
Input_Sequence_Length	Model total input feature dimension	2400
Input_size	Input image size	640

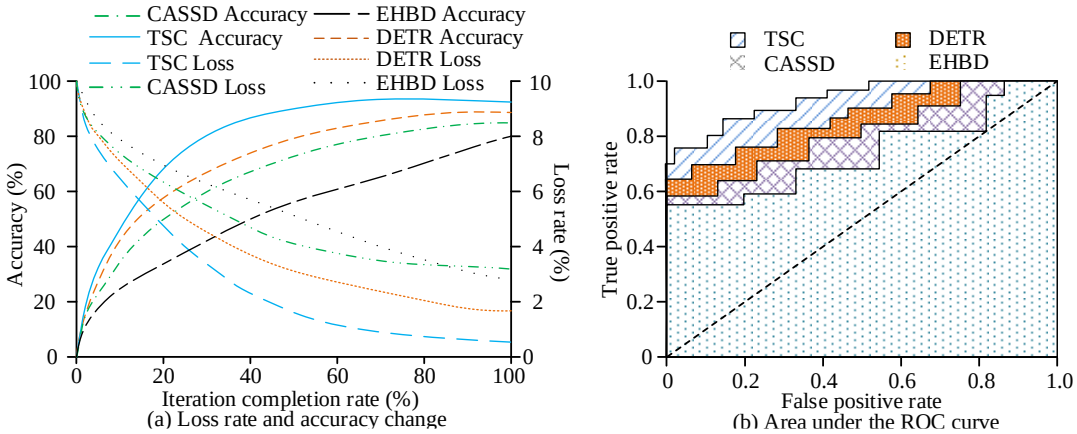


Figure 6: Comparison of accuracy (%), loss rate (%), and ROC curve of various algorithms on a balanced training subset (sample size 22000) (source from: author self-drawn)

In Figure 6(a), the TSC algorithm converged at 60% and achieved an accuracy of 92.7% with a loss rate of 0.5%. The DETR algorithm reached an accuracy of 89.2% and a loss rate of 1.8%. The CASSD algorithm showed an accuracy of 84.5% with a loss rate of 3.2%, while the EHBD algorithm had an accuracy of 80.0% and a loss rate of 1.4%. The accuracy of the standard classifiers ResNet-50 and EfficientNet-B0 under the same conditions was 85.6% and 87.3%, respectively, with loss rates of 2.0% and 1.7%, respectively. Figure 6(b) showed that the ROC curve of the TSC algorithm completely enclosed the curves of all comparison algorithms. The area under the TSC ROC curve was 0.942, compared to 0.875 for DETR, 0.791 for CASSD, and 0.741 for EHBD.

One hundred sound spectrograms were selected from different categories for classification tests. The sampling method involved randomly selecting 100 spectrograms from the test set of each category to ensure category balance and overall representativeness, without reuse. The comparison results are shown in Figure 7.

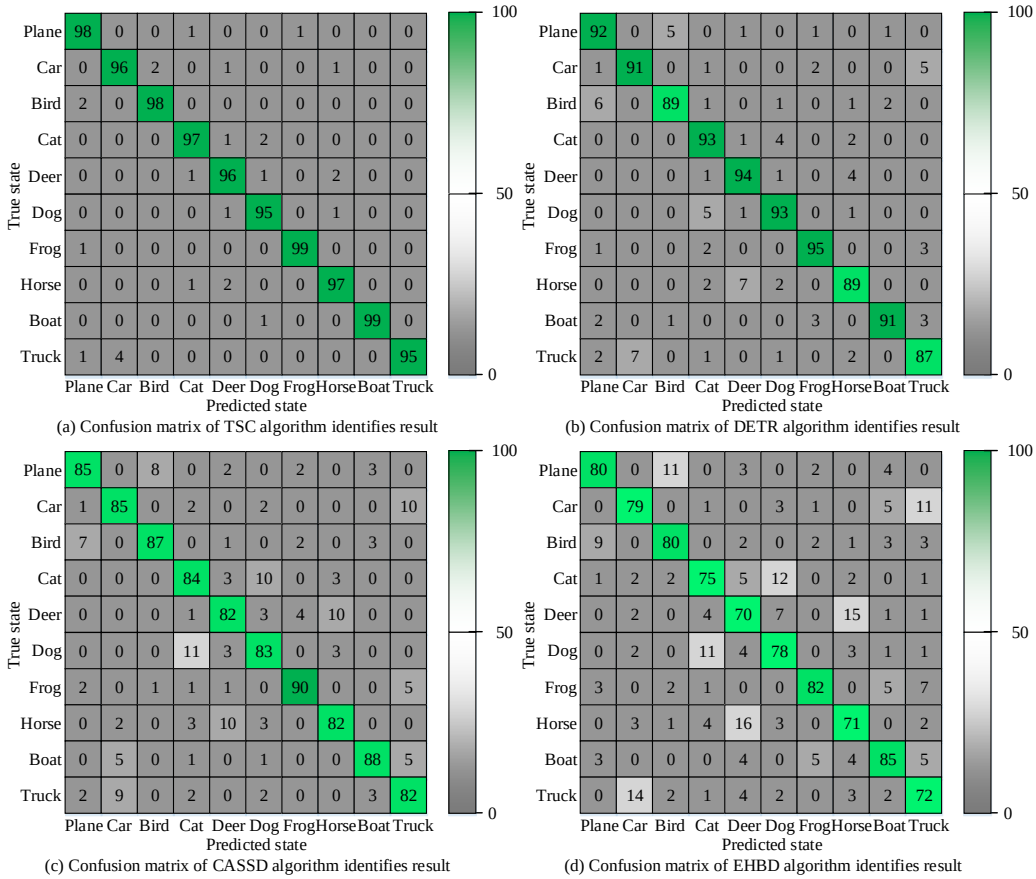


Figure 7: Classification test results of 100 spectrograms using various algorithms(Source from: author self-drawn)

As shown in Figure 7(a), the TSC algorithm correctly classified over 95% of the sound spectrograms in each category. Among them, the number of correctly classified sound spectrograms for frogs and boats was the highest, with both achieving a 99% accuracy. After the calculation, the average recognition rate of the TSC algorithm in all categories has reached 97.0%. Figure 7(b) showed that the DETR algorithm achieved an average recognition rate of 91.4%. Figures 7(c) and 7(d) showed that the CASSD and EHBD algorithms had average recognition rates of 84.8% and 77.2%, respectively. These results demonstrated that the TSC algorithm exhibited significantly better image recognition performance than the comparison algorithms.

To further verify the recognition ability of the TSC algorithm, this study selected and generated 1000 spectrograms for each category from the dataset (a total of 3000) to calculate the confidence interval, and compared the confidence intervals of each algorithm at different confidence levels. The confidence interval was calculated using the Wilson Score method, with 1000 test samples (instead of the original 100) for each category, to provide more stable interval estimates. To ensure the reliability of the confidence interval, the original probability output by the model was calibrated using the Platt scaling method to improve the consistency between the probability value and the actual confidence level. The results are shown in Table 3.

Table 3: Confidence intervals of dogs, cats, and birds under different confidence levels

Category	Image category	Accuracy (%)	Confidence			
			80%	85%	90%	95%
TSC	Bird sounds	95.3	(0.93,0.98)	(0.92,0.98)	(0.91,0.99)	(0.91,0.99)
	Car sounds	96.9	(0.91,1.00)	(0.90,1.00)	(0.89,1.00)	(0.88,1.00)
	Natural environment sounds	98.1	(0.92,1.00)	(0.92,1.00)	(0.91,1.00)	(0.90,1.00)
DETR	Bird sounds	93.1	(0.88,0.98)	(0.87,0.99)	(0.86,1.00)	(0.85,1.00)
	Car sounds	92.9	(0.86,0.94)	(0.86,0.94)	(0.85,0.95)	(0.84,0.96)
	Natural environment sounds	89.5	(0.76,0.90)	(0.75,0.91)	(0.74,0.92)	(0.72,0.94)
CASSD	Bird sounds	83.4	(0.76,0.90)	(0.75,0.91)	(0.74,0.92)	(0.72,0.94)
	Car sounds	84.5	(0.77,0.91)	(0.76,0.92)	(0.75,0.93)	(0.73,0.95)
	Natural environment sounds	86.8	(0.80,0.90)	(0.79,0.91)	(0.78,0.91)	(0.78,0.92)
EHBD	Bird sounds	78.69	(0.71,0.86)	(0.70,0.87)	(0.69,0.88)	(0.68,0.90)
	Car sounds	75.6	(0.68,0.82)	(0.67,0.83)	(0.66,0.84)	(0.67,0.83)
	Natural environment sounds	80.2	(0.75,0.85)	(0.74,0.86)	(0.73,0.87)	(0.72,0.88)

The accuracy in Table 3 is the Top-1 accuracy for each category, calculated based on the test samples of each category. The TSC algorithm achieved recognition accuracies of 95.3% for bird sounds, 96.9% for car sounds, and 98.1% for natural environment sounds. At the 95% confidence level, the confidence intervals were (0.91, 0.99) for bird sounds, (0.88, 1.00) for car sounds, and (0.90, 1.00) for natural environment sounds. In contrast, for the DETR algorithm, the confidence intervals at the same confidence level were (0.85, 1.00), (0.84, 0.96), and (0.72, 0.94), respectively. For the CASSD algorithm, the intervals were (0.72, 0.94), (0.73, 0.95), and (0.78, 0.92); and for the EHBD algorithm, they were (0.68, 0.90), (0.67, 0.83), and (0.72, 0.88). The upper limit of the partial confidence interval is close to 1.00, reflecting the high accuracy of the model in these categories. To avoid interval estimation bias caused by small sample sizes, this study employed the bootstrap sampling method (repeated 1000 times) to verify interval stability, and the results were consistent with the reports in the table.

4.2 Practical validation of the TP-based model

Based on the validation of the TSC algorithm, the performance of the Y-TSC model was further evaluated. A set of real-world images was used as the sample for testing. The performance of each model was assessed based on the quality of output images and processing time. Models based on DETR, CASSD, and EHBD were used for comparison. To verify whether the Y-TSC model had advantages in processing speed, a group of 200 images was used to test the processing efficiency of each model. The results are shown in Figure 8.

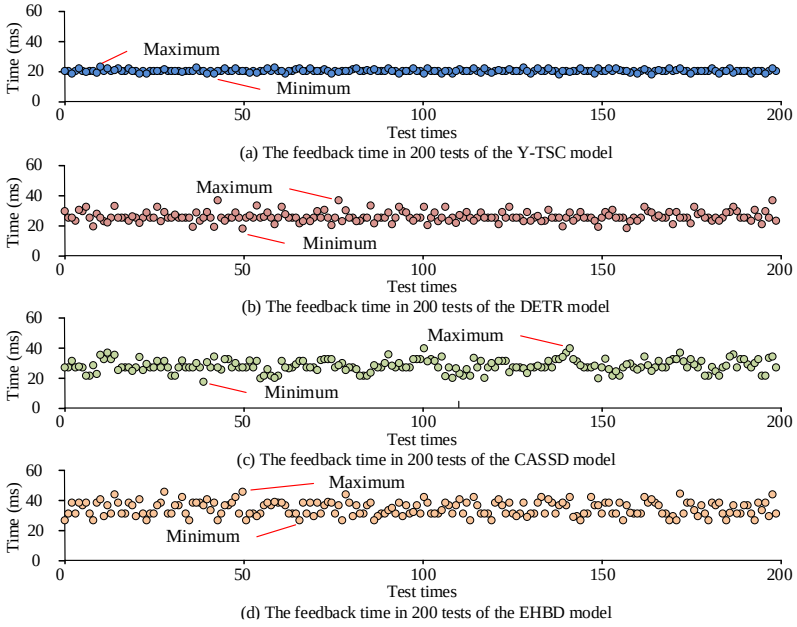


Figure 8: Feedback time of each model for 200 images (ms) (Source from: author self-drawn)

As shown in Figure 8(a), the feedback time of the Y-TSC model was concentrated around 20 ms, with a maximum of 23.1 ms and a minimum of 18.5 ms. In contrast, Figures 8(b), 8(c), and 8(d) showed that the response time distributions of the comparison models were more scattered. The DETR model had a maximum response time of 39.58 ms and a minimum of 18.6 ms. The CASSD model showed a maximum of 42.1 ms and a minimum of 18.6 ms, while the EHBD model ranged from 22.2 ms to 43.1 ms. These results indicated that the Y-TSC model significantly outperformed the comparison models in processing efficiency.

To further evaluate the accuracy of the Y-TSC model, the feature recognition accuracy of each model was compared across 200 tests, covering low-level, mid-level, and high-level features, as well as overall recognition accuracy. The results are shown in Figure 9.

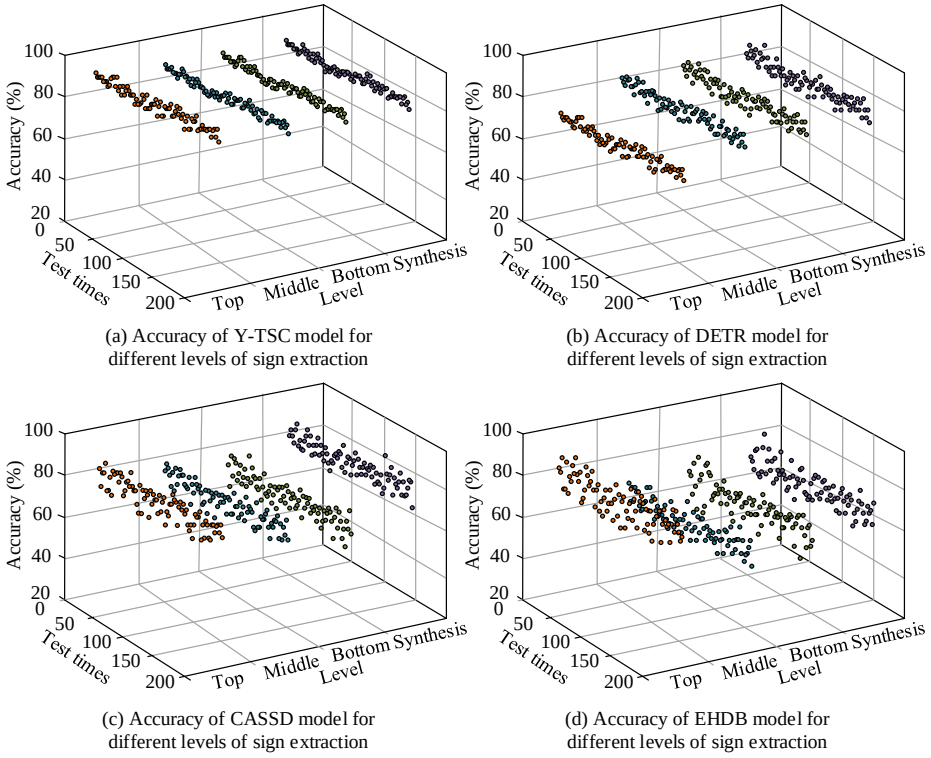
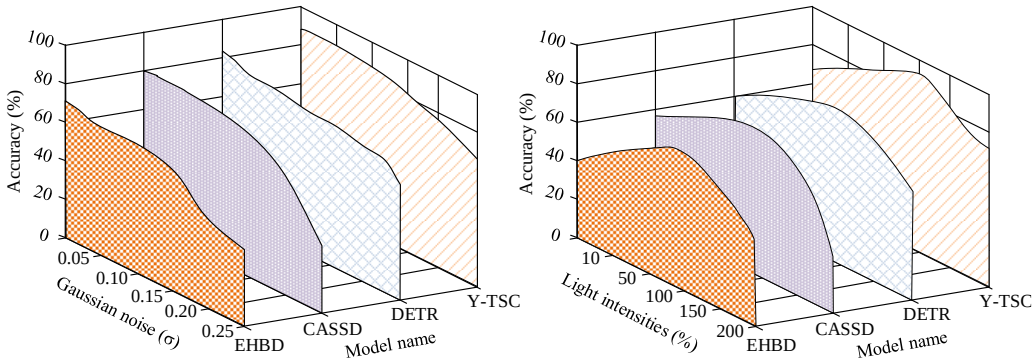


Figure 9: Recognition accuracy of each model on 200 images (%) (Source from: author self-drawn)

As shown in Figure 9(a), the Y-TSC model consistently maintained a feature recognition accuracy above 90% for each level, with an overall accuracy also at 90%. The test results were mostly aligned, indicating high model stability. In Figure 9(b), although the DETR model maintained an overall accuracy above 80%, its accuracy for low-level features remained around 70%, which might have limited its overall performance. Figures 9(c) and 9(d) showed that both the CASSD and EHBD models achieved significantly lower recognition accuracy across all layers, with more dispersed data points. These findings suggested that the Y-TSC model provided higher accuracy.

To verify the robustness of the models, all models were tested under different levels of Gaussian noise and lighting intensity. The results are presented in Figure 10.



(a) Accuracy under different Gaussian noise intensities (b) Accuracy under different light intensities intensities

Figure 10: Accuracy (%) of each model under different Gaussian noise and lighting conditions (Source from: author self-drawn)

As shown in Figure 10(a), the accuracy of all models decreased with increasing Gaussian noise. The accuracy of the Y-TSC model decreased from 90.2% at a 0.05σ noise level to 71.2% at a 0.25σ noise level. In comparison, the accuracy of the DETR model declined from 82.5% to 70.6%, the CASSD model from 79.5% to 40.7%, and the EHBD model from 70.7% to 52.3%. Figure 10(b) showed that all models experienced decreased accuracy under both low and high lighting conditions. At a 10% lighting intensity, the Y-TSC model achieved an accuracy of 69.2%, which was significantly higher than that of the comparison models. At 200% lighting intensity, the Y-TSC model still maintained a high accuracy of 71.2%, again surpassing the others. In summary, the Y-TSC model consistently outperformed the comparison models under high noise, highlighting, and low-lighting conditions, demonstrating stronger robustness.

Ablation experiments were conducted to quantitatively evaluate the independent contributions of each innovative component in the Y-TSC model. Five control models were designed in total: (a) Baseline: using only the trimmed YOLOv5 backbone network. (b) Add a spatial attention module on the basis of (a). (c) On the basis of (b), upgrade the feature fusion method from bilinear pooling to trilinear pooling; (d) On the basis of (c), introduce CAE to replace the original classifier; (e) Complete the Y-TSC model. Using average recognition accuracy, model parameter count, GFLOPs, and CPU latency as evaluation metrics. All models were independently run 5 times, presenting indicator results in the form of mean ± standard deviation. The results of the ablation experiment are shown in Table 4.

Table 4: Results of the ablation experiment

Models	Accuracy/%	Parameter count/M	GFLOPs	CPU latency/ms
a	84.5±0.3*	4.1	9.8	45.3±0.5*
b	86.1±0.4*	4.3	10.4	48.2±0.6*
c	89.2±0.2*	4.7	12.9	55.7±0.6*
d	91.8±0.3*	7.5	16.7	68.8±0.8*
e	92.7±0.2	4.5	10.1	50.4±0.5

Note: "*" indicates significant differences compared to the complete Y-TSC model, $P<0.01$.

As shown in Table 4, the recognition accuracy of model (a) is the lowest, only 84.5%. The recognition accuracy of the complete model (e) is the highest, reaching 92.7%, significantly higher than the other four models ($P<0.01$). And the parameter count, GFLOPs, and CPU latency of the complete model (e) are lower than those of the (d) model. This indicates that the components proposed by this study institute have made significant contributions to the model's performance, successfully achieving a balance between

accuracy and efficiency, resulting in the best overall performance.

5 Discussion

To address the challenge in existing image processing technologies of balancing both accuracy and efficiency, this study proposed a method for image recognition based on TP by analyzing image textures at three levels: low, middle, and high. The complete image recognition process was implemented through the proposed Y-TSC model, which integrated a Coupled Spatial Attention Mechanism, a CAE, and the YOLOv5 backbone. Experimental results showed that the TSC algorithm, used for image feature fusion and recognition, converged at 60% training progress with an accuracy of 92.7%, a loss rate of 0.5%, and a ROC value of 0.942. On the AudioSet dataset, the TSC algorithm achieved an average recognition accuracy of 97.0% across ten image categories. In the performance evaluation of the Y-TSC model, the feedback time never exceeded 23.1 ms, which was significantly lower than that of the comparison models. The Y-TSC model effectively extracted texture features from the low, middle, and high levels of images, enabling accurate image recognition. In the robustness tests, the Y-TSC model demonstrated strong resistance to Gaussian noise, maintaining an accuracy of 71.2% even under a noise intensity of 0.25σ. Under lighting conditions with only 10% of normal brightness, the model achieved an accuracy of 69.2%, and under 200% lighting intensity, it reached 71.2%.

The performance differences between this study and existing SOTA methods mainly stem from differences in architecture design, training strategies, and computational budgets. Compared to He et al. [6], who only used trilinear pooling for multimodal fusion without optimizing computational efficiency, this study introduces low-rank projection and structured pruning to reduce GFLOPs to 10.1 while maintaining feature interaction capability. Unlike Hu et al. [9], who relied on Jetson TX2 dedicated hardware to achieve real-time performance, this study achieved an inference speed of 23.1 ms on a general-purpose GPU through global channel pruning and progressive fine-tuning strategies. These improvements in architecture and training system have enabled the proposed Y-TSC model to achieve better overall performance under the same computational budget.

6 Conclusion

In summary, the proposed method and model successfully met the requirements of both accuracy and efficiency in the image recognition process, demonstrating a certain degree of resistance to interference. However, this study did not evaluate the resource and energy consumption involved in model construction, nor did it assess the economic feasibility of the recognition process. Future research will focus on addressing these issues and continuously improving and optimizing the model.

Funding

The research is supported by the Research Foundation of the Natural Science Foundation of Hunan Province, (Grant No. 2024JJ7189); the Social Science Project of Hunan Provincial Achievement Review Association, (Grant No.XSP24YBC319)；Hunan Province General Higher Education Teaching Reform Research Project (HNJG-20231101); Hunan Province General Higher Education Teaching Reform Research Project (HNJG-20231094).

References

[1] Putri R K, Athoillah M. Detection of facial mask using deep learning classification algorithm. Journal of Data Science and Intelligent Systems, 2024, 2(1): 58-63. https://doi.org/10.47852/bonviewJDSIS32021067

[2] Premnath S P, Gowr P S, Ananth J P, Arumugam S R. Image enhancement and blur pixel identification with optimization-enabled deep learning for image restoration. Signal, Image and Video Processing, 2024, 18(5): 4525-4540. https://doi.org/10.1007/s11760-024-03092-6

[3] Kumar HN N, Kumar A S, Prasad MS G, Shah M A. Automatic facial expression recognition combining texture and shape features from prominent facial regions. IET Image Processing, 2023, 17(4): 1111-1125. https://doi.org/10.1049/ipr2.12700

[4] SHI J, XU Y, CAO B. Fine-grained Image Classification Based on Adaptive Trilinear Pooling Network. Computer Engineering, 2023, 49(5): 239-246.

[5] Mpia H N, Syasimwa L M, Muyisa D M. Comparative machine learning models for predicting loan fructification in a semi-urban area. Archives of Advanced Engineering Science, 2025, 3(2): 124-134. https://doi.org/10.47852/bonviewAAES42022418

[6] He Y, Gan M G, Ma Q. Online video visual relation detection with hierarchical multi-modal fusion. Multimedia Tools and Applications, 2024, 83(24): 65707-65727. https://doi.org/10.1007/s11042-023-15310-3

[7] Sun R, Hu K, Martens K A E, Hagenbuchner M, Tsoi A C, Bennamoun M, Lewis S J G, Wang Z. Higher order polynomial transformer for fine-grained freezing of gait detection. IEEE transactions on neural networks and learning systems, 2023, 35(9): 12746-12759. https://doi.org/10.1109/TNNLS.2023.3264647

[8] Lee S B, Cho Y J, Yoon S H, Lee Y Y, Kim S H, Lee S, Cheon J. E. Automated segmentation of whole-body CT images for body composition analysis in pediatric patients using a deep neural network. European Radiology, 2022, 32(12): 8463-8472. https://doi.org/10.1007/s00330-022-08829-w

[9] Hu Y, Shuai Z, Yang H, Wan G, Zhang Y, Xie C, Lu X. ESDAR-net: towards high-accuracy and real-time driver action recognition for embedded systems. Multimedia Tools and Applications, 2024, 83(6): 18281-18307.

https://doi.org/10.1007/s11042-023-15777-0

[10] Grimm C, Fei T, Warsitz E, Farhoud R, Breddermann T, Haeb-Umbach R. Warping of radar data into camera image for cross-modal supervision in automotive applications. IEEE Transactions on Vehicular Technology, 2022, 71(9): 9435-9449. https://doi.org/10.1109/TVT.2022.3182411

[11] Wang S, Wang Z, Wang S, Ye Y. Deep image compression toward machine vision: A unified optimization framework. IEEE Transactions on Circuits and Systems for Video Technology, 2022, 33(6): 2979-2989. https://doi.org/10.1109/TCSVT.2022.3230843

[12] Vasanthi M, Seetharaman K. Facial image recognition for biometric authentication systems using a combination of geometrical feature points and low-level visual features. Journal of King Saud University-Computer and Information Sciences, 2022, 34(7): 4109-4121. https://doi.org/10.1016/j.jksuci.2020.11.028

[13] Hassan E, Shams M Y, Hikal N A, Elmougy S. The effect of choosing optimizer algorithms to improve computer vision tasks: a comparative study. Multimedia Tools and Applications, 2023, 82(11): 16591-16633. https://doi.org/10.1007/s11042-022-13820-0

[14] G Sahu, Seal A, Bhattacharjee D, Frischer R, Krejcar O. A novel parameter adaptive dual channel MSPCNN based single image dehazing for intelligent transportation systems. IEEE Transactions on Intelligent Transportation Systems, 2022, 24(3): 3027-3047. https://doi.org/10.1109/TITS.2022.3225797

[15] Frants V, Agaian S, Panetta K. QCNN-H: Single-image dehazing using quaternion neural networks. IEEE Transactions on Cybernetics, 2023, 53(9): 5448-5458. https://doi.org/10.1109/TCYB.2023.3238640

[16] Patil N. An enhanced segmentation technique and improved support vector machine classifier for facial image recognition. International Journal of Intelligent Computing and Cybernetics, 2022, 15(2): 302-317. https://doi.org/10.1108/IJICC-08-2021-0172

[17] Nath M, Mitra P, Kumar D. A novel residual learning-based deep learning model integrated with attention mechanism and SVM for identifying tea plant diseases. International Journal of Computers and Applications, 2023, 45(6): 471-484. https://doi.org/10.1080/1206212X.2023.2235750

[18] Puzyrev V, Elders C. Unsupervised seismic facies classification using deep convolutional autoencoder. Geophysics, 2022, 87(4): 125-132. https://doi.org/10.1190/geo2021-0016.1

[19] Zhou M, Wu L, Liu S, Li J. UAV forest fire detection based on lightweight YOLOv5 model. Multimedia Tools and Applications, 2024, 83(22): 61777-61788. https://doi.org/10.1007/s11042-023-15770-7

[20] Saadallah A, Jakobs M, Morik K. Explainable online ensemble of deep neural network pruning for time series forecasting. Machine Learning, 2022, 111(9): 3459-3487. https://doi.org/10.1007/s10994-022-06218-4