Machine Learning (2026) 115:61
https://doi.org/10.1007/s10994-026-07008-y

FuDoBa: Fusing Document and Knowledge Graph Based Representations with Bayesian Optimisation

Boshko Koloski[1,2] · **Senja Pollak**[1] · **Roberto Navigli**[3] · **Blaž Škrlj**[1]

Received: 23 April 2025 / Revised: 1 February 2026 / Accepted: 2 February 2026

Abstract
Building on the success of large language models (LLMs), LLM-based representations have dominated the document representation landscape, achieving strong performance on document embedding benchmarks. However, high-dimensional, computationally expensive LLM embeddings can be too generic or inefficient for domain-specific and resource-scarce applications. To address these limitations, we introduce FuDoBa—a Bayesian optimisation-based representation learning method that integrates LLM embeddings with domain-specific structured knowledge, sourced both locally and from external repositories such as WikiData. This fusion produces low-dimensional, task-relevant representations while reducing training complexity and yielding interpretable early-fusion weights for improved classification performance. We demonstrate the effectiveness of our approach on six datasets across two domains, showing that when paired with robust AutoML-based classifiers, our method performs on par with, or surpasses, proprietary LLM-only embedding baselines, while offering modality-wise interpretability and a smaller dimensional footprint.

Keywords Document classification · Bayesian optimisation · Representation learning · Knowledge graphs · Early fusion · Multimodal learning

Editors: Riccardo Guidotti, Anna Monreale, Dino Pedreschi.

✉ Boshko Koloski
boshko.koloski@ijs.si

✉ Blaž Škrlj
blaz.skrlj@ijs.si

Senja Pollak
senja.pollak@ijs.si

Roberto Navigli
navigli@diag.uniroma1.it

[1] Jožef Stefan Institute, Ljubljana, Slovenia

[2] Jožef Stefan Postgraduate School, Ljubljana, Slovenia

[3] Sapienza NLP Group, Sapienza University of Rome, Rome, Italy

Published online: 06 March 2026

1 Introduction

Efficient and rich document representations are the building blocks for many natural language processing (NLP) tasks such as classification or clustering (Muennighoff et al., 2023). Contemporary methods for representing documents focus on distilling representations from either pre-trained language models (PLMs) such as BERT (Devlin et al., 2019) or large language models (LLMs) such as Llama3 (Grattafiori, 2024), exploiting the rich semantic knowledge acquired during pre-training on vast text corpora. For instance, Sentence-BERT (Reimers et al., 2019) builds document representation by pooling over pre-trained BERT-based word embeddings, which are further refined through contrastive learning and Siamese networks. Similarly, LLM2Vec (Behnamghader et al., 2024) disentangles the causal masking of LLMs to a bi-directional one, further post-training the LLM on a masked next token prediction task and finally, training with a contrastive training objective, similarly to Sentence-BERT, refining the final representations via contrastive mean pooling.

Despite good performance on public benchmarks such as MTEB (Muennighoff et al., 2023), contrastive pre-training models require acquiring a dataset of triplet sentences (i.e., query, positive answer, and negative answer), which is often infeasible and costly. That said, even assuming such a dataset is available, training is costly and requires extra compute. The best-performing representations to date often come from proprietary companies and their mechanisms are largely unknown (Muennighoff et al., 2023). Current approaches face two notable limitations: first, these embeddings typically reside in high-dimensional spaces, often exceeding 1000 dimensions, creating practical challenges for efficient storage and computation (especially for AutoML model training); second, their generalist nature makes them suboptimal for domain-specific applications without expensive fine-tuning, which in some cases is infeasible.

One way to overcome the costly training requirements while still leveraging the expressive power of PLM/LLM derived representations is to introduce additional knowledge from auxiliary representations. Previous studies proposed representation enrichment via additional representations, either via representation alignment (Chen et al., 2022) or representation fusion (Koloski et al., 2022). In many cases, the approach of introducing auxiliary knowledge to document representations and searching over a space of possible classifiers with AutoML performs on par with task-specific fine-tuned PLMs such as BERT, without training costly classifiers (Škrlj et al., 2021b).

Recently, Koloski et al. (2024) proposed BabelFusion, involving the introduction of *global knowledge*from a knowledge graph by entity-linking to BabelNet's (Navigli et al., 2010) synsets connected to a subgraph of WikiData5m (Wang et al., 2021). They showed that projecting the inputs into low-dimensions not only produces low-dimensional spaces but can generate models that usually out-perform the language-only representations in the high dimensional space. However, they concluded that aligning short, online-discussion texts with a knowledge graph is suboptimal. Indeed, this proved deteriorating for the downstream performance of AutoML classifiers, in comparison to that of weak classifiers fitted on the original space. As an alternative to their approach, we hypothesise that extracting domain-specific knowledge graphs, resulting in a *local knowledge* graph,[1] can provide local context that could complement to a global knowledge graph such as WikiData, aiding LLMs downstream.

[1] We refer to them as LocKG throughout the remainder of the paper.

In this work, we propose *FuDoBa* (Fig. 1), a novel Bayesian optimisation-based early fusion *representation learning* methodology that builds on the idea of Koloski et al. (2024). It combines the semantic richness of LLM representations with the structured information from knowledge graphs and proposes the construction of local knowledge by extracting structured relations (e.g., knowledge triplets) from the dataset using Relation Extraction methods. Our approach systematically leverages external knowledge sources such as WikiData alongside these domain-specific, locally extracted knowledge structures to create more contextualised and task-relevant document representations. By employing Bayesian optimisation techniques, we identify importance parameters that maximize performance while minimising dimensionality. Furthermore, the optimised global weights assigned by Bayesian optimisation to the different representation sources serve as a task-specific interpretation of the embeddings' importance.

The main contributions of this work are as follows:

- We demonstrate that LLM-based embeddings can be further improved by integrating both global contextual information and fine-grained, domain-specific local knowledge. In particular, we systematically evaluate the impact of constructing and incorporating a domain-specific knowledge graph–capturing domain-dependant relations between entities.
- We experimentally demonstrate that training AutoML in low-dimensional spaces yields downstream models that perform as well as–or even better than–those produced using conventional methods.
- We introduce a novel, Bayesian optimisation–based method for low-dimensional early fusion of document and knowledge graph representations.
- We evaluate pre-trained tabular foundational models as multi-modal learners, showing that they can perform on-par with time-constrained AutoML models, on our proposed FuDoBa representations.

The remainder of this paper is structured as follows. Section 2 reviews the relevant literature, and Sect. 3 outlines the proposed methodology. Section 4 describes the experimental setup, and Sect. 5 presents the results. Section 6 discusses the findings, and Sect. 7 concludes the paper. The Appendix details the method's implementation specifics in Appendix A, a deeper analysis of relation extractors within the local knowledge graph in Appendix B, and additional experimental results in Appendix C.

2 Related Work

The task of document representation has evolved from pre-training shallow neural networks (NNs) on web scale data (Le & Mikolov, 2014) through transformer encoders (Devlin et al., 2019) refined by contrastive learning (Gao et al., 2021; Li & Li, 2024; Reimers et al., 2019), to adaptations of decoder-only LLMs (Behnamghader et al., 2024; Khosla et al., 2025) and synthetic data approaches (Wang et al., 2024). While these techniques output powerful high-performant embeddings (Muennighoff et al., 2023), significant challenges remain (Koloski et al., 2024). Training and adapting these large models to novel domains requires substantial computational resources (Reimers et al., 2019). Data acquisition remains labori-

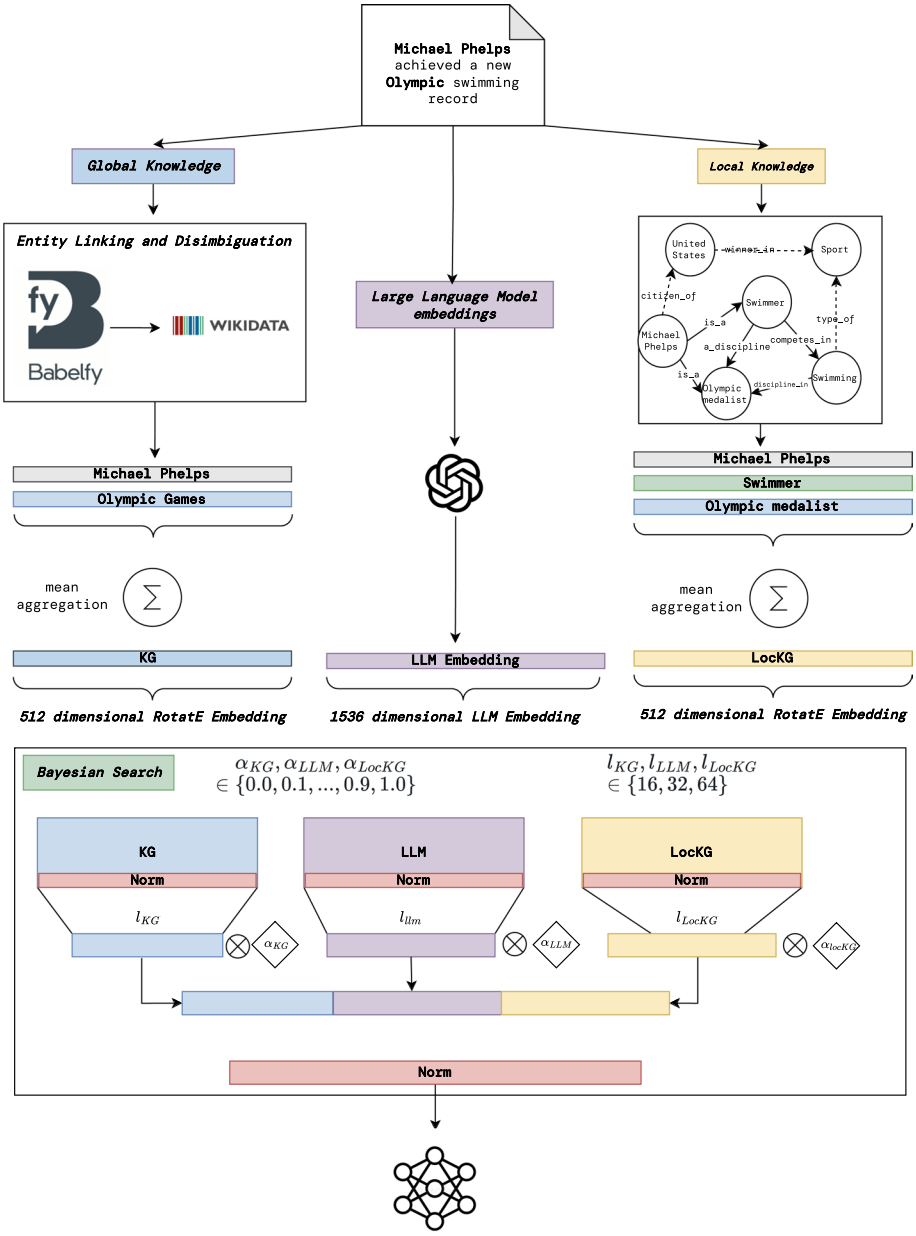

Fig. 1 Overview of our proposed *FuDoBa* framework for representation enrichment. Starting from an LLM-based embedding (purple) external knowledge is incorporated via two parallel pathways: a global knowledge graph (light blue) branch that links entities using BabelFy (Moro et al., 2014) and WikiData5m (Wang et al., 2021) embeddings, and a novel local knowledge graph branch (yellow) that constructs domain-specific knowledge-graph via relation-extraction models. Representations are projected into a lower-dimensional space (l) and weighted by an importance parameter (α), with ElasticNet normalisation applied both before and after fusion. The concatenated representation is then processed by a learning framework e.g. AutoGluon (Erickson et al., 2020) AutoML for model search under a constrained time budget (Color figure online)

ous and potentially costly (Wang et al., 2024). To avoid costly retraining, researchers have proposed utilising additional representations in LLM-based document representation. Chen et al. (2022) enhanced document classification through zero-shot alignment between labels and knowledge graph concepts. Recent work (Cocchi et al., 2025) demonstrated that adding reflection tokens to vision-language models with Wikipedia improves visual question answering performance. Tax2Vec (Škrlj et al., 2021a) leveraged the WordNet taxonomy to enhance TF-IDF features for short document classification. Koloski et al. (2022) introduced Sentence-Transformers (Reimers et al., 2019), sub-symbolic and WikiData5m (Wang et al., 2021) knowledge-graph embeddings, improving fake news classification over language-only representations. Ostendorff et al. (2019) proposed early-fusing a knowledge graph with BERT (Devlin et al., 2019) representations and refining them via a 2-layer MLP, showing remarkable results for multi-class classification. The benefits of local knowledge graphs (LocKG) have been noted in the context of information retrieval by Sarmah et al. (2024), who investigated how LocKG improves downstream performance in document search and retrieval. Similarly, Fan et al. (2019) applied LocKG to the task of multi-document summarization. However, all of the mentioned approaches operate in high-dimensional spaces, presenting challenges due to the curse of dimensionality (Aggarwal et al., 2001). Projecting these representations into low-dimensional space via SVD Škrlj and Petković (2021) by preserving top-k singular values can compress dimensionality and facilitate representation fusion (Koduri, 2012; Koloski et al., 2024). For efficient learning in these spaces, automated machine learning approaches like AutoGluon (Erickson et al., 2020) excel on benchmarks through dual optimisation of parameters and model ensembles. While previous work (Koloski et al., 2024) combines low-dimensional fusion of knowledge and text utilised tree-based models TPOT (Le et al., 2020), our work focuses on incorporating AutoGluon (Erickson et al.,2020), a stronger model which also searches neural models. As this work connects to the field of data fusion, particularly early-fusion, we refer interested readers to (Jiao et al., 2024; Lahat et al., 2015) for further details. The goal of this study is to explore whether, by employing modality-specific low-dimensional projection—including local and global knowledge graphs and utilising strong AutoML models—we can perform on par, or similarly to original high-dimensional embeddings. In Table 1, we compare our work with similar investigations. Notably, our approach introduces domain constructed local knowledge graphs, utilises stronger AutoML models, introduces modality specific interpretable weights, and does not require access to embedding model weights nor requires custom models.

3 Methodology

We introduce FuDoBa, a novel multimodal fusion framework that employs Bayesian optimisation to determine the optimal contribution of each modality's low-dimensional projection for improved classification performance.

3.1 Document Representations

Our approach builds on the BabelFusion representations (Koloski et al., 2024) by using LLM- and KG-based representations as the backbone for document representation. Koloski

Table 1 Comparison of different methods for leveraging knowledge graphs for LLM representation improvement

Feature	Text-KG	Het. Ens.	BabelFusion	FuDoBa
	Alignment (Chen et al., 2022)	(Koloski et al., 2022)	(Koloski et al., 2024)	(This work)
Model weights	✓	×	×	×
Word-disambiguation	×	×	BabelFy (Moro et al., 2014)	BabelFy (Moro et al., 2014)
Low-dimensional space	×	×	✓	✓
Modality importance	×	×	×	✓
Architecture	Zero-shot(*)	Custom NN	TPOT (Le et al., 2020)	AutoGluon (Erickson et al., 2020)
External KG	ConceptNet (Speer et al., 2017)	WikiData5m (Wang et al., 2021)	WikiData5m (Wang et al., 2021)	WikiData5m (Wang et al., 2021)
Local KG	×	×	×	✓

et al. (2024) observed that mapping textual content to a general-domain KG sometimes results in only a few matched entities for specific domains. To address this limitation, the solution our approach proposes is the construction of a local KG (LocKG) using relation extraction techniques. Our framework integrates three complementary embedding modalities:

- **Text Embeddings**[2] Embeddings are derived from OpenAI's `text-embedding-3-small` model and capture rich contextual information from text. They are represented in 1536 dimensions.
- **Knowledge Graph (KG) Embeddings** Generated via BabelFusion (Koloski et al., 2024), these embeddings map textual terms to WikiData5M (Wang et al., 2021) entities using BabelFy (Moro et al., 2014)'s entity linking and word sense disambiguation and aggregate them with 512-dimensional RotatE (Sun et al., 2019) embeddings.
- **Local Knowledge Graph (LocKG) Embeddings** These embeddings are obtained by extracting knowledge triplets using relation extractors and then embedding them with 512-dimensional RotatE (Sun et al., 2019) embeddings to capture localised relational structures. In our experiments, we employ prompt-guided `gpt-4o-mini` as an extractor.[3]

[2] We refer to these as LLM-based representations throughout the manuscript, as they are trained to learn semantic textual representations using common LLM pretraining objectives.

[3] Implementation details are present in Appendix A. In the Appendix B we show that this extractor can be changed for a smaller, open-sourced one, achieving similar results.

3.2 Preliminaries

Two key techniques further underpin our pipeline:

- **Dimensionality Reduction via Truncated SVD** Given a data matrix $\boldsymbol{A} \in \mathbb{R}^{N \times d}$, its truncated SVD (Klema & Laub, 1980) retains the top p singular components: $\boldsymbol{A} \approx \boldsymbol{U}_p \boldsymbol{\Sigma}_p \boldsymbol{V}_p^T$, where $\boldsymbol{U}_p \in \mathbb{R}^{N \times p}$, $\boldsymbol{\Sigma}_p \in \mathbb{R}^{p \times p}$ (diagonal), and $\boldsymbol{V}_p \in \mathbb{R}^{d \times p}$. The columns of $\boldsymbol{V}_p$ represent the principal directions of variation in the column space. Projecting the original data onto these directions yields the reduced-dimension representation $\boldsymbol{P} = \boldsymbol{A}\boldsymbol{V}_p \in \mathbb{R}^{N \times p}$, which captures the maximal variance in p dimensions.
- **Normalisation** We employ normalisation techniques before and after fusion. The L_k -norm of a vector $\boldsymbol{x} \in \mathbb{R}^d$ is defined as $\|\boldsymbol{x}\|_k = (\sum_{i=1}^{d} |x_i|^k)^{1/k}$. We utilize Elastic Net style normalisation, defined for a vector $\boldsymbol{x}$ as:

$$N(\boldsymbol{x}) = \frac{\boldsymbol{x}}{w_1 \|\boldsymbol{x}\|_1 + w_2 \|\boldsymbol{x}\|_2},$$

where $w_1, w_2 \geq 0$ are weighting factors (e.g., $w_1 = w_2 = 0.5$). This balances the regularising effects of L1 and L2 norms.

3.3 FuDoBa: Fusion Mechanism and Parameter Optimisation

Koloski et al. (2024) demonstrated that projecting KGs enriched high-dimensional LLM-embeddings to lower-dimensions have a comparable or better performance than base representations. However, their approach involved concatenating modalities before projection and weighting features with equal weights post-projection. We argue that by doing so information was weighted equally, which can impact performance while projecting, as different signals might be important for different tasks. We propose a different strategy where each modality's embedding, $\boldsymbol{E}_m$ ($m \in \{\text{llm}, \text{kg}, \text{loc}\}$), is first projected independently via Truncated SVD (Klema & Laub, 1980) to a lower dimension l_m, yielding $\boldsymbol{P}_m = \text{SVD}_{p_m}(\text{Normalize}(\boldsymbol{E}_m))$. This step aims to capture the most important information from each modality separately. Subsequently, each projection $\boldsymbol{P}_m$ is scaled by a factor α_m before concatenation. We hypothesise that the scaling factor α_m per modality can act as a learned global importance weight for the modality's contribution within the fusion, potentially improving performance and offering insights into the relative importance of modalities for a given task,[4]

The fusion process is governed by a hyper-parameter vector $\boldsymbol{\theta}$, which comprises the projection dimensions and scaling factors for all modalities. In the case of the LLM modality, l_{llm} denotes the projection dimension, while α_{llm} represents the corresponding importance weight; thus, the full parameter space can be described as

$$\boldsymbol{\theta} = (\underbrace{l_{\text{llm}}, l_{\text{kg}}, l_{\text{loc}}}_{\text{projection dimensions}}, \underbrace{\alpha_{\text{llm}}, \alpha_{\text{kg}}, \alpha_{\text{loc}}}_{\text{modality importance}}).$$

[4] N.B. when a α is equal to zero (0.0) the modality is excluded from the learning process as uninformative.

We define the search space Θ for these hyper-parameters as follows: each projection dimension l_m is chosen from the discrete set $\{16, 32, 64\}$, and each scaling factor α_m is sampled from a quantised uniform distribution over $[0, 1]$ with quantisation step $q = 0.1$.

Algorithm 1 FuDoBa: low-dimensional multi-modal embedding fusion

Require: Modality Embeddings $\{\boldsymbol{E}_m\}_{m\in\mathcal{M}}$ ($\mathcal{M} = \{\text{llm}, \text{kg}, \text{loc}\}$), Labels $\boldsymbol{y}$
Require: Importance weights $\{\alpha_m\}_{m\in\mathcal{M}}$ and Projection dimensions $\{l_m\}_{m\in\mathcal{M}}$
Require: Classifier $\mathcal{C}$, Cross-Validation folds k
Ensure: Mean CV Macro F1-score f, Final classifier $\mathcal{C}$ (fit on full data)
— *Step 1: Generate Fused Representation* —
for $m \in \mathcal{M}$ **do**
  $\boldsymbol{E}_m^{\text{norm}} \leftarrow N(\boldsymbol{E}_m)$ ▷ Elastic Net normalisation
  $\boldsymbol{P}_m \leftarrow \text{SVD}_{l_m}(\boldsymbol{E}_m^{\text{norm}})$ ▷ Truncated SVD to dimension l_m
  $\boldsymbol{S}_m \leftarrow \alpha_m \cdot \boldsymbol{P}_m$ ▷ Scale projection by importance weight α_m
end for
$\boldsymbol{S}_{\text{concat}} \leftarrow [\boldsymbol{S}_{\text{llm}}, \boldsymbol{S}_{\text{kg}}, \boldsymbol{S}_{\text{loc}}]$ ▷ Concatenate scaled embeddings
$\boldsymbol{X} \leftarrow N(\boldsymbol{S}_{\text{concat}})$ ▷ Normalize fused representation
— *Step 2: Evaluate via Cross-Validation* —
$f_{\text{cv}} \leftarrow 0$
for $i \in \{1, \ldots, k\}$ **do** ▷ k-fold Cross-Validation loop
  $(\boldsymbol{X}_{\text{train},i}, \boldsymbol{y}_{\text{train},i}), (\boldsymbol{X}_{\text{val},i}, \boldsymbol{y}_{\text{val},i}) \leftarrow$ Get fold i split from $(\boldsymbol{X}, \boldsymbol{y})$
  $\mathcal{C}.\texttt{fit}(\boldsymbol{X}_{\text{train},i}, \boldsymbol{y}_{\text{train},i})$ ▷ Fit classifier on training part of fold i
  $\boldsymbol{y}_{\text{val},i}^{\text{pred}} \leftarrow \mathcal{C}.\texttt{predict}(\boldsymbol{X}_{\text{val},i})$ ▷ Predict on validation part of fold i
  $f_i \leftarrow \text{Macro-F1}(\boldsymbol{y}_{\text{val},i}, \boldsymbol{y}_{\text{val},i}^{\text{pred}})$ ▷ Calculate Macro F1-score for the fold
  $f_{\text{cv}} \leftarrow f_{\text{cv}} + f_i$
end for
$f \leftarrow f_{\text{cv}}/k$ ▷ Mean CV F1-macro score
$\mathcal{C}.\texttt{fit}(\boldsymbol{X}, \boldsymbol{y})$ ▷ Re-fit on the full dataset $\boldsymbol{X}, \boldsymbol{y}$
return $\mathcal{C}$, f ▷ Return the final fitted classifier and the CV score

Algorithm 1 details the complete procedure for generating a fused representation $\boldsymbol{X}(\boldsymbol{\theta})$ given a specific hyper-parameter vector $\boldsymbol{\theta}$, and subsequently evaluating its quality by computing the fivefold cross-validation macro F1-score of an AutoGluon (Erickson et al., 2020) model, denoted as $f(\boldsymbol{\theta})$. This score serves as our objective function, which we try to maximize, conditioned on the input parameters θ.

3.4 Bayesian Optimisation for Optimal Fusion Parameters

Having defined the parametrized fusion mechanism and the objective function $f(\boldsymbol{\theta})$ based on cross-validation performance of the fitted classifier model (Algorithm 1), our goal is to find the optimal hyper-parameters $\boldsymbol{\theta}^*$ that maximize this objective: $\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}\in\Theta} f(\boldsymbol{\theta})$. Since evaluating $f(\boldsymbol{\theta})$ is computationally expensive, we employ Bayesian optimisation (BO) (Snoek et al., 2012) to efficiently search the hyper-parameter space Θ. BO iteratively builds a probabilistic surrogate model of the objective function and uses an acquisition function to guide the selection of subsequent hyper-parameters to evaluate. We model $f(\boldsymbol{\theta})$ using a Gaussian Process (GP) prior (Rasmussen & Williams, 2005):

$$f(\boldsymbol{\theta}) \sim \mathcal{GP}\big(\mu(\boldsymbol{\theta}), k(\boldsymbol{\theta}, \boldsymbol{\theta}')\big),$$

where $\mu(\boldsymbol{\theta})$ is the mean function and $k(\boldsymbol{\theta}, \boldsymbol{\theta}')$ is a kernel function, in our case the Matérn Kernel (Rasmussen & Williams, 2005). Given n observations of previous hyper-parameter evaluations and their corresponding cross-validation Macro-F1-scores, $\mathcal{D}_n = \{(\boldsymbol{\theta}_i, f(\boldsymbol{\theta}_i))\}_{i=1}^n$, the GP yields a posterior predictive distribution $P(f(\boldsymbol{\theta})|\mathcal{D}_n) = \mathcal{N}(\mu_n(\boldsymbol{\theta}), \sigma_n^2(\boldsymbol{\theta}))$. Let $f^* = \max_{1 \le i \le n} f(\boldsymbol{\theta}_i)$ be the current best observed cross-validation Macro-F1-score. We use the Expected Improvement (EI) acquisition function to select the next point:

$$\mathrm{EI}(\boldsymbol{\theta}) = \underbrace{(\mu_n(\boldsymbol{\theta}) - f^*)\,\Phi(Z)}_{\text{exploitation}} + \underbrace{\sigma_n(\boldsymbol{\theta})\,\phi(Z)}_{\text{exploration}}, \quad \text{where } Z = \frac{\mu_n(\boldsymbol{\theta}) - f^*}{\sigma_n(\boldsymbol{\theta}) + \epsilon}.$$

here, $\mu_n(\theta)$ is the predicted mean of $f(\theta)$ based on the current observations $\mathcal{D}_n$, while $\sigma_n(\theta)$ is the predicted standard deviation, representing the uncertainty of that prediction. Φ and ϕ denote the standard normal cumulative distribution function (CDF) and probability density function (PDF), respectively, and ϵ is a small constant added for numerical stability. The Expected Improvement (EI) acquisition function balances exploitation–selecting points with high predicted performance–and exploration–selecting points with high uncertainty.

The next hyper-parameter configuration to evaluate is chosen by maximising the acquisition function:

$$\boldsymbol{\theta}_{n+1} = \arg\max_{\boldsymbol{\theta} \in \Theta} \mathrm{EI}(\boldsymbol{\theta}).$$

This iterative BO procedure allows for efficient exploration of the complex interplay between projection dimensions and importance weights, guiding the search towards an optimal fusion strategy.

4 Experimental Setting

In this section, we detail the experimental setting. We limit ourselves to the problem of classification, we describe the datasets and the experimental questions in Sects. 4.1 and 4.2, respectively.

4.1 Datasets

Following similar evaluation strategies as Koloski et al. (2024), we evaluate our proposed method on six distinct datasets across two classification tasks: sentiment analysis and news genre classification. The sentiment analysis datasets include the Amazon Reviews collection (specifically, Books, Dvd, and Music subforums) and a hate speech detection dataset comprising social media posts (Ranasinghe et al., 2020). The news genre datasets are MLDoc (Schwenk & Li, 2018) (four genres) and the more recent XGenre (Kuzman et al., 2022) (nine genres). Four datasets involve binary sentiment classification, while two address multi-class news categorisation. All datasets are in English. For consistency and

reproducibility, all experiments utilize the original train-test splits accompanying each dataset. Table 2 provides detailed statistics for each dataset, including the number of train/test documents and average word counts.

4.2 Experimental Setup

We design experiments to (i) quantify the value of enriching LLM document embeddings with structured knowledge, (ii) evaluate whether our low-dimensional, modality-weighted fusion (FuDoBa) preserves or improves downstream performance relative to training on high-dimensional features, and (iii) assess how the choice of downstream tabular classifier affects performance and the fusion configurations selected by Bayesian optimisation. All experiments use the original train–test splits of each dataset and report Macro-F1 unless stated otherwise.

E1: End-to-end classification with high-dimensional features. We first measure the downstream impact of adding global and/or local knowledge to LLM embeddings without dimensionality reduction. Specifically, we compare **LLM** against **LLM+KG**, **LLM+LocKG**, and **LLM+KG+LocKG**, where KG features are obtained via entity linking to WikiData5M and LocKG features are obtained via relation extraction followed by RotatE embeddings. A single downstream learner is trained directly on each feature space. This experiment addresses:

- **RQ1** Does enriching LLM embeddings with global and/or local knowledge improve downstream classification performance?
- **RQ2** When improvements occur, how do global KG and local KG contributions differ across datasets?

E2: Low-dimensional fusion and representation learning. Next, we evaluate whether low-dimensional fusion can match or exceed the performance of training on the original high-dimensional representations. We compare: (i) **FuDoBa**, which projects each modality independently, applies modality weights α_m, and concatenates the weighted projections; and (ii) **FuDoBa-CP** (concat-then-project), where modalities are concatenated first and then projected to a fixed dimension. This experiment addresses:

Table 2 Overview and statistics of the datasets used, similarly to (Koloski et al., 2024)

Dataset	Domain	Classes	Train size	Test size	Avg. length (words)
Books	Sentiment	2	2000	2000	155.80
Dvd	Sentiment	2	2000	2000	161.29
Music	Sentiment	2	2000	2000	130.12
Hatespeech (Ranasinghe et al., 2020)	Sentiment	2	13,240	860	22.85
MLDoc (Schwenk & Li, 2018)	News	4	11,000	4000	235.15
Xgenre (Kuzman et al., 2022)	News	9	1650	272	1256.92

- **RQ3** Can FuDoBa's fused, low-dimensional representations match or exceed performance achieved in high-dimensional feature spaces?
- **RQ4** Is joint optimisation of per-modality projection (FuDoBa) sizes and modality weights superior to concat-then-project fusion (FuDoBa-CP)?

E3: Few-shot learning behaviour. To understand how FuDoBa behaves under limited supervision, we repeat the LLM vs FuDoBa comparison under increasing label fractions. This experiment addresses:

- **RQ5** How does FuDoBa compare to LLM-only representations in few-shot learning regimes?

E4: Impact of downstream learner. Finally, to test whether FuDoBa is tied to a particular modelling bias, we rerun the FuDoBa optimisation procedure using different downstream learners (AutoGluon, TabPFNv2 (Hollmann et al., 2025), TabICL (Qu et al., 2025), RealMLP (Holzmüller et al., 2024), XGBoost (Chen et al., 2016)) as the evaluation function. This experiment addresses:

- **RQ6** How sensitive are the learned fusion configurations and resulting performance to the choice of downstream classifier?

Training protocol and optimisation budget. AutoGluon (Erickson et al., 2020) is used as the default downstream classifier . For FuDoBa, we run Bayesian optimisation for 50 evaluations and score each candidate configuration using 5-fold cross-validation Macro-F1. Each AutoGluon evaluation is constrained to a 5 min budget on a commodity CPU machine (Appendix A). To account for randomness in Bayesian optimisation and AutoML search, we repeat experiments with three random seeds and report mean and standard deviation.

5 Results

We proceed by presenting our main results. Section 5.1 details the downstream performance, Section 5.2 analyses few-shot performance, Section 5.3 assesses the importanceof classifier selection, Section 5.4 examines the impact of dimensionality and modalityimportance, and Section 5.5 analyses coverage and the extracted graph. Additionalin-depth results are provided in the Appendices.

5.1 End-to-End Classification

Figure 2 and Tables 3, 11 present our end-to-end classification results. We compare baseline LLM embeddings against methods incorporating Global and/or Local KGs, evaluating both simple concatenation (FuDoBa-CP) and our proposed FuDoBa approach.

We find that on average, LLM-enriched representations with both local and global knowledge, boost performance compared to LLM-only representations. Furthermore, we find that on average, the best performing representation is the one utilising knowledge from both, local and global setting. We find that the enriched LLM representation (LLM+KG+LocKG)

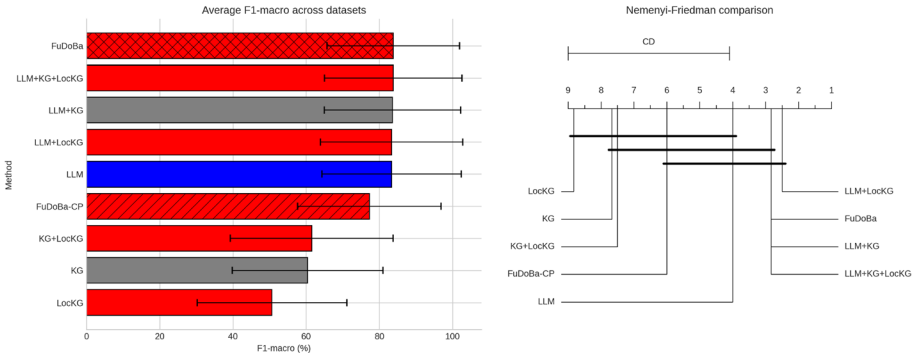

Fig. 2 Left: Average F1-score and ranking performance across datasets for AutoGluon-trained high-dimensional multi-modal representations versus the proposed FuDoBa method. The left panel shows that FuDoBa outperforms the strong LLM baseline (blue), achieving highest average Macro F1 and close to its high-dimensional variant. Representations in red incorporate local knowledge, a novel contribution of this work, while those in gray leverage global knowledge from WikiData as described in (Koloski et al., 2024). Right: Critical-difference diagram of average Macro-F1 ranks over $N = 6$ datasets for $K = 9$ methods. Overall differences are significant (Friedman test, $p < 0.001$). Nemenyi post-hoc comparisons control FWER at $\alpha = 0.05$ with $\mathrm{CD} = 4.904$; methods connected by a bar are not significantly different (Color figure online)

Table 3 Aggregated results across representations, best performing results per dataset are bolded, second best are italic

	Books	Dvd	Hatespeech	MLdoc	Music	Xgenre	Avg. F1	Avg. rank
Single modality								
LLM	93.75 ± 0.27	*93.62 ± 0.37*	75.77 ± 1.64	96.45 ± 0.14	92.73 ± 0.23	47.59 ± 0.67	83.32	4.00
KG	60.20 ± 0.50	63.85 ± 0.60	61.57 ± 0.56	88.58 ± 0.35	63.67 ± 0.14	24.35 ± 2.18	60.37	7.67
LocKG	64.44 ± 0.29	62.71 ± 0.60	54.89 ± 1.62	50.11 ± 0.55	61.36 ± 0.76	10.41 ± 0.65	50.65	8.83
High dimensional fusion								
LLM+KG	93.48 ± 0.51	92.73 ± 0.59	*76.81 ± 0.40*	96.53 ± 0.27	**93.52 ± 0.08**	*48.36 ± 1.35*	83.57	*2.83*
LLM+LocKG	**93.78 ± 0.48**	**93.82 ± 0.08**	76.21 ± 0.24	*96.77 ± 0.34*	92.85 ± 0.26	46.64 ± 0.29	83.35	**2.50**
KG+LocKG	65.56 ± 1.91	67.35 ± 0.60	60.24 ± 0.82	87.47 ± 0.75	68.20 ± 0.60	20.30 ± 3.38	61.52	7.50
LM+KG+LocKG	92.98 ± 0.15	93.60 ± 0.53	**78.43 ± 0.27**	**97.00 ± 0.20**	92.75 ± 0.58	47.83 ± 0.47	83.77	*2.83*
Low dimensional fusion								
FuDoBa-CP	87.53 ± 0.72	87.88 ± 0.66	66.60 ± 1.29	91.78 ± 0.73	87.95 ± 0.26	41.76 ± 4.18	77.25	6.00
FuDoBa	*93.77 ± 0.50*	93.53 ± 0.28	75.88 ± 0.38	96.51 ± 0.25	*92.98 ± 0.28*	**50.04 ± 0.16**	**83.78**	*2.83*

achieves perfromance on par with FuDoBa. To assess if the differences are significant, we use Friedman test with Nemenyi post-hoc correction following Demšar (2006) for robustness and standard practice in multi-dataset comparisons.[5] We find that our method performs on-par or better compared to the LLM-based representations, with both representations being in the same ranking space with no statistically significant difference between the two. Similarly, we find no statistifically significant difference between LLM-only and FuDoBa representations. While simple high-dimensional concatenation generated the best performing score, the resulting space is high-dimensional and often infeasible to store. Compared to that FuDoBa, operates in low-dimensional space, offering better footprint.

Finally, Fig. 3 plots the F1 performance difference ($\Delta\mathrm{F1} = \mathrm{FuDoBa} - \mathrm{LLM}$) against the relative dimensionality ($\Delta \log_2 \mathrm{Dimension}$). The visualisation demonstrates that, despite operating in a substantially lower-dimensional space, FuDoBa achieves comparable performance to high-dimensional counterparts, and often exceeds them.

Against the standard LLM (left), *FuDoBa* achieves a mean performance gain of $0.47\% \pm 0.96$, with specific tasks like *Xgenre* showing significant improvements. Even when compared against the more complex LLM+KG+LocKG pipeline (right), which operates in even higher dimensionality, the mean $\Delta\mathrm{F1}$ remains positive ($0.03\% \pm 1.55$). This confirms that *FuDoBa* representations are not only more efficient but are capable of surpassing the performance of input spaces that are over an order of magnitude larger, effectively overcoming the dimensionality gap.

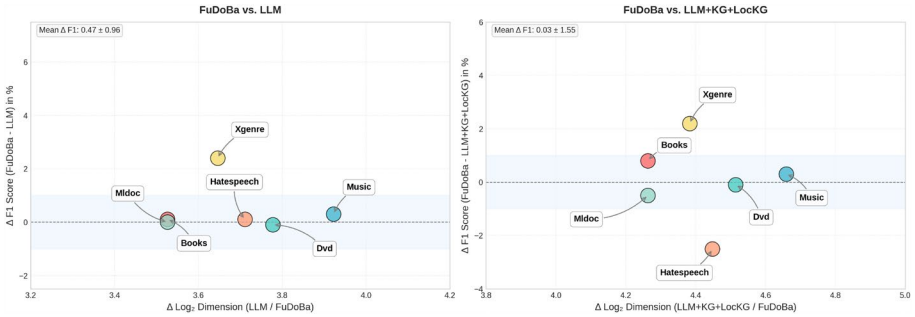

Fig. 3 F1-score difference versus difference in $\log_2$ of final dimensionality. Left FuDoBa versus LLM baseline. Right FuDoBa versus LLM+KG+LocKG with all knowledge sources integrated in high dimensions. The blue shaded area indicates practical equivalence where F1-scores differ by less than one percentage point. Despite operating in significantly lower-dimensional spaces (101–133 vs. 1536–2560 dimensions), FuDoBa achieves comparable or superior performance (Color figure online)

[5] We use the Friedman test because it is a non-parametric alternative to repeated-measures ANOVA and is appropriate when comparing multiple models across the same datasets or folds. It does not assume normality of performance differences and instead relies on rankings, making it well-suited for typical machine-learning evaluation settings (as recommended by Demšar (2006)). Once the Friedman test indicates a statistically significant difference among models, we apply the Nemenyi post-hoc test to perform pairwise comparisons while controlling for the increased risk of Type I errors due to multiple testing.

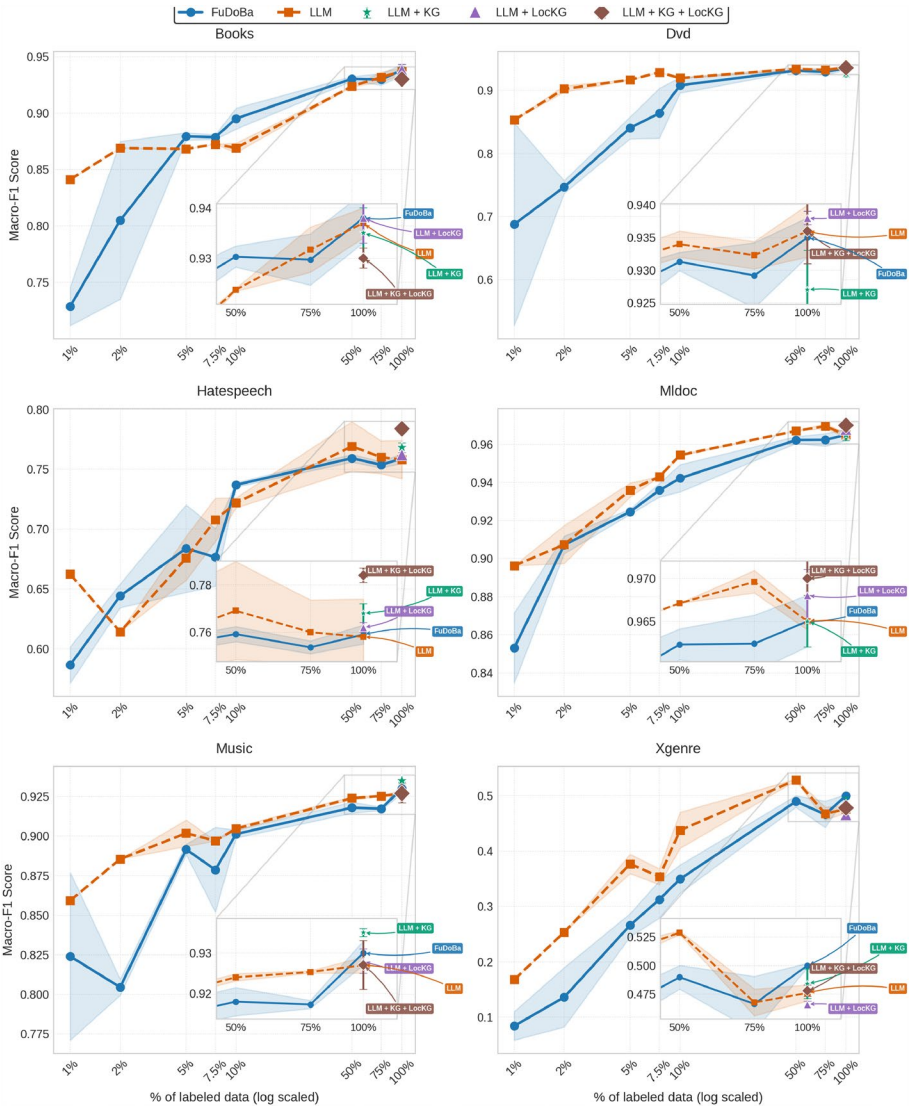

Fig. 4 Few shot-classification comparison between LLM-based and FuDoBa optimised representations, both trained with AutoGluon. The inset shows the final standings on full data

5.2 Few-Shot Learning

Figure 4 reports few-shot learning curves (Macro-F1) for FuDoBa versus an LLM-only baseline across the six English datasets.

In the ultra-low supervision regime (e.g., 1–2% labeled data, corresponding to only 20–40 labeled instances in the Amazon datasets and ≈16–33 in XGenre), the LLM baseline is often strongest, indicating that compressing representations can initially reduce separability when supervision is extremely scarce. However, FuDoBa closes the gap rapidly as

labeled data increases (≈ 5–10%) and remains competitive thereafter, matching the baseline closely in the 50–100% regime across most datasets.

The inset comparisons further show that KG/LocKG signals provide small but dataset-dependent gains at high label fractions (e.g., combined KG+LocKG benefits Hatespeech and MLDoc), while FuDoBa achieves similar performance with substantially reduced dimensionality, supporting it as an efficient and robust approach for multimodal embedding fusion under limited supervision.

5.3 Impact of Classifier Selection

To test whether FuDoBa is tied to a specific downstream learner, we rerun the *entire FuDoBa optimisation* separately for each learner: for a fixed search space and identical BO budget (50 steps), we use each learner as the evaluation function that scores candidate fusion configurations (projection sizes and modality importance weights). This setting is

Table 4 Impact of downstream evaluator (classifier) on FuDoBa representations (%)

Dataset	AutoGluon	RealMLP	TabICL	TabPFN	XGB
Macro-F1					
Books	93.77 ± 0.50	93.10 ± 0.60	*94.20 ± 0.20*	**94.42 ± 0.20**	92.98 ± 0.40
Dvd	93.53 ± 0.30	92.45 ± 1.00	*93.65 ± 0.40*	**93.68 ± 0.00**	92.80 ± 0.30
Hatespeech	*75.88 ± 0.40*	74.63 ± 0.30	**76.03 ± 0.30**	74.99 ± 0.80	72.56 ± 1.10
MLDoc	96.51 ± 0.30	96.72 ± 0.10	**97.55 ± 0.20**	*97.13 ± 0.00*	96.26 ± 0.10
Music	92.98 ± 0.30	92.67 ± 0.30	**93.45 ± 0.20**	*93.30 ± 0.10*	92.32 ± 0.20
Xgenre	**50.04 ± 0.20**	45.94 ± 0.50	*48.77 ± 0.80*	46.74 ± 0.50	48.21 ± 1.60
Avg score	83.78	82.58	**83.94**	83.38	82.52
Avg rank	2.7	4.2	**1.5**	2.2	4.5
Dataset	AutoGluon	RealMLP	TabICL	TabPFN	XGB
Recall					
Books	93.77 ± 0.50	93.10 ± 0.60	*94.20 ± 0.20*	**94.42 ± 0.20**	92.98 ± 0.40
Dvd	93.53 ± 0.30	92.45 ± 1.00	*93.65 ± 0.40*	**93.68 ± 0.00**	92.80 ± 0.30
Hatespeech	**74.25 ± 0.80**	73.70 ± 0.40	*73.79 ± 0.30*	72.52 ± 0.90	70.31 ± 1.00
MLDoc	96.50 ± 0.30	96.72 ± 0.10	**97.55 ± 0.20**	*97.13 ± 0.00*	96.25 ± 0.10
Music	92.98 ± 0.30	92.67 ± 0.30	**93.45 ± 0.20**	*93.30 ± 0.20*	92.32 ± 0.20
Xgenre	**64.62 ± 0.20**	57.97 ± 1.40	*62.68 ± 0.90*	61.70 ± 0.60	58.57 ± 2.40
Avg Score	**85.94**	84.43	85.89	85.46	83.87
Avg Rank	2.5	4.0	**1.7**	2.2	4.7
Dataset	AutoGluon	RealMLP	TabICL	TabPFN	XGB
Precision					
Books	93.77 ± 0.50	93.12 ± 0.60	*94.20 ± 0.20*	**94.42 ± 0.20**	93.01 ± 0.40
Dvd	93.55 ± 0.30	92.47 ± 1.00	*93.65 ± 0.40*	**93.69 ± 0.00**	92.80 ± 0.30
Hatespeech	78.82 ± 1.20	75.94 ± 0.50	80.73 ± 0.30	**81.15 ± 0.60**	78.62 ± 1.10
MLDoc	96.52 ± 0.20	96.73 ± 0.10	97.56 ± 0.20	*97.14 ± 0.00*	96.28 ± 0.10
Music	92.99 ± 0.30	92.67 ± 0.30	93.45 ± 0.20	*93.30 ± 0.20*	92.32 ± 0.20
Xgenre	**56.46 ± 0.10**	49.62 ± 2.90	51.71 ± 4.10	54.81 ± 0.30	*55.60±1.40*
Avg score	85.35	83.43	85.22	**85.75**	84.77
Avg rank	2.8	4.3	2.0	**1.7**	4.2

Mean±std over runs, reported in percentage points

Best per dataset in bold, second-best italic

intentionally stronger than a post-hoc learner swap: different learners may prefer different modality mixtures and weightings, so the learned fusion can shift depending on the learner used during optimisation. Table 4 shows that, despite this coupling, FuDoBa remains robust across model families. TabICL yields the strongest overall Macro-F1 (83.94%) and the best average rank on Macro-F1 (1.5) and Recall (1.7), indicating that it both guides the search toward consistently good fusion configurations and exploits the resulting representations reliably across datasets. TabPFN induces a more precision aligned representations, achieving the highest average Precision (85.75%) and leading on high-signal datasets such as *Books* and *Dvd*, suggesting that when FuDoBa is optimised under TabPFN, the selected modality weighting tends to favor configurations that reduce false positives. AutoGluon remains highly competitive and attains the best average Recall (85.94%), and it dominates *Xgenre* across Macro-F1/Recall/Precision, which is consistent with the BO objective favoring modality mixtures that recover positives in the hardest setting. This suggests that the choice of learner actively shapes the optimisation landscape, driving FuDoBa to converge on modality weights that complement the learner's specific inductive biases while maintaining high overall performance

Figure 5 further showcases an efficiency-performance trade-off that is directly relevant in this setting because BO requires repeated learner evaluations. TabICL provides a favorable Pareto point, matching or slightly exceeding AutoGluon in average Macro-F1 (83.94 vs. 83.78%) while being substantially cheaper per BO step on average (roughly $\sim$50 vs. $\sim$300 s). XGB is the fastest option but trails in Macro-F1, while RealMLP is slower and less competitive on average. TabPFN is typically moderate in runtime but shows stronger dataset-dependent variability, which can make the optimisation cost less predictable.

Finally, Fig. 6 suggests broadly comparable train–test gaps across learners, indicating that the observed differences are primarily attributable to (i) the learner's inductive bias and (ii) how it shapes FuDoBa's selected fusion configuration, rather than systematic overfitting under a particular learner. Taken together, these results show that FuDoBa can be effectively optimised with diverse learners–including pre-trained tabular foundation models (TabICL/TabPFN)–and that the learner choice controls both *what* fusion is discovered and *how* it performs, yielding predictable shifts in precision-recall behavior.

5.4 Analysis of the Impact of Modality Importance and Dimensionality

Next, we examine how the performance of our low-dimensional, FuDoBa representations compares to the high-dimensional concatenated representation Fig. 7 illustrates the relationship between F1-score and final feature dimension (log_2) scale) across datasets. While regression suggests a general trend of higher dimensions correlating with higher F1-scores, *FuDoBa* consistently operates at very low dimensions (approx. 2^6 to 2^7, see Fig. 8 and Table 11). Despite this low dimensionality, *FuDoBa* frequently achieves a highly competitive performance, often exceeding the general dimensionality-performance trend and sometimes matching or surpassing higher-dimensional approaches ($> 2^{10}$) (as shown in Fig. 2)

Given our interpretable modality weighting schema defined by parameters Θ, we next analyse the selected modality importances α and representational dimensions l across the datasets.

Figure 8 (left) shows that the optimiser consistently assigns the largest weight to the LLM modality on most datasets (Books: 0.77, Dvd: 0.80, Hatespeech: 0.90, MLDoc: 0.77,

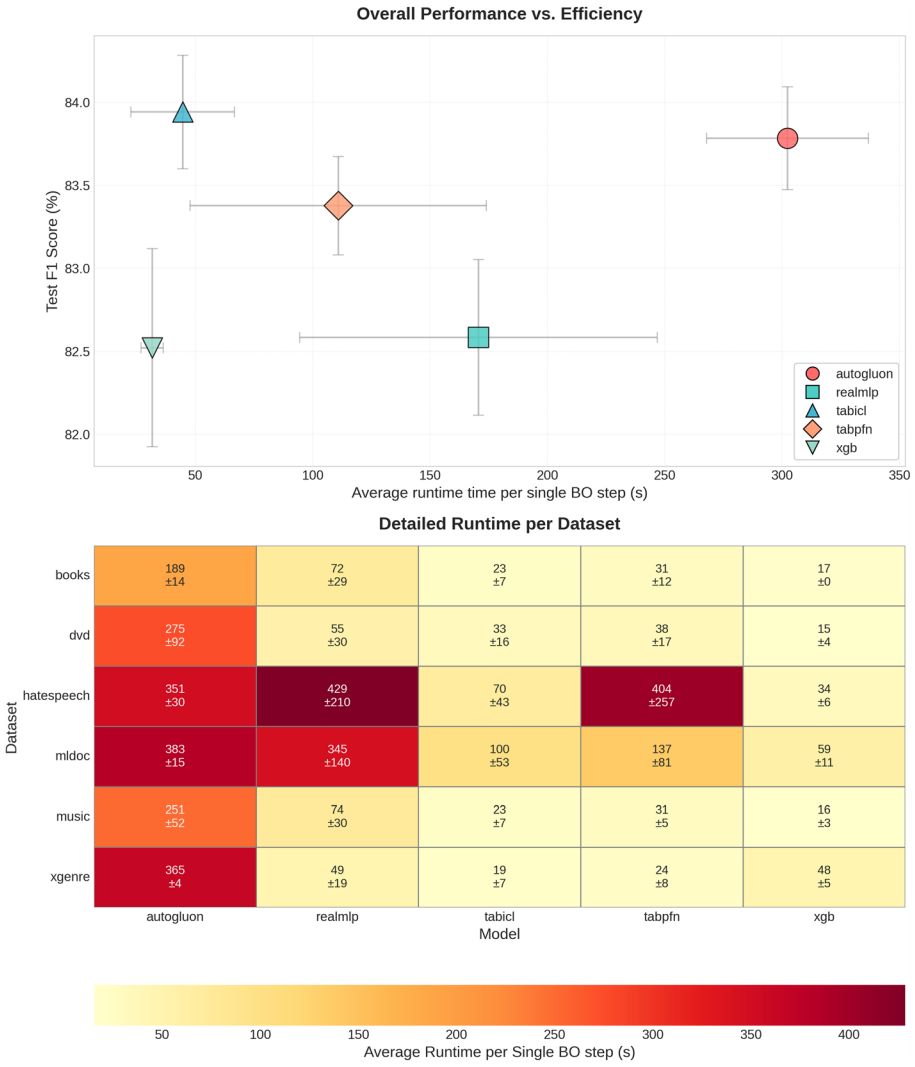

Fig. 5 Macro-F1 versus evaluation cost when running FuDoBa with different downstream classifier on fixed FuDoBa representations. Top: average Macro-F1 against average runtime per BO step, highlighting the efficiency–performance trade-off across learners. Bottom: per-dataset runtime (mean±std) for each learner, showing dataset-dependent variability

XGenre: 0.80), while Music exhibits near-uniform weighting across modalities ($\alpha_{\mathrm{KG}}=0.60$, $\alpha_{\mathrm{LocKG}}=0.57$, $\alpha_{\mathrm{LLM}}=0.60$). Notably, for XGenre the graph modalities are effectively discarded ($\alpha_{\mathrm{KG}}=\alpha_{\mathrm{LocKG}}=0$), indicating that KG-derived signals do not improve performance in this setting. In contrast, Dvd and MLDoc receive substantial non-zero KG and LocKG weights (roughly 0.37–0.47), suggesting complementary structured information beyond the LLM embeddings. Overall, the mean weights indicate a dominant but not exclusive role for LLM representations ($\overline{\alpha}_{\mathrm{LLM}} \approx 0.77$), with KG modalities contributing in a dataset-dependent manner ($\overline{\alpha}_{\mathrm{LocKG}} \approx 0.35$, $\overline{\alpha}_{\mathrm{KG}} \approx 0.31$). The variability indicated by the error

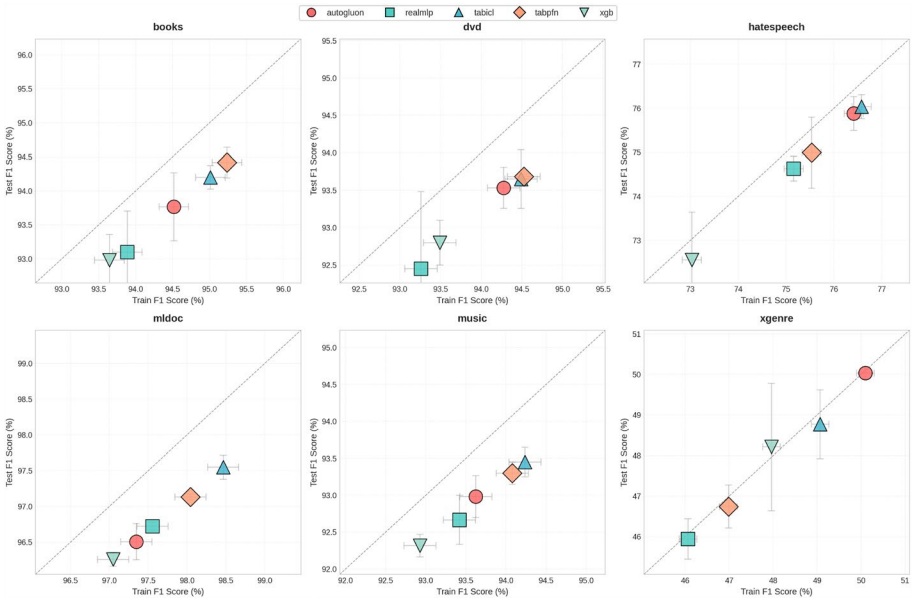

Fig. 6 Generalisation of different frameworks on the FuDoBa representations. Train versus test Macro-F1 (mean±std) for each dataset with the $y = x$ reference line; swapping the learner changes the operating point while maintaining similar generalisation gaps on the same fixed FuDoBa embedding

bars further suggests that multiple modality trade-offs can be near-optimal, reinforcing the value of adaptive fusion rather than fixed weighting.

Regarding representational dimensionality (Fig. 8, right), the optimiser typically retains a larger subspace for LLM embeddings ($l_{\mathrm{LLM}} = 48 - 64$), consistent with compressing a dense 1536-dimensional signal into a compact set of principal components. However, this is not uniform: on Dvd the LLM is aggressively compressed ($l_{\mathrm{LLM}} = 21$) while the KG subspaces remain comparatively large ($l_{\mathrm{KG}} = 48$, $l_{\mathrm{LocKG}} = 42$), implying that useful graph information may be distributed across more components even when its overall scaling is moderate. Across datasets, fused representations remain compact (approximately 101–133 dimensions by summing modality projections; and smaller in practice when α=0), yielding an order-of-magnitude reduction relative to the original LLM embedding size while preserving competitive performance. In our experiments, this corresponds to roughly 6–9% of the 1536-d LLM space (and about 4–5% of the full concatenated 2560-d space).

5.5 Analysis of the Mapped Knowledge

A comparative analysis of the global KG approach, inspired by the previous work (Koloski et al., 2024), and our local knowledge graph extraction method (LocKG), summarised in Table 5, reveals key differences in coverage. The KG approach consistently identifies more linked entries per document, particularly in longer datasets such as XGenre and MLDoc. In contrast, LocKG reduces the proportion of documents with no extracted information–for example, in the Hatespeech dataset, this percentage drops from 22% (KG) to 17% (LocKG).

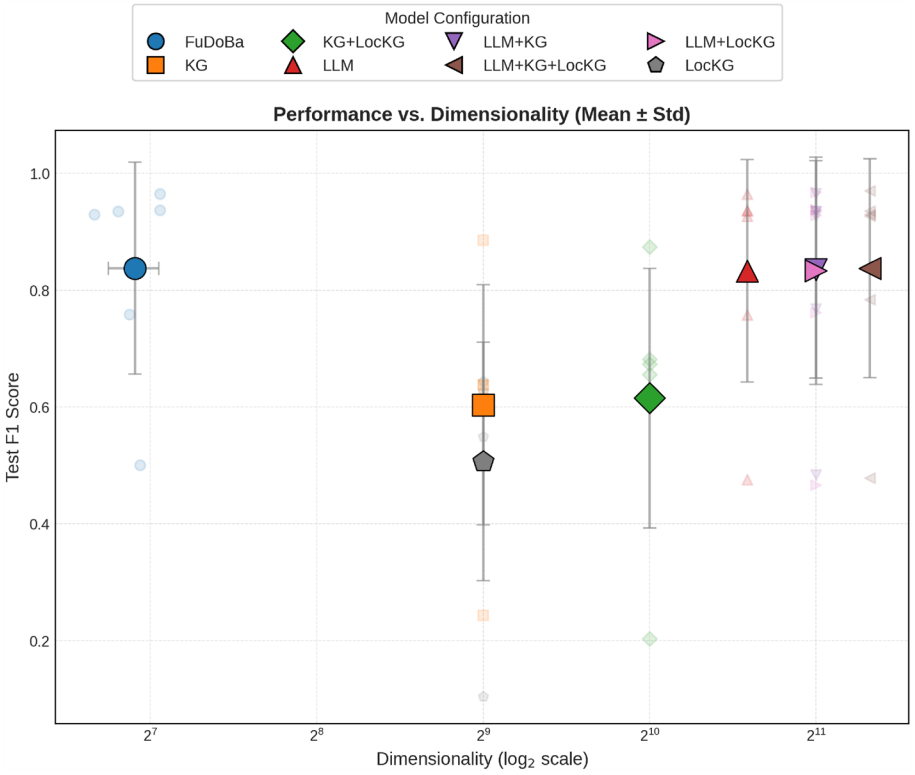

Fig. 7 The number of dimensions impacts performance. We see an interesting trend across datasets, with enriched representation methods performing on-par or better than models working on high dimensional fusion, and out-performing the LLM only baselines. FuDoBa performs the best if simultaneously considering F1-score and embedding dimension (computational complexity)

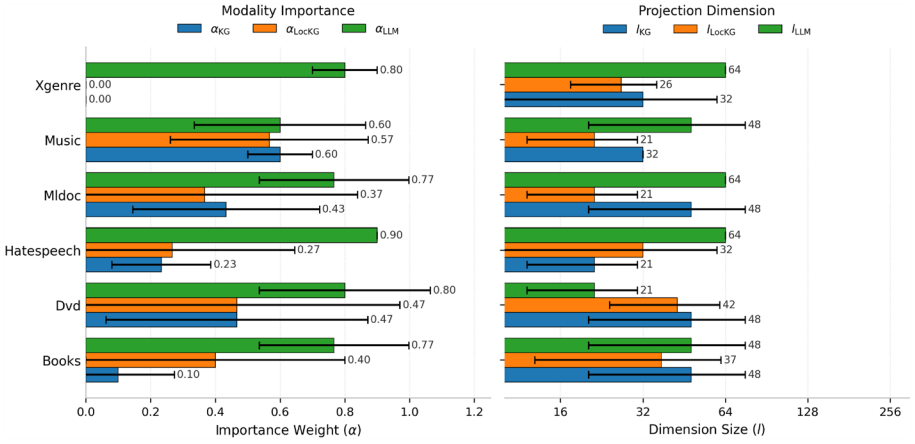

Fig. 8 Modality importance weight α and projection dimension l. It can be observed that LLM-based representation dominates—it is most commonly selected by the Bayesian optimisation as one of the key input components

Table 5 A comparison of the extracted entities per article between the Global and Local knowledge graphs reveals that the locally constructed KG provides better coverage–fewer articles lack mapped entities

Dataset	Docs w/o (%)		Mean ± Std		Max		Median	
	KG	LocKG	KG	LocKG	KG	LocKG	KG	LocKG
Music	2	0	18.03 ± 19.27	15.59 ± 7.89	189	57	12	14
Dvd	3	0	21.34 ± 25.08	17.49 ± 9.13	272	107	13	16
Books	3	0	19.00 ± 22.41	16.79 ± 8.40	203	78	12	15
Hatespeech	22	17	2.83 ± 2.63	4.99 ± 3.59	22	35	2	5
XGenre	0	0	61.90 ± 48.08	28.34 ± 11.65	281	116	49	27
MLDoc	0	0	48.04 ± 37.01	24.68 ± 11.15	440	126	37	23

Expectedly, the local KG maps a lower number of entities per document compared to the global graph. Docs w/o (%) denotes the proportion of documents without extracted entities

Additionally, LocKG yields an average of 4.99 entities per document in Hatespeech, compared to just 2.83 from the KG method.

These results highlight LocKG's robustness in extracting meaningful information from short, unstructured texts and support the premise that local knowledge graphs offer unique advantages in handling diverse, challenging data sources. From an embedding modeling perspective, we attribute part of this improvement to the aggregation mechanism used–specifically, the averaging approach shown in Fig. 1. While aggregating across a larger number of entities may introduce noise, the lower noise levels in LocKG suggest that the resulting representations are more specific. That said, global KGs still capture richer entity interactions and benefit from larger, more curated training corpora.

Next, we present illustrative examples of the mapped entities from both the Local and Global Knowledge Graphs in Table 6 for the Books, Music, DVD, and HateSpeech datasets. For longer-form collections such as MLDoc and XGenre, a tabular presentation of representative samples is less informative due to the richer and denser entity structure; we therefore additionally provide qualitative knowledge-graph visualisations. Specifically, Figs. 9 and 10 illustrate two complementary presentation styles: (i) a broader neighbourhood view highlighting diverse entity types and relations, and (ii) a more focused view emphasising salient relations around central entities.

6 Discussion

While LLM-based representations alone are powerful, our results (Sect. 5) demonstrate that enriching them with global and local knowledge improves average downstream performance (**RQ1**). Furthermore, we find that low-dimensional projection of multimodal inputs can yield competitive or superior scores while significantly reducing the memory footprint (**RQ3**). This suggests that for use cases prioritising both performance and storage efficiency, simultaneous optimisation of per-modality dimensionality and importance can offer a practical solution for conserving resources while improving downstream results. We note that 50 BO runs and the 3 projection dimensions (16, 32, 64), might represent a particularly restrictive budget and we expect that increasing the space of dimensions and the run count would generate more promising results. We note that our proposed fusing methodology is general and can work for any set of modalities $\mathcal{M}$. For example for image and text fusion, frozen, modality-specific encoders can be used to fuse the modalities.

Table 6 Illustrative samples with extracted global KG concepts and local KG triples (green: positive, pink: negative)

Domain	Text	KG signals
Books	100% recomend to learn about spy histor	**Global:** learn, spy **Local:** (spy histor, UsedFor, learning)
Books	Tried very hard to get through this book, but in the end gave up. Don't need a paragraph to describe a blade of grass. Put me to sleep	**Global:** trial, difficult, get_through, book, ... **Local:** (book, NotCapableOf, put me to sleep); (blade of grass, NotDesires, paragraph); (I, Desires, to get through this book); (I, NotDesires, to describe a blade of grass)
DVD	Ordered item arrived earlier than advertised and in perfect condition. Very satisfied and willing to do business with amazon again	**Global:** great, seller, videodisk, come_in, perfect, condition, gratitude, thank_you **Local:** (ordered item, IsA, product); (ordered item, HasProperty, perfect condition); (ordered item, CausesDesire, satisfaction); (satisfaction, MotivatedByGoal, willingness to do business with amazon again)
DVD	I don't think I'm going to be around in 2025 to give an accurate evaluation. If I'm still here I hope I still remember who neil young is, and I'm still rocking in the free world	**Global:** think, be_around, accurate, hope, remember, young, rock, free_world, ... **Local:** (i, Desires, accurate evaluation); (i, Desires, neil young); (i, CapableOf, remembering); (i, CapableOf, rocking); (neil young, IsA, musician); (free world, HasProperty, freedom)
Music	A potentially and stunned melodic death metal album. I'm not considder their best work but this album is their most complete. Incredible melodies, excellent musicianship, stunning guitar riffs. An awesome record by this guys!!	**Global:** potentially, amazed, death_metal, album, best_work, ... **Local:** (melodic death metal album, IsA, album); (this album, HasProperty, most complete); (this album, HasProperty, incredible melodies); (this album, HasProperty, excellent musicianship); (this album, HasProperty, stunning guitar riffs)
Music	The album is rushed, being produced just in 8 days. It did not showcase Casey's talent and sales charts speak even louder. Poor Casey. She deserves a better album than this	**Global:** album, produce, day, chart, deserve, better, ... **Local:** (album, HasProperty, rushed); (album, HasProperty, produced in 8 days); (album, UsedFor, showcase talent); (album, CreatedBy, casey); (casey, Desires, better album)
HateSpe	@USER CORRECTION: The Liberals won a false-majority on the backs of a lot of close races – emphasis on "false." #ElectoralReform	**Global:** liberal, win, close, race, emphasis, false **Local:** (liberals, HasProperty, false-majority); (false-majority, CausesDesire, electoral reform); (close races, PartOf, false-majority)
HateSpe	@USER @USER We need gun control! Lol	**Global:** necessitate, gun, gun_control, control **Local:** (gun control, Desires, user)

From a resource perspective, this approach presents notable advantages: the optimisation approach operates on low-dimensional representations and can be performed on commodity hardware, alleviating the need for costly LLM-based fine-tuning. We also hypothesise that per-modality projection induces a form of dataset-specific embedding alignment, which may enhance AutoML model generalisation. This aspect is particularly relevant in resource-constrained academic environments, where efficient optimisation procedures can yield performance comparable to, or exceeding, that of LLM-only representations.

Our findings further indicate that no single configuration is universally optimal, even within similar domains. Optimal projection dimensions l and modality importance weights α vary depending on the dataset and the fixed search budget (**RQ2**), highlighting the need for optimisation-based approaches to identify budget-constrained representations. In terms

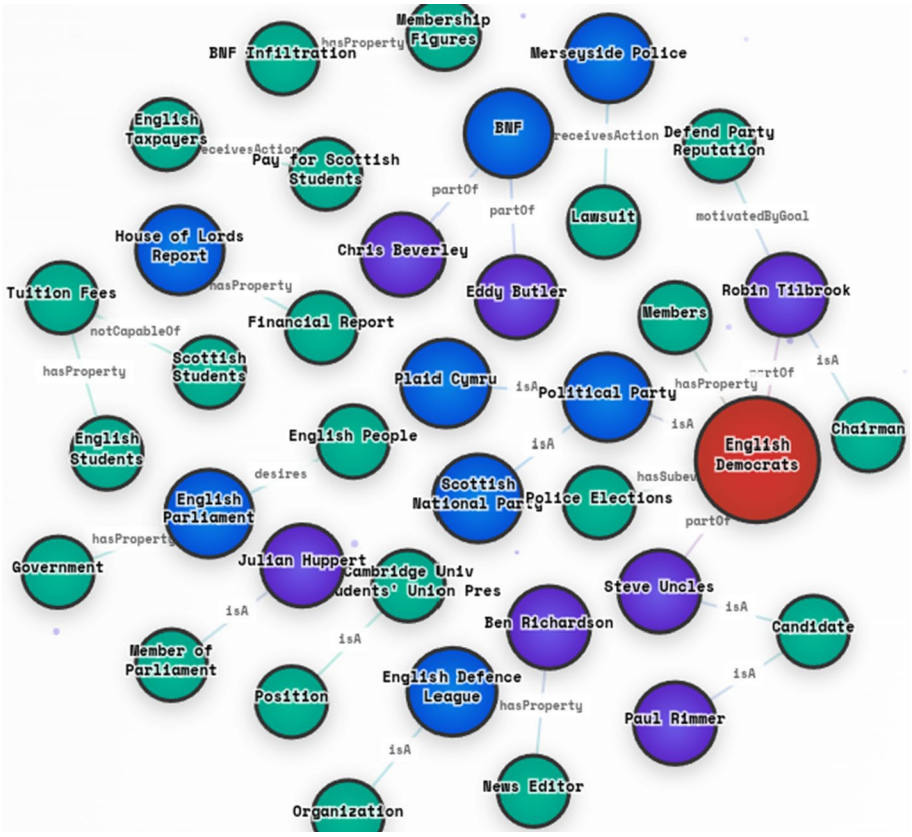

Fig. 9 Example local KG neighborhood extracted for MLDoc, illustrating a broader relational context around mapped entities

of cost-effectiveness, our method could be adapted to use pre-trained open-source relation extractors instead of LLM-based ones, with minimal impact on downstream performance for certain tasks (see Appendix), thereby further reducing computational demands. Our per-modality projection and weighting method (FuDoBa) also outperforms the concat-then-project approach (FuDoBa-CP). We hypothesise that AutoGluon's NNs may better adapt to the manifold of concatenated low-dimensional representations, capturing both initial heterogeneity and relative importance. In contrast, the concat-then-project variant produces an orthogonal, linear SVD space in which equally weighted modalities may be over-represented in the projected fused representation, with AutoGluon models potentially amplifying this imbalance during training (**RQ4**).

While modality weighting can also be approached using more sophisticated NNs (e.g. via routing) or genetic algorithms (GAs), these methods often require more data and compute resources such as GPU or memory. For example, GA-based searches–such as optimising modality weights with linear classifiers (Škrlj et al., 2021b)—typically involve training numerous candidates over several generations, which significantly increases memory usage. We suggest that, given sufficient computational resources, our optimisation framework

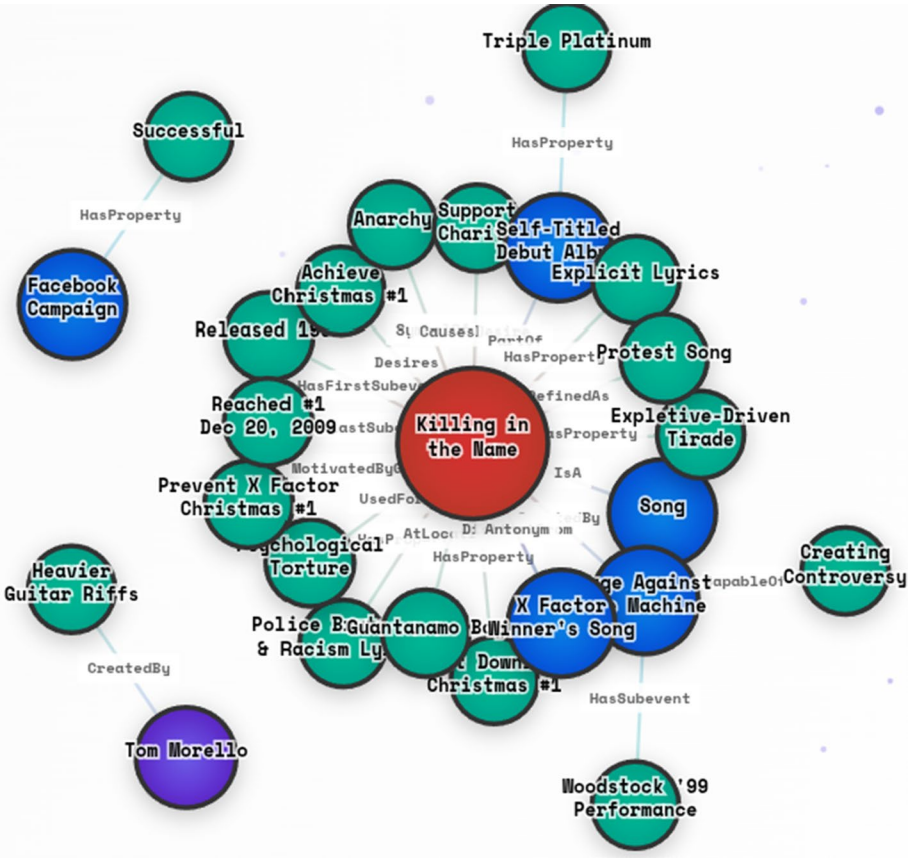

Fig. 10 Example local KG neighborhood for XGenre, illustrating a focused view of salient relations around a central entity

could potentially be adapted to parallel search strategies like GAs, though this remains to be validated.

FuDoBa's behaviour in few-shot regimes differs from the full-data setting (**RQ5**). In the ultra-low supervision regime (approximately 1–2% labeled data), the LLM-only baseline is often strongest, suggesting that aggressively compressing representations can initially reduce class separability when the downstream learner has too little signal to recover an effective decision boundary. However, FuDoBa closes the gap rapidly as supervision increases (roughly 5–10% labeled data) and remains competitive thereafter, matching or occasionally surpassing the LLM baseline in the mid-to-high supervision regime while using an order-of-magnitude fewer dimensions. We attribute this trend to the fact that, once a modest number of labels is available, FuDoBa's low-dimensional fusion acts as an implicit regulariser that reduces noise and redundancy, allowing tabular learners to exploit complementary KG and LocKG signals more effectively. Practically, these results indicate that LLM-only embeddings may be preferable for extremely label-scarce scenarios, whereas FuDoBa becomes an efficient and robust alternative once minimal supervision is available, offering comparable performance with substantially reduced computational and storage footprint.

We also investigated how recently published, pre-trained in-context tabular models perform as downstream learners for modality fusion in text classification within FuDoBa (**RQ6**). In our experiments, while AutoGluon often achieved higher training scores, the tabular foundation models generalised better at test time, with TabICL yielding the strongest Macro-F1 on average and TabPFN achieving the highest Precision. Interestingly, when run on similarly sized hardware, these models performed on par with, or better than, AutoGluon while requiring substantially less runtime per Bayesian optimisation evaluation, making them attractive as faster evaluators during fusion search. However, their practicality can be limited in settings with many classes, for example in TabPFN supports only a small maximum number of classes, and their inference-time compute requirements may be higher than those of lightweight deep models (resulting from AutoGluon's run), which can be prohibitive for some end users. Overall, these results suggest that in-context tabular models can serve as robust surrogates for faster, and potentially more stable, fusion prototyping under RQ6, provided their constraints are satisfied.

A question might arise whether it's better to train classifier models on LLM embeddings or instead use direct LLM prompting. The embedding-based approach offers significant benefits: generating embeddings is much faster and cheaper–often 10 to 100 times less costly (see Appendix A), where we show that using text-embedding-3-small for an average 8000-character document is up to 575 times less expensive than prompting a GPT-4 Turbo model. This also allows using less resource-intensive local models for the final classification step, as we show in this work. Furthermore, these embeddings can be reused for many different tasks later on, like unsupervised learning (i.e. clustering) or checking for shifts in the data overtime (i.e. diachronic analysis). This method can also reduce the risk of leaking sensitive data compared to prompting methods that need both input and output examples, which could reveal business secrets. However, there are drawbacks, notably the risk of data overlap where the data used to train the classifier might have already been seen by the LLM during its pre-training. On the other hand, direct LLM prompting is better at quickly adjusting to changes in the data, may needfewer examples to get started, and might perform better for certain tasks. There's also potential for synergy, where LLMs help create approximate *soft* labels to make training data for embedding models. Despite these points, the need for powerful hardware to run large LLMs for prompting is still a major practical challenge. This highlights the value of simpler, efficient local models, which often work well with the embedding approach.

Finally, we observe that different relation extractors produce different graphs (Tables 8 and 9), but achieve similar average scores (see Sect. 7). We believe this results from differences in underlying training data and architectures, which nonetheless capture domain-specific knowledge. When trained with link-completion methods such as rotatE (Sun et al., 2019), it seems that the resulting representations appear to normalize across extractors. While this remains a hypothesis, future analysis could explore how outputs from different extractors might be effectively combined.

7 Conclusions and Further Work

In this work, we introduce FuDoBa, a framework designed to overcome the challenges of high dimensionality and costly adaptation in Large Language Model (LLM) embeddings. FuDoBa employs Bayesian optimisation to fuse core LLM representations with structured knowledge extracted from both global and local Knowledge Graphs. By adaptively learning optimal low-dimensional projections and interpretable importance weights for each modality, the framework balances information preservation with compactness, resulting in enhanced, task-specific representations.

Our experiments show that FuDoBa achieves classification performance comparable to or better than high-dimensional baselines (including LLM-only and simple concatenation approaches) while significantly reducing feature dimensionality (often using only a fraction of the original space). This demonstrates FuDoBa as a practical, compute-efficient alternative to resource-intensive LLM fine-tuning, particularly when domain adaptation is required and computing resources are limited. Moreover, our results indicate that in high-dimensional spaces, our approach can outperform LLM-only models, highlighting the benefit of incorporating external knowledge for improved performance.

Despite these promising results, FuDoBa has several limitations. Its performance relative to simpler baselines varies across datasets; for instance, in the Hatespeech dataset, a high-dimensional concatenation method outperformed FuDoBa, indicating potential sensitivity to factors such as text noise and domain variability. Additionally, the quality of the fused knowledge–particularly from the Local Knowledge Graph–depends heavily on the accuracy of the underlying relation extraction models, meaning that extraction errors can degrade the final representation. Our AutoML setup also relies on relatively short optimisation runs (approximately 5 min), which may limit the effectiveness of the search. Finally, although FuDoBa reduces downstream training time through dimensionality reduction, the Bayesian optimisation phase introduces additional computational overhead. While its sequential nature is well-suited for low-memory environments, it may be suboptimal on high-resource systems where parallel search strategies would be more efficient.

For future work, we plan to apply and analyse FuDoBa on multiple domains and novel datasets. We will also compare our embedding-based approach against direct LLM prompting to assess downstream performance differences. Because FuDoBa scales naturally to any number of modalities, we intend to investigate the integration of knowledge graphs with image data. Finally, having evaluated only English datasets to date, we will extend our experiments to additional languages.

Appendix A

Implementation Details

Software Details

We implemented our code using Python 3.11, organising our project with the UV package organiser. For KG embedding, we leveraged the PyKeen library configured with 512-dimensional embeddings, 1000 training epochs, and a batch size of 8192. For experimental and Bayesian hyperparameter search, we employed the Bayesian search functionality from Weights & Biases (wandb), limiting the search to 50 runs. Additionally, we used AutoGluon 1.2 with the `good_quality` preset and parallel fitting. The AutoGluon settings were as follows:

- `fit_strategy = 'parallel'`
- `num_bag_folds = 5`
- `num_bag_sets = 1`
- `time_limit = 300 s`

Hardware

We evaluated all of our experiments on a standard, commodity PC.

```
############### CPU ###############
Architecture:                    x86_64
  CPU op-mode(s):                32-bit, 64-bit
  Address sizes:                 39 bits physical, 48 bits virtual
  Byte Order:                    Little Endian
CPU(s):                          12
  On-line CPU(s) list:           0-11
Vendor ID:                       GenuineIntel
  Model name:                    Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz
    CPU family:                  6
    Model:                       158
    Thread(s) per core:          2
    Core(s) per socket:          6
    Socket(s):                   1
    Stepping:                    10
    CPU max MHz:                 4700,0000
    CPU min MHz:                 800,0000
    BogoMIPS:                    7399.70
Caches (sum of all):
  L1d:                           192 KiB (6 instances)
  L1i:                           192 KiB (6 instances)
  L2:                            1,5 MiB (6 instances)
  L3:                            12 MiB (1 instance)
NUMA:
  NUMA node(s):                  1
  NUMA node0 CPU(s):             0-11

############### Memory ###############
               total        used        free      shared  buff/cache   available
Mem:            62Gi        15Gi        27Gi       1,6Gi        19Gi        45Gi
Swap:          1,9Gi       1,0Gi       940Mi
```

Listing 1 System description

Prompt for Relation Extraction

In our work, we used the following prompt for knowledge extraction and graph construction:

```
You are an expert in knowledge extraction and graph construction. Your task is
     to extract and represent information from the provided text as a
     collection of knowledge graph triplets in a JSON array. Each element in
     the array should be an object with keys "entity1", "relation", and
     "entity2", using the following allowed relations only:
IsA, PartOf, UsedFor, CapableOf, HasProperty, AtLocation, Causes,
     CausesDesire, Desires, MadeOf, HasSubevent, HasFirstSubevent,
     HasLastSubevent, NotCapableOf, NotDesires, NotHasProperty, Antonym,
     DefinedAs, DerivedFrom, DistinctFrom, Entails, ReceivesAction,
     MotivatedByGoal, CreatedBy, SymbolOf, EtymologicallyRelatedTo, FormOf,
     InstanceOf.

Please follow these guidelines:
1. Standardized and Unique Entities:
   - Extract only clear, general concepts exactly as they appear in the text.
   - Normalize entities by using lower-case and singular forms when applicable
       to avoid near duplicates.
2. Concise Entities:
   - Ensure each entity represents a single, clear concept; avoid combining
       multiple concepts into one entity.
3. Robust Mapping:
    - Do not derive or reinterpret entities-use the exact wording from the
        text so that each entity can be directly traced back.
4. Simplicity in Relationships:
   - Use the allowed relations to denote simple and clear interactions between
       entities.

Document:
"""{document}"""
```

Listing 2 Prompt text for knowledge extraction and graph construction

Appendix B

On the Impact of the Relation Extractor Models

We next assess the impact of the Relation Extractor (RE) model on downstream task performance (Table 7). Within our LLM+KG+LocKG framework, we compare the performance when using the *gpt4o-mini* as an extractor versus two smaller, open models (ReBeL (Huguet Cabot & Navigli, 2021), ReLiK (Barba et al., 2024)) as the RE component.

Across the six datasets, average F1-scores were highly similar for all three RE choices. Although gpt4o-mini yielded a slightly higher mean score, the overall difference between the models was marginal, spanning approximately 1.25 percentage points. A One-Way ANOVA confirmed that these minor variations in mean F1-scores are not statistically significant ($p = 0.9931$).

Therefore, the choice among gpt4o-mini, ReBeL, and ReLiK as the Relation Extractor did not significantly influence downstream performance in this experimental setting,

Table 7 Comparison of average macro F1 test scores (%) across datasets for different relationextractors

Dataset relation extractor	Books	Dvd	Hatespeech	MLDoc	Music	Xgenre	Average Perf. (%)
gpt4o-mini	92.95	93.20	78.40	97.34	93.15	50.95	84.33 ± 17.60
Rebel	93.10	93.70	78.45	95.99	93.15	44.08	83.08 ± 20.12
ReLiK	**93.35**	**93.75**	76.94	96.14	93.00	50.02	83.87 ± 17.97

suggesting that smaller, open-access models like ReBeL (Huguet Cabot & Navigli, 2021) and ReLiK (Barba et al., 2024) serve as viable and effective alternatives to the larger, pay-per-use gpt4o-mini for this specific component within our framework. This supports the flexibility of our methodology, allowing for comparable performance with potentially lower costs and greater accessibility.

Qualitative Analysis of the Extracted Relations

Next, we conduct a qualitative examination of the extracted relations between different methods. Table 8 shows significant differences in the models' extraction capabilities, both in quantity (with gpt4o-mini producing 3–8 times more triplets than its counterparts) and in the level of detail and abstractness–with gpt4o-mini generating more concise, high-level constructs like *book*, *government*, or *movie*. On the extracted triplets side, gpt-4o-mini consistently employs the generic "HasProperty" relation across all datasets, indicating a flexible approach to relationship categorisation. In contrast, Rebel and Relik tend toward more specific, domain-relevant relations such as "country," "performer," or "author," suggesting more structured extraction guidelines.

The qualitative differences between the models shown in Table 9 are even more pronounced, with each model extracting distinctly different semantic information from the same text. For example, in the MLDoc dataset, Rebel extracts the factual political affiliation "(Alain Lamassoure, member of political party, UDF)," while Relik focuses on organisational structures "(France, executive body, European Commission)," and gpt-4o-mini captures capability assertions "(France, CapableOf, more tax cuts)." This, combined with the

Table 8 Extracted graph statistics per model and dataset

Method	Unique Ent.	Unique Rel.	Total triples	Unique triples	Most Freq. entity	Most Freq. relation
Hatespeech						
Rebel	8972	245	23,534	15,507	@USER	Different from
Relik	2239	232	7341	5573	@USER	Country
gpt-4o-mini	14,695	478	48,582	45,413	User	HasProperty
MLDoc						
Rebel	13,903	188	31,139	21,315	United States	Country
Relik	40,482	319	178,950	105,731	Its	Country
gpt-4o-mini	90,676	4269	251,844	244,296	Government	HasProperty
Music						
Rebel	5433	115	7279	6200	The beatles	Performer
Relik	6454	151	16,511	13,146	This album	Performer
gpt-4o-mini	17,469	480	45,894	43,643	Album	HasProperty
Dvd						
Rebel	4767	151	7470	6102	Dvd	Cast member
Relik	5986	212	20,606	15,702	It	Cast member
gpt-4o-mini	17,994	614	50,721	48,062	Movie	HasProperty
Books						
Rebel	4916	152	7046	5649	Harvard University	Author
Relik	5020	218	15,376	11,453	This book	Author
gpt-4o-mini	17,208	843	47,746	45,414	Book	HasProperty

Table 9 Extracted graph examples per model on same example per dataset

Model	Example triple
Hatespeech	
Rebel	(@USER, part of, BB)
Relik	(Ford, member of, conservatives)
gpt-4o-mini	(she, NotDesires, leaving)
Xgenre	
Rebel	(Anything Brilliant, creator, Jenn Co)
Relik	(her, religion or worldview, church)
gpt-4o-mini	(jenn co, instanceOf, fashion stylist)
MLDoc	
Rebel	(Alain Lamassoure, member of political party, UDF)
Relik	(France, executive body, Europoean Commission)
gpt-4o-mini	(france, CapableOf, more tax cuts)
Music	
Rebel	(1956, point in time, 1956)
Relik	(this, distribution format, LP)
gpt-4o-mini	(mr. farlow, DistinctFrom, wes montgomery)
Dvd	
Rebel	(Amazon, product or material produced, product)
Relik	(the Dvd, distribution format, Dvd)
gpt-4o-mini	(ordered item, IsA, product)
Books	
Rebel	(spy, studied by, histor)
Relik	(American, ethnic group, American Indians)
gpt-4o-mini	(spy histor, UsedFor, learning)

results from Table 7, shows that despite the differences between the models–each with its own semantic focus–each captures a specific local view. Future work might consider how these specific domain views, unique to each model, could be incorporated to yield a more robust local representation. The reason for these differences likely stems from the distinct architectural and algorithmic designs of each model, with gpt4o-mini being a generalist foundational LLM, while both Rebel and Relik are specialist PLMs.

Appendix C

Complementary Results

Comparison ofLLM-Embeddings to LLM Querying Prices

We next compare the costs associated with two approaches for LLM-based tasks: direct prompting versus generating embeddings for downstream use. This analysis contrasts the estimated cost of using various proprietary (closed-source) and hosted open-source LLMs for a representative prompting task against the baseline cost of using OpenAI's text-embed-

ding-3-small model for embedding the same input data. The base price per 1 M tokens for the embedding model is 0.02$.

The comparison is based on the following assumptions: the prompting task involves 2000 input and 100 output tokens, while the baseline embedding task uses 2000 input tokens with text-embedding-3-small (estimated cost ≈ $0.00004). All prices are based on publicly available data from service providers as of April 15, 2025.

Our findings indicate that the prompting approach is substantially more expensive than using the text-embedding-3-small baseline for embeddings. The cost ratio ranges from approximately 2.3 times higher (for Gemini 1.5 Flash) up to 575 times higher (for GPT-4 Turbo), as detailed in the subsequent analysis (Table 10)

We do a meta-analysis of the importances of the θ parameters in our Bayesian search. For each dataset in Figs. 11, 12, 13, 14, 15, 16, we show the Bayesian evolution in the leftmost plot. Using the shaded area, we mark the best score found ($f*$ from Algorithm 1) so far, coupled with the feature importance–captured by fitting a Random Forest to the parameters and the achieved score–and correlation analysis in the rightmost plot. We see that across datasets, different parameters are important (e.g. for the Books dataset, the α_{kg} parameter seems the most important, possibly explaining why it resulted in the limit-value of 0). While for the larger news genre dataset, we see that l_{LLM} is most important. We also see that the best-value hit at different points for different datasets, with most datasets hitting it in the latter half, implying that a longer search might have resulted in better scores, probably improving the downstream performance.

Table 10 Cost ratio of LLM prompting versus `text-embedding-3-small` embedding (≈ 2000 token input around 8000 characters)

Model	Input price ($/1 M tk)	Output price ($/1 M tk)	Est. prompt cost ($)	Baseline Emb. cost ($)	Ratio (prompt/emb) (approx. × times)	Ref
Baseline embedding model						
`text-embedding-3-small`	0.02	N/A	N/A	≈ 0.00004	1×	1
Gemini prompting models						
`Gemini 1.5 flash`	0.0375	0.15	≈ 0.00009	≈ 0.00004	≈ 2.3×	2
`Gemini 1.5 pro`	1.25	5.00	≈ 0.00300	≈ 0.00004	≈ 75×	2
GPT (OpenAI) prompting models						
`gpt-4o-mini`	0.15	0.60	≈ 0.00036	≈ 0.00004	≈ 9×	1
`gpt-3.5-turbo-0125`	0.50	1.50	≈ 0.00115	≈ 0.00004	≈ 29×	1
`gpt-4o`	5.00	15.00	≈ 0.01150	≈ 0.00004	≈ 288×	1
`gpt-4-turbo`	10.00	30.00	≈ 0.02300	≈ 0.00004	≈ 575×	1

1 Pricing source: OpenAI API Pricing—https://openai.com/api/pricing/

2 Pricing source: Google AI Gemini API Pricing—https://ai.google.dev/gemini-api/docs/pricing*Feature importance between searches*

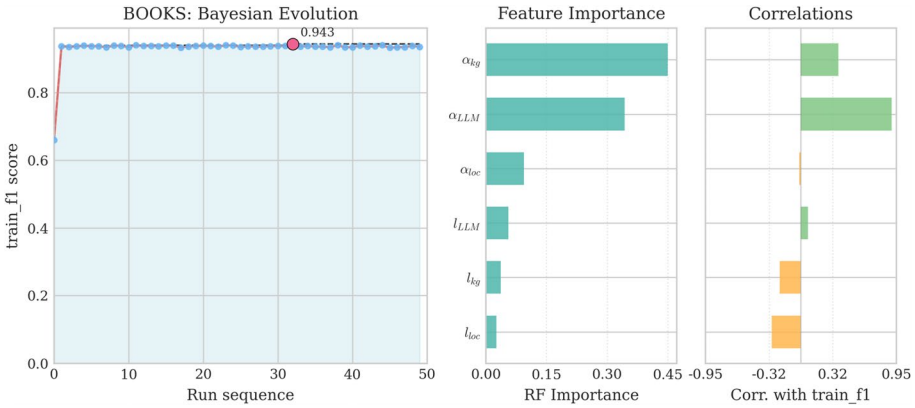

Fig. 11 Feature importances of Bayesian evolution for dataset books

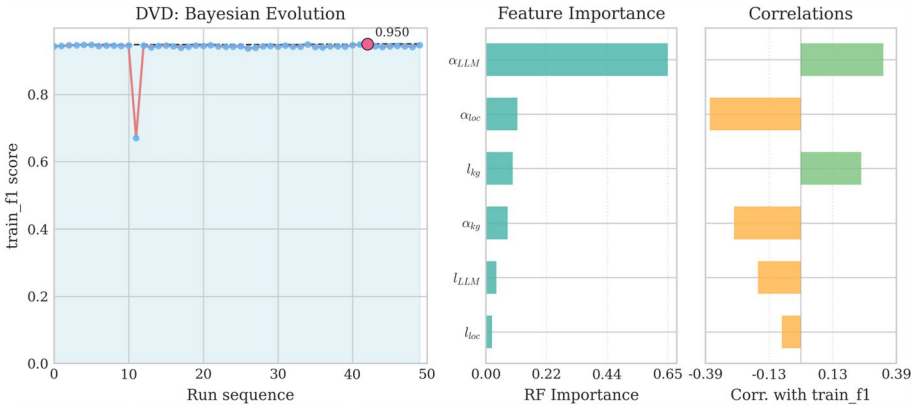

Fig. 12 Feature importances of Bayesian evolution for dataset Dvd

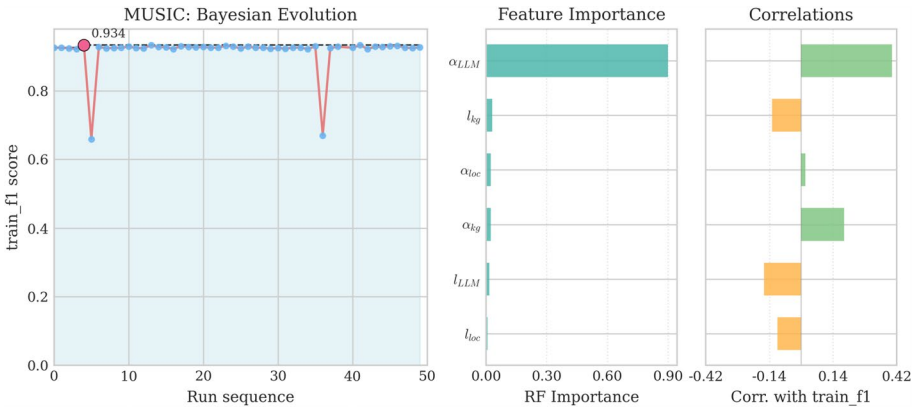

Fig. 13 Feature importances of Bayesian evolution for dataset music

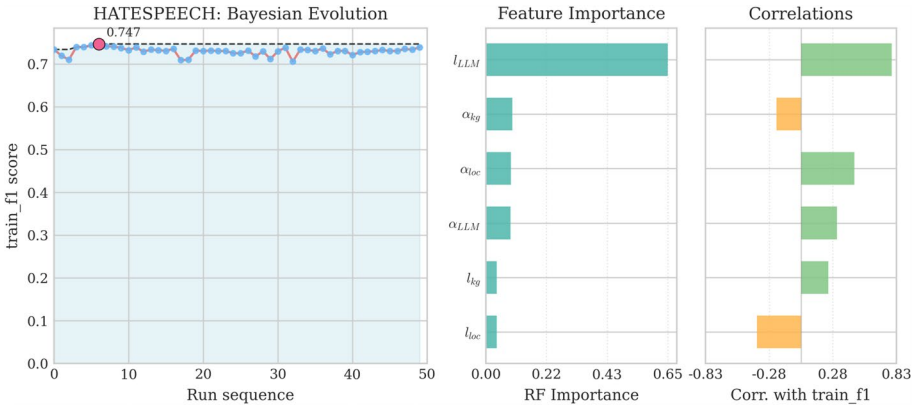

Fig. 14 Feature importances of Bayesian evolution for dataset hate-speech

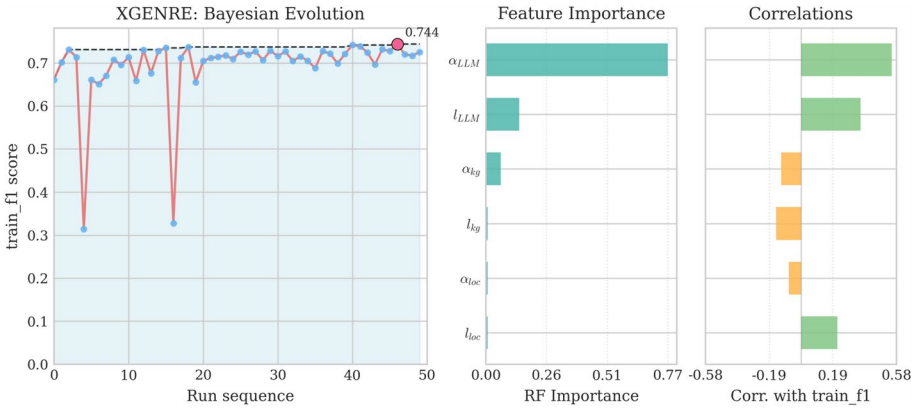

Fig. 15 Feature importances of Bayesian evolution for dataset Xgenre

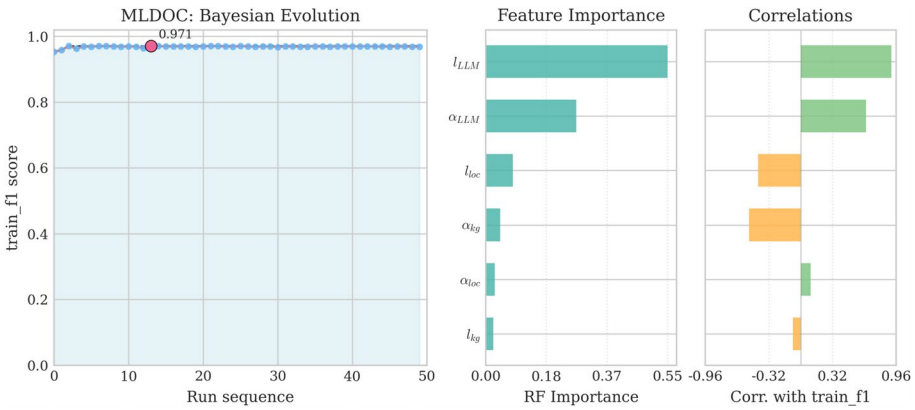

Fig. 16 Feature importances of Bayesian evolution for dataset MLdoc

Dimension to Score Results

Here, we present the relationship between the final projection dimension and F1-score across different datasets for each approach. Consistent with the averaged results, our approach (blue circle, FuDoBa) achieves scores comparable to–or better than–the LLM-only representation (red diamond). In general, higher-dimensional representations tend to yield better F1-scores, although the R-scores exhibit negligible trends (Fig. 17).

Detailed Search Results

See Table 11.

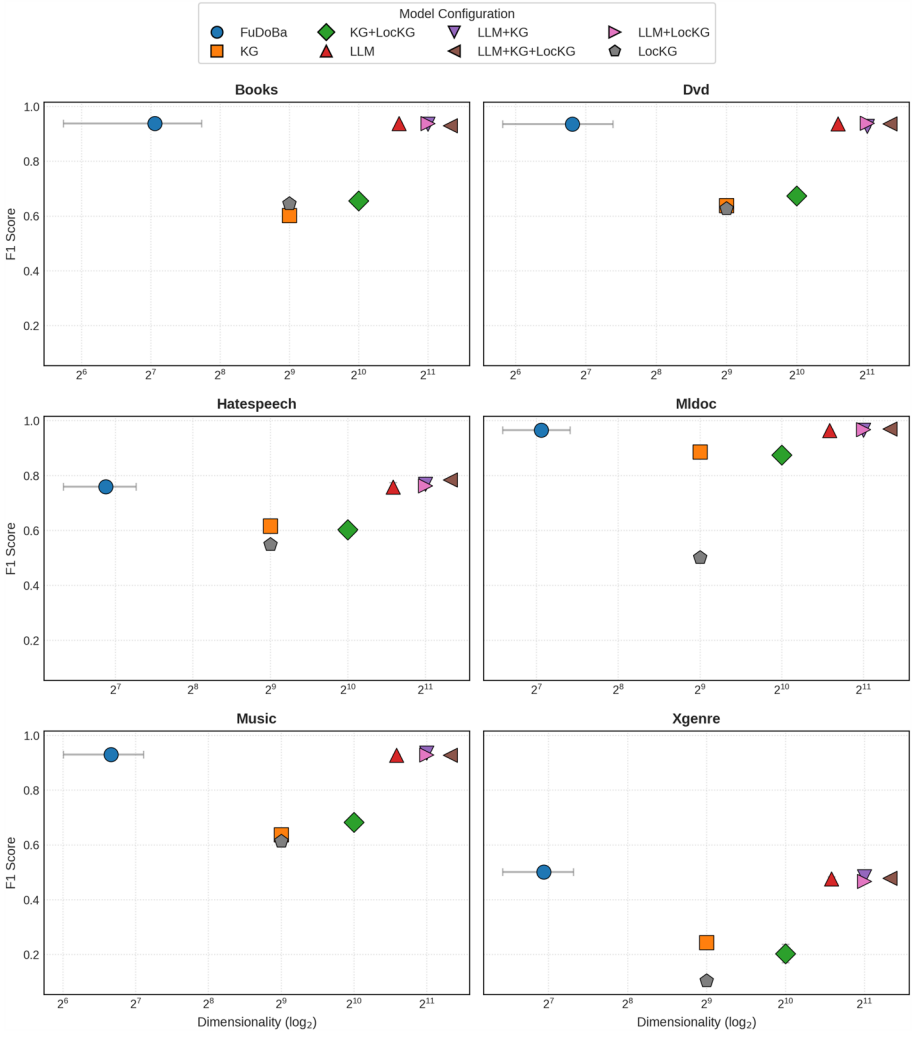

Fig. 17 Per-dataset analysis on the impact of dimensionality (x-axis, log2-scaled) and the achieved scores (y-axis, macro F1-score)

Table 11 Results per dataset with full dimensionality

Method	Mult Text	Mult KG	Mult Loc	Proj Text	Proj KG	Proj Loc	Full Dim	F1	Recall	Precision
Books										
LLM	✓	×	×	×	×	×	1536	$0.937_{\pm 0.003}$	$0.938_{\pm}0.003$	$0.938_{\pm}0.003$
LLM + KG	✓	✓	×	×	×	×	2048	$0.935_{\pm 0.005}$	$0.935_{\pm 0.005}$	$0.935_{\pm 0.005}$
LLM + LocKG	✓	×	✓	×	×	×	2048	$\mathbf{0.938_{\pm}0.005}$	$\mathbf{0.938_{\pm}0.005}$	$\mathbf{0.938_{\pm}0.005}$
LLM + KG + LocKG	✓	✓	✓	×	×	×	2560	$0.930_{\pm 0.002}$	$0.930_{\pm 0.002}$	$0.930_{\pm 0.002}$
FuDoBa	$0.767_{\pm 0.231}$	$0.100_{\pm 0.173}$	$0.400_{\pm 0.400}$	$48.000_{\pm 27.713}$	$48.000_{\pm 27.713}$	$37.333_{\pm 24.440}$	133.3	$\mathbf{0.938_{\pm}0.005}$	$\mathbf{0.938_{\pm}0.005}$	$\mathbf{0.938_{\pm}0.005}$
Dvd										
LLM	✓	×	×	×	×	×	1536	$0.936_{\pm 0.004}$	$0.936_{\pm 0.004}$	$0.936_{\pm 0.003}$
LLM + KG	✓	✓	×	×	×	×	2048	$0.927_{\pm 0.006}$	$0.927_{\pm 0.006}$	$0.928_{\pm 0.006}$
LLM + LocKG	✓	×	✓	×	×	×	2048	$\mathbf{0.938_{\pm}0.001}$	$\mathbf{0.938_{\pm}0.001}$	$\mathbf{0.939_{\pm}0.001}$
LLM + KG + LocKG	✓	✓	✓	×	×	×	2560	$0.936_{\pm 0.005}$	$0.936_{\pm 0.005}$	$0.937_{\pm 0.004}$
FuDoBa	$0.800_{\pm 0.265}$	$0.467_{\pm 0.404}$	$0.467_{\pm 0.503}$	$21.333_{\pm 9.238}$	$48.000_{\pm 27.713}$	$42.667_{\pm 18.475}$	112.0	$0.935_{\pm 0.003}$	$0.935_{\pm 0.003}$	$0.935_{\pm 0.003}$
Hatespeech										
LLM	✓	×	×	×	×	×	1536	$0.758_{\pm 0.016}$	$0.743_{\pm 0.021}$	$0.786_{\pm 0.012}$
LLM + KG	✓	✓	×	×	×	×	2048	$0.768_{\pm 0.004}$	$0.762_{\pm 0.006}$	$0.776_{\pm 0.003}$
LLM + LocKG	✓	×	✓	×	×	×	2048	$0.762_{\pm 0.002}$	$0.757_{\pm 0.002}$	$0.768_{\pm 0.008}$
LLM + KG + LocKG	✓	✓	✓	×	×	×	2560	$\mathbf{0.784_{\pm}0.003}$	$\mathbf{0.773_{\pm}0.001}$	$\mathbf{0.801_{\pm}0.010}$
FuDoBa	$0.900_{\pm 0.000}$	$0.233_{\pm 0.153}$	$0.267_{\pm 0.379}$	$64.000_{\pm 0.000}$	$21.333_{\pm 9.238}$	$32.000_{\pm 27.713}$	117.3	$0.759_{\pm 0.004}$	$0.742_{\pm 0.008}$	$0.788_{\pm 0.012}$
MLDoc										
LLM	✓	×	×	×	×	×	1536	$0.965_{\pm 0.001}$	$0.964_{\pm 0.001}$	$0.965_{\pm 0.001}$
LLM + KG	✓	✓	×	×	×	×	2048	$0.965_{\pm 0.003}$	$0.965_{\pm 0.003}$	$0.966_{\pm 0.003}$
LLM + LocKG	✓	×	✓	×	×	×	2048	$0.968_{\pm 0.003}$	$0.968_{\pm 0.003}$	$0.968_{\pm 0.003}$

Table 11 (continued)

Method	Mult Text	Mult KG	Mult Loc	Proj Text	Proj KG	Proj Loc	Full Dim	F1	Recall	Precision
LLM + KG + LocKG	✓	✓	✓	×	×	×	2560	$\mathbf{0.970_{\pm 0.002}}$	$\mathbf{0.970_{\pm 0.002}}$	$\mathbf{0.970_{\pm 0.002}}$
FuDoBa	$0.767_{\pm 0.231}$	$0.433_{\pm 0.289}$	$0.367_{\pm 0.473}$	$64.000_{\pm 0.000}$	$48.000_{\pm 27.713}$	$21.333_{\pm 9.238}$	133.3	$0.965_{\pm 0.003}$	$0.965_{\pm 0.003}$	$0.965_{\pm 0.002}$
Music										
LLM	✓	×	×	×	×	×	1536	$0.927_{\pm 0.002}$	$0.927_{\pm 0.002}$	$0.928_{\pm 0.002}$
LLM + KG	✓	✓	×	×	×	×	2048	$\mathbf{0.935_{\pm 0.001}}$	$\mathbf{0.935_{\pm 0.001}}$	$\mathbf{0.935_{\pm 0.001}}$
LLM + LocKG	✓	×	✓	×	×	×	2048	$0.928_{\pm 0.003}$	$0.928_{\pm 0.003}$	$0.929_{\pm 0.003}$
LLM + KG + LocKG	✓	✓	✓	×	×	×	2560	$0.927_{\pm 0.006}$	$0.927_{\pm 0.006}$	$0.928_{\pm 0.006}$
FuDoBa	$0.600_{\pm 0.265}$	$0.600_{\pm 0.100}$	$0.567_{\pm 0.306}$	$48.000_{\pm 27.713}$	$32.000_{\pm 0.000}$	$21.333_{\pm 9.238}$	101.3	$0.930_{\pm 0.003}$	$0.930_{\pm 0.003}$	$0.930_{\pm 0.003}$
XGenre										
LLM	✓	×	×	×	×	×	1536	$0.476_{\pm 0.007}$	$0.606_{\pm 0.004}$	$0.470_{\pm 0.006}$
LLM + KG	✓	✓	×	×	×	×	2048	$0.484_{\pm 0.013}$	$0.587_{\pm 0.007}$	$0.528_{\pm 0.032}$
LLM + LocKG	✓	×	✓	×	×	×	2048	$0.466_{\pm 0.003}$	$0.580_{\pm 0.005}$	$0.493_{\pm 0.037}$
LLM + KG + LocKG	✓	✓	✓	×	×	×	2560	$0.478_{\pm 0.005}$	$0.608_{\pm 0.007}$	$0.511_{\pm 0.030}$
FuDoBa	$0.800_{\pm 0.100}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$64.000_{\pm 0.000}$	$32.000_{\pm 27.713}$	$26.667_{\pm 9.238}$	122.7	$\mathbf{0.500_{\pm 0.002}}$	$\mathbf{0.646_{\pm 0.002}}$	$\mathbf{0.565_{\pm 0.001}}$

For baselines, ✓/× indicate whether a modality is included. For FuDoBa, the "Mult" columns report the average modality weights α and the "Proj" columns report the selected projection dimensions l

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s10994-026-07008-y.

Acknowledgements We acknowledge the financial support of the Slovenian Research and Innovation Agency (ARIS) through grants GC-0001 (Artificial Intelligence for Science), GC-0002 (Large Language Models for Digital Humanities), L2-50070 (Embeddings-based Techniques for Media Monitoring Applications), J5-3102 (Hatespeech in contemporary conceptualizations of nationalism, racism, gender and migration) and the core research programme P2-0103 (Knowledge Technologies). The work of B.K. was supported by the Young Researcher Grant PR-12394. The work of R.N. was funded from CREATIVE project (CRossmodal understanding and gEnerATIon of Visual and tExtual content) funded by the MUR Progetti di Ricerca di Rilevante Interesse Nazionale programme (PRIN 2020). We thank Jaya Caporusso for her feedback on an earlier draft of this manuscript.

Author Contributions B.K. wrote the main manuscript text, prepared the experimental data, set up the experimental environment, executed the experiments, and collected evidence, as well as generated all figures and tables. S.P. and R.N. reviewed the draft and provided guidance on the experimental design. B.S. led the experimental design setup, screened the results, and helped revise the manuscript. All authors reviewed and approved the final version.

Data Availability No datasets were generated or analysed during the current study.

Code Availability The complete code, including the pre-calculated embeddings, intermediate results, and scripts to reproduce the plots, will be made available upon acceptance on the following GitHub repository: https://github.com/bkolosk1/fudoba.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

Aggarwal, C.C., Hinneburg, A., Keim, D.A. (2001). On the surprising behavior of distance metrics in high dimensional spaces. In: *Proceedings of the 8th international conference on database theory*. (pp. 420–434). Springer.

Barba, E., Orlando, R., Cabot, P.-L.H., & Navigli, R. (2024). ReLiK: Retrieve, read and link: Fast and accurate entity linking and relation extraction on an academic budget. https://openreview.net/forum?id=b0IRscfEOb

BehnamGhader, P., Adlakha, V., Mosbach, M., Bahdanau, D., Chapados, N., & Reddy, S. (2024). LLM2Vec: Large language models are secretly powerful text encoders. In: *First conference on language modeling*.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 785–794). ACM.

Chen, Q., Wang, W., Huang, K., & Coenen, F. (2022). Zero-shot text classification via knowledge graph embedding for social media data. *IEEE Internet of Things Journal, 9*(12), 9205–9213. https://doi.org/10.1109/JIOT.2021.3093065

Cocchi, F., Moratelli, N., Cornia, M., Baraldi, L., & Cucchiara, R. (2025). Augmenting multimodal LLMs with self-reflective tokens for knowledge-based visual question answering.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research, 7*, 1–30.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Long and Short PapersIn J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies* (Vol. 1, pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., & Smola, A. (2020). Autogluon-tabular: Robust and accurate automl for structured data. Preprint retrieved from https://arxiv.org/abs/2003.06505

Fan, A., Gardent, C., Braud, C., & Bordes, A. (2019). Using local knowledge graphconstruction to scale Seq2Seq models to multi-document inputs. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 4186–4196). Association for Computational Linguistics.

Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple contrastive learning of sentence embeddings. In M. F. Moens, X. Huang, L. Specia, & S. W. T. Yih (Eds.), *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 6894–6910). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.552

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A. & Yang, A. (2024). The Llama 3 Herd of models

Hollmann, N., Müller, S., Eggensperger, K., & Hutter, F. (2025). Accurate predictions on small data with a tabular foundation model. *Nature, 637*, 319–326.

Holzmüller, D., Grinsztajn, L., & Steinwart, I. (2024). Better by default: Strong pre-tuned MLPs and boosted trees on tabular data. In: The thirty-eighth annual conference on neural information processing systems. https://openreview.net/forum?id=3BNPUDvqMt

Huguet Cabot, P.-L., & Navigli, R. (2021). REBEL: Relation extraction by end-to-end language generation. In M.-F. Moens, X. Huang, L. Specia, & S.W.-T. Yih (Eds.), *Findings of the association for computational linguistics: EMNLP 2021* (pp. 2370–2381). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-emnlp.204

Jiao, T., Guo, C., Feng, X., Chen, Y., & Song, J. (2024). A comprehensive survey on deep learning multimodal fusion: Methods, technologies and applications. *Computers, Materials and Continua, 80*(1), 1–35. https://doi.org/10.32604/cmc.2024.053204

Khosla, S., Tiwari, A., Kafle, K., Jenni, S., Zhao, H., Collomosse, J., & Shi, J. (2025). MAGNET: Augmenting generative decoders with representation learning and infilling capabilities. Preprint retrieved from https://arxiv.org/abs/2501.08648

Klema, V., & Laub, A. (1980). The singular value decomposition: Its computation and some applications. *IEEE Transactions on Automatic Control, 25*(2), 164–176. https://doi.org/10.1109/TAC.1980.1102314

Koduri, S. (2012). Multisensor data fusion with singular value decomposition. In: *2012 UKSim 14th international conference on computer modelling and simulation*, (pp. 422–426). https://doi.org/10.1109/UKSim.2012.65

Koloski, B., Stepišnik Perdih, T., Robnik-Šikonja, M., Pollak, S., & Škrlj, B. (2022). Knowledge graph informed fake news classification via heterogeneous representation ensembles. *Neurocomputing, 496*, 208–226. https://doi.org/10.1016/j.neucom.2022.01.096

Koloski, B., Pollak, S., Navigli, R., & Škrlj, B. (2025). Automl-guided fusion of entity and llm-based representations fordocument classification. *Discovery science: 27th international conference, DS 2024, Pisa, Italy, October 14–16, 2024, proceedings, part I* (pp. 101–115). Springer. https://doi.org/10.1007/978-3-031-78977-9_7

Kuzman, T., Rupnik, P., & Ljubešić, N. (2022). The GINCO training dataset for web genre identification of documents out in the wild. In N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, & S. Piperidis (Eds.), *Proceedings of the thirteenth language resources and evaluation conference* (pp. 1584–1594). European Language Resources Association.

Lahat, D., Adali, T., & Jutten, C. (2015). Multimodal data fusion: An overview of methods, challenges, and prospects. *Proceedings of the IEEE, 103*(9), 1449–1477.

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In Xing, E.P., Jebara, T. (Eds.), *Proceedings of the 31st international conference on machine learning. Proceedings of machine learning research*, (vol. 32, pp. 1188–1196). PMLR. https://proceedings.mlr.press/v32/le14.html

Le, T. T., Fu, W., & Moore, J. H. (2020). Scaling tree-based automated machine learning to biomedical big data with a feature set selector. *Bioinformatics, 36*(1), 250–256.

Li, X., & Li, J. (2024). AoE: Angle-optimized embeddings for semantic textual similarity. In: Ku, L.-W., Martins, A., & Srikumar, V. (Eds.), *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: long papers)*, (pp. 1825–1839). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.acl-long.101

Moro, A., Raganato, A., & Navigli, R. (2014). Entity linking meets word sense disambiguation: A unified approach. *Transactions of the Association for Computational Linguistics, 2*, 231–244. https://doi.org/10.1162/tacl_a_00179

Muennighoff, N., Tazi, N., Magne, L., & Reimers, N. (2023). MTEB: Massive text embedding benchmark. In A. Vlachos & I. Augenstein (Eds.), *Proceedings of the 17th conference of the European chapter of the association for computational linguistics* (pp. 2014–2037). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.eacl-main.148

Navigli, R., & Ponzetto, S.P. (2010). BabelNet: Building a very large multilingual semantic network. In: Hajič, J., Carberry, S., Clark, S., & Nivre, J. (Eds.), *Proceedings of the 48th annual meeting of the association for computational linguistics*, (pp. 216–225). Association for Computational Linguistics. https://aclanthology.org/P10-1023

Ostendorff, M., Bourgonje, P., Berger, M., Moreno-Schneider, J., Rehm, G., & Gipp, B. (2019). Enriching bert with knowledge graph embeddings for document classification. Preprint retrieved from https://arxiv.org/abs/1909.08402

Qu, J., Holzmüller, D., Varoquaux, G., & Morvan, M.L. (2025). TabICL: A tabular foundation model for in-context learning on large data. In: Forty-second international conference on machine learning. https://openreview.net/forum?id=0VvD1PmNzM

Ranasinghe, T., & Zampieri, M. (2020). Multilingual offensive language identification with cross-lingual embeddings. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)* (pp. 5838–5844). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.470

Rasmussen, C. E., & Williams, C. K. I. (2005). *Gaussian processes for machine learning*. The MIT Press. https://doi.org/10.7551/mitpress/3206.001.0001

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 3982–3992). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1410

Sarmah, B., Mehta, D., Hall, B., Rao, R., Patel, S., & Pasquali, S. (2024). Hybridrag: Integrating knowledge graphs and vector retrieval augmented generation for efficient information extraction. In: *Proceedings of the 5th ACM international conference on AI in finance*, (pp. 608–616). Association for Computing Machinery. https://doi.org/10.1145/3677052.3698671

Schwenk, H., & Li, X. (2018). A corpus for multilingual document classification in eight languages. In N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, & T. Tokunaga (Eds.), *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*. European Language Resources Association (ELRA).

Škrlj, B., & Petković, M. (2021). Compressibility of distributed document representations. In: *2021 IEEE international conference on data mining (ICDM)*, (pp. 1330–1335). https://doi.org/10.1109/ICDM51629.2021.00166

Škrlj, B., Martinc, M., Kralj, J., Lavrač, N., & Pollak, S. (2021a). tax2vec: Constructing interpretable features from taxonomies for short text classification. *Computer Speech & Language, 65*, Article 101104. https://doi.org/10.1016/j.csl.2020.101104

Škrlj, B., Martinc, M., Lavrač, N., & Pollak, S. (2021b). Autobot: Evolving neuro-symbolic representations for explainable low resource text classification. *Machine Learning*. https://doi.org/10.1007/s10994-021-05968-x

Snoek, J., Larochelle, H., & Adams, R.P. (2012) Practical bayesian optimization of machine learning algorithms. In: Bartlett, P.L., Pereira, F.C.N., Burges, C.J.C., Bottou, L., & Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a Meeting Held December 3–6, 2012, Lake Tahoe, Nevada, United States, pp. 2960–2968

Speer, R., Chin, J., & Havasi, C. (2017) Conceptnet 5.5: An open multilingual graph of general knowledge. In: *Proceedings of the thirty-first AAAI conference on artificial intelligence*, (pp. 4444–4451). AAAI Press.

Sun, Z., Deng, Z.-H., Nie, J.-Y., & Tang, J. (2019). Rotate: Knowledge graph embedding by relational rotation incomplex space. In: *International conference on learning representations*, (pp. 1–18). https://openreview.net/forum?id=HkgEQnRqYQ

Wang, X., Gao, T., Zhu, Z., Zhang, Z., Liu, Z., Li, J., & Tang, J. (2021). KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics, 9*, 176–194. https://doi.org/10.1162/tacl_a_00360

Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., & Wei, F. (2024). Improving text embeddings with large language models. In L.-W. Ku, A. Martins, & V. Srikumar (Eds.), *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: long papers)* (pp. 11897–11916). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.acl-long.642