IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 June 2025, accepted 19 July 2025, date of publication 5 August 2025, date of current version 15 August 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3595894

RESEARCH ARTICLE

Benchmarking Sentence Encoders in Associating Indicators With Sustainable Development Goals and Targets

ANA GJORGJEVIKJ[1,2], KOSTADIN MISHEV[1], DIMITAR TRAJANOV[1,3], (Member, IEEE), AND LJUPCO KOCAREV[1,4], (Fellow, IEEE)

[1]Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, 1000 Skopje, North Macedonia
[2]Computer Systems Department, Jožef Stefan Institute, 1000 Ljubljana, Slovenia
[3]Department of Computer Science, Metropolitan College, Boston University, Boston, MA 02215, USA
[4]Macedonian Academy of Sciences and Arts, 1000 Skopje, North Macedonia

Corresponding author: Ana Gjorgjevikj (ana.gjorgjevikj@ijs.si)

This work was supported in part by Slovenian Research Agency under Program Grant P2-0098 and Project Grant GC-0001, and in part by European Union under Grant 101211695 (Horizon Europe MSCA-PF AutoLLMSelect).

ABSTRACT The United Nations' 2030 Agenda for Sustainable Development balances the economic, environmental, and social dimension of sustainable development in 17 Sustainable Development Goals (SDGs), monitored through a well-defined set of targets and global indicators. Although essential for humanity's future well-being, this monitoring is still challenging due to the variable quality of the statistical data of global indicators compiled at the national level and the diversity of indicators used to monitor sustainable development at the subnational level. Associating indicators other than the global ones with the SDGs/targets may help not only to expand the statistical data, but to better align the efforts toward sustainable development taken at (sub)national level. This article presents a model-agnostic framework for associating such indicators with the SDGs and targets by comparing their textual descriptions in a common representation space. While removing the dependence on the quantity and quality of the statistical data of the indicators, it provides human experts with data-driven suggestions on the complex and not always obvious associations between the indicators and the SDGs/targets. A comprehensive domain-specific benchmarking of a diverse sentence encoder portfolio was performed first, followed by fine-tuning of the best ones on a newly created dataset. Five sets of indicators used at the (sub)national level of governance (around 800 indicators in total) were used for the evaluation. Finally, the influence of 40 factors on the results was analyzed using explainable artificial intelligence (xAI) methods. The results show that 1) certain sentence encoders are better suited to solving the task than others (potentially due to their diverse pre-training datasets), 2) the fine-tuning not only improves the predictive performance over the baselines but also reduces the sensitivity to changes in indicator description length (performance drops even by up to 17% for baseline models as length increases, but remains comparable for fine-tuned models), and 3) better selected training instances have the potential to improve the performance even further (taking into account the limited fine-tuning dataset currently used and the insights from the xAI analysis). Most importantly, this article contributes to filling the existing gap in comprehensive benchmarking of AI models in solving the problem.

INDEX TERMS Machine learning, natural language processing, representation learning, sustainable development.

I. INTRODUCTION

The 2030 Agenda for Sustainable Development [1] of the United Nations (UN), adopted in September 2015,

The associate editor coordinating the review of this manuscript and approving it for publication was Loris Belcastro.

represents a plan to take action in the most crucial areas for the well-being of the planet and humanity. The Agenda consists of 17 Sustainable Development Goals (SDGs) and 169 targets, which describe what needs to be achieved by 2030 to ensure a sustainable future. For example, SDG 1 is devoted to ending poverty in all its forms everywhere, while its target 1.1 specifically requires eradicating extreme poverty, measured as people living on less than $1.25 a day, by 2030 [1] (for details on the SDGs, see the Appendix A). Progress is monitored through the Global indicator framework for the SDGs and targets of the Agenda [2], adopted in July 2017, refined annually, and including 231 unique indicators at the time of writing.[1] The Agenda defines the SDGs and their targets as integrated and indivisible, balancing the three dimensions of sustainable development, i.e., the economic, social, and environmental dimension [1]. Therefore, trying to achieve the SDGs and targets in isolation can lead to unintended outcomes [3]. The interactions between the SDGs may be positive when coordinated actions lead to beneficial outcomes at a lower cost or with a higher impact, or negative when actions lead to trade-offs [4]. Consequently, achieving the Agenda as a whole requires knowledge of the SDG dependencies, the strength of the dependencies, the direction of influence, the reversibility of the effects and the certainty of the perceived outcomes in the regional context [3]. However, the interactions between the SDGs are not defined in the Agenda itself and may depend on the context (e.g., geographic region, time, level of governance). As most of the actions supporting the Agenda take place at local, regional, and national levels [5], achieving the Agenda requires accurate monitoring of the effects that the policies/actions taken at those levels of governance have on progress. Although the role of regional and local governments was recognized in the Agenda itself [1], the latest SDG Reports pointed to their central role in achieving the Agenda since 65% of the targets are actually linked to their work [6].

However, there are several challenges related to SDG monitoring through indicators from the Global framework or other locally relevant indicators reported to date. For example, the statistical data of the Global indicator framework, collected and compiled at the national level, still has gaps with respect to its timelines, geographic coverage, and disaggregation by required dimensions as a result of the uneven statistical capacity of the countries [6], [7]. For example, the SDG Report 2023 [6] highlights that more than 50% of the latest available data come from 2020 and 2021, while the lack of internationally comparable data is particularly noticeable for SDGs 5, 13, and 16, for which more than half of the 193 countries lack such data. This makes progress monitoring challenging and was especially emphasized during the COVID-19 pandemic when even well-established methods for data collection (e.g., in-person) became unavailable [7]. The need for innovative data collection methods, non-standard data sources, and data integration from multiple sources is evident, but only through careful design and evaluation [6], [7]. Furthermore, the use of the Global indicator framework in measuring progress toward the SDGs is not always a straightforward task at the regional or local levels of governance [7], [8], [9], [10], [11]. At those levels, the selection of an appropriate indicator framework for monitoring sustainability may be burdened by competing objectives of the process, e.g., the need for context-specific indicators that are better suited to local needs vs. indicators from international frameworks that allow comparability on a global level [9]. Therefore, it is not uncommon to use locally relevant indicator sets to monitor sustainability at those levels. For example, a research article [9] identified 67 initiatives developing indicator sets for urban areas. On the other hand, not all available indicator sets for monitoring urban sustainability are aligned with the 2030 Agenda [8]. Harmonization and homogeneity of the data are pointed as key challenges when analyzing the SDGs at the regional level in the European Union (EU), as such data can be scarce or come from multiple sources [11]. In such circumstances, many initiatives aim at ''localizing'' the SDGs and assisting in their integration into local policies, as further described in Section II-A. Localization is the process of defining/implementing/monitoring strategies for achieving SDGs at the local level, i.e., translating the Agenda into local results [12].

From our literature review, several key challenges that motivated this article were identified:

- Variety of indicator frameworks used at different levels of governance and geographic regions.
- Necessity to properly associate locally relevant indicator frameworks with the SDGs/targets in order to properly monitor the effects that local policies have on the 2030 Agenda.
- Necessity to find even the less obvious associations between locally relevant indicators and SDGs/targets, which is a nontrivial task due to context-dependent interactions between the SDGs/targets.
- Necessity to complement/facilitate the nontrivial and time-consuming manual mapping process done by human experts through thoroughly evaluated data-driven methods, capable of inferring such less obvious associations in a transparent manner.
- Variable statistical data quality and quantity in certain geographic regions, even for the indicators from the Global framework, which requires consideration of alternative types of data available in large volumes but underutilized for the purpose. An example is textual data available in the form of SDG-related scientific publications, Voluntary National/Local Reviews, progress reports, news articles, and similar (see Section II-B).
- Lack of comprehensive benchmarks of the strengths and weaknesses of the various artificial intelligence (AI) models for processing text, e.g., (large) language

[1] https://unstats.un.org/sdgs/indicators/indicators-list

models or sentence encoders, in associating indicators to SDGs/targets (see Section II-B).

Our contributions to tackle the challenges mentioned above consist of the following:

- Developing a text-driven model-agnostic framework Embed4SD, to find associations between indicators and the (1) 17 SDGs and (2) 169 targets.
- Comprehensive benchmarking of the potential of a diverse portfolio of publicly available pre-trained general-purpose sentence encoders in solving the problem.
- Evaluating the fine-tuned sentence encoders on two "main" tasks, i.e., multi-class classification of an indicator to one of the (1) 17 SDGs and (2) 169 targets, as well as two "auxiliary" (zero-shot classification) tasks, unseen during fine-tuning and validation.
- Evaluating the fine-tuned sentence encoders using five indicator sets used at national, regional, and local levels of governance (with around 800 indicators in total), which differ in their purpose and characteristics.
- Creating a new domain-specific dataset to enable sentence encoder fine-tuning.
- Post-hoc analyses using methods from explainable artificial intelligence (xAI) to better understand the factors influencing the results and gain insight for future improvements.
- Complementing and potentially facilitating the non-trivial and time-consuming manual indicator mapping process done by human experts by providing them with data-driven suggestions on indicators associations to the SDGs/targets.
- Public availability of the framework to allow for reproducibility, critical assessment, and improvement.

Through the proposed framework and experiments, the following research questions were addressed:

1) How can textual data and general-purpose pre-trained sentence encoders be used to automate the process of associating national, regional, and local indicators with the SDGs and targets from the UN 2030 Agenda?
2) What improvement does domain-specific fine-tuning of general-purpose pre-trained sentence encoders bring to their performance in solving the main and auxiliary tasks?
3) What kind of textual data should be used to describe the SDGs, targets, and indicators when using the proposed framework?

The rest of the article is organized as follows. It starts with a brief overview of different initiatives that facilitate SDG monitoring at national and subnational levels of governance, as well as the related scientific literature studying the text classification to SDGs/targets. It is followed by two sections describing the dataset creation process, the pre-trained sentence encoders benchmarking, fine-tuning, validation, and testing, as well as the post-hoc analysis of the factors influencing the test results, both in terms of the methodology itself and in terms of the experimental choices. Finally, the validation and test results are presented, along with a discussion of the research questions. Embed4SD implementation is available on GitHub [13].

II. RELATED WORK

This section briefly describes initiatives aimed at facilitating SDG monitoring at the national or subnational level, mainly by aligning different indicator sets with the SDGs, targets, or global indicators from the 2030 Agenda. The interactions between the SDGs, targets, indicators, and policies have already been studied by the research community, and several secondary studies [5], [14] have summarized the primary from different aspects. In the second subsection, we focus on those using textual data for those purposes.

A. FACILITATING THE SDG MONITORING AT (SUB)NATIONAL LEVEL

The 2030 Agenda encourages UN member states to conduct reviews of their progress in implementing the Agenda on national and subnational levels on a regular basis [1] and share experiences through Voluntary National Reviews (VNRs) of the High-level Political Forum for Sustainable Development. In addition to the indicators from the Global framework, in their VNRs, the member states can report the use of additional national or subnational indicators to measure the progress in achieving the SDGs and are encouraged to include an annex with data [15]. Voluntary Local Reviews (VLRs) are subnational reviews on progress in achieving the SDGs produced by regional or local governments. According to the Guidelines for VLRs from 2020 [16], the review should provide information on the indicator sets used, i.e., if those are already available indicator sets or newly developed ones. In the later case, details on the methodology should be provided.

The European Union (EU) SDG indicator set was adopted in 2017 and consists of 100 indicators, of which 33 monitor multiple SDGs, and 68 are aligned with indicators from the Global framework [17]. It allows monitoring of the progress in achieving the SDGs in the context of EU policies, and its development was led by the statistical office of the EU – EUROSTAT in cooperation with other relevant institutions [17]. The project URBAN2030, supported by the European Commission Directorate General for Urban and Regional Policies and realized by the Joint Research Centre, aimed to offer EU cities a framework for developing VLRs and to help achieving the SDGs at the local or regional level [18]. Output of the project was the first edition of the European Handbook for SDG Voluntary Local Reviews in 2020 [19]. The project URBAN2030-II resulted in a second edition of the Handbook in 2022 [10]. The 72 example indicators in the second edition, coming from international institutions, European institutions, research institutes, and regional governments, help regional and local governments in monitoring progress towards the SDGs and 54 targets [10].

The project REGIONS2030, supported by the European Parliament and realized by the Joint Research Centre, aimed to identify indicators relevant for monitoring the SDG at the regional level [11], [20]. It started with a set of 83 indicators [20], tested in several pilot regions, and resulted in a final set of 116 indicators [11]. As part of the United for Smart Sustainable Cities (U4SSC) UN initiative, coordinated by the International Telecommunication Union (ITU), United Nations Economic Commission for Europe (UNECE), and United Nations Human Settlements Programme (UN-Habitat), a set of key performance indicators[2] was developed to allow cities to measure their progress in becoming smart and sustainable through ICT, as well as measure their progress in achieving the SDGs. The UN Sustainable Development Solutions Network (SDSN)[3] aims to mobilize various institutions worldwide to take actions to achieve the SDGs. Among the many initiatives, some aim to improve the monitoring of the SDGs at the urban or regional level by aligning local indicators with the SDGs.

B. METHODS FOR TEXT CLASSIFICATION TO SDGs OR TARGETS

The latest advances in machine learning (ML) have huge potential to help solve sustainable development problems. However, some of the obstacles to their application are related to the required domain-specific knowledge which ML practitioners usually lack, as well as the unavailability of standardized benchmarks for the problems [21]. Furthermore, the use of AI (in general) in achieving the SDGs requires awareness of the SDG interactions and sufficient oversight since it can have both positive and negative impacts [22], [23]. When it comes to the use of natural language processing (NLP) advances based on deep learning (DL) in solving SDG-related problems, our literature review showed that it started attracting attention only in the last few years. The use of textual data to associate external indicators with the SDGs, targets, or global indicators from the 2030 Agenda (as done in this article) is uncommon, but there is a growing interest in ML-based classification of text to SDGs in general. This subsection briefly summarizes such research articles.

Soriano et al. [24] presented an approach to classify short target and indicator descriptions to SDGs using several language models based on BERT. The dataset consisted of 400 sentences that described global targets and indicators, and were labeled with SDGs. Two types of experiments were conducted, the first encoding the descriptions in a common vector space and classifying them based on their k nearest neighbors, while the second fine-tuning the models for multi-class classification to SDGs. ChatGPT was evaluated as well. The accuracy of the fine-tuned classifiers did not exceed 0.7, and ChatGPT had an accuracy of 0.84($\pm$0.04). Matsui et al. [25] fine-tuned a BERT model (pre-trained on Japanese text) on 3,758 sentences related to the SDGs to perform multi-label classification of a sentence to the 17 SDGs. The authors then used the predictions for a set of indicators translated to Japanese to study the SDG co-occurrence and to visualize SDG interlinks. Li et al. [26] presented a method for mapping text to SDGs and targets through a lexicon of search queries relevant to each SDG/target. The relevance of an SDG/target for a text was determined using the number of its mentions in the text. Sovrano et al. [27] presented a method for multi-label classification of UN Resolutions to SDGs at the paragraph level. Text representation methods such as Term Frequency – Inverse Document Frequency (TF-IDF), average GloVe embeddings, and pre-trained Universal Sentence Encoder models were used. Meier et al. [28] presented an open-source R package detecting mentions of SDGs in text using existing labeling methods already presented in the research literature. Addition of new methods was also possible. The methods recognized mentions of SDG-related keywords in text. Wulff et al. [29] extended the previous research paper [28] by showing that an ensemble method combining different labeling methods improved the performance of a single method. In addition, the authors compared the performance of the different labeling methods and concluded that fine-tuning language models for that purpose was a promising but still unexplored research direction. Hajikhani and Cole [30] compared models specifically developed to detect SDGs in text with general-purpose large language models (LLMs) such as GPT-3.5. The models used TF-IDF weighting, Word2Vec embeddings, and the Doc2Vec method [31] in combination with ML classifiers trained on text from scientific publications. The authors concluded that specialized models were more robust and precise but general-purpose LLMs were able to identify SDGs in a broader set of texts.

The goal of the Open SDG (OSDG) project [32] was to integrate various methods for classifying text to SDGs based on ontologies, supervised or unsupervised ML, by creating an ontology of more than 14,000 relevant keywords and mapping them to the themes from Microsoft Academic Graph. Any new text was first classified against those themes through methods using TF-IDF weighting and then mapped to the OSDG ontology. The updated framework, OSDG 2.0, was presented in [33]. It combined keyword-based text classification to SDGs with ML-based classification models. The OSDG Community Dataset, consisting of text excerpts labeled with the SDGs from 1 to 16, was made publicly available. Angin et al. [34] fine-tuned pre-trained BERT and RoBERTa models on the OSDG Community Dataset for multi-label text classification to SDGs. They also considered a conventional NLP pipeline (TF-IDF weighting and ML classifiers). The highest F1 score in the multi-label classification was 0.91, achieved with a fine-tuned RoBERTa model. Hsu et al. [35] classified text against the SDGs using a combination of conventional NLP methods, i.e., a topic model classifier and a semantic link classifier. Fonseca et al. [36] presented a method for mapping patent

[2]https://u4ssc.itu.int/u4ssc-kpi/
[3]https://www.unsdsn.org/

documents to SDGs. TF-IDF, Word2Vec embeddings, and Doc2Vec were used for text representation in combination with ML classifiers trained on text from scientific publications. Guisiano et al. [37] presented a method for multi-label classification of text to SDGs. The training/test data consisted of 724 texts with an average of 374 words and was used to fine-tune a pre-trained BERT model for the purpose. Smith et al. [38] applied NLP methods, including Doc2Vec and network analysis, to yearly UN SDG Progress and Information reports to study SDG interactions. Fotopoulou et al. [39] presented a knowledge graph that facilitates the tracking of the progress in achieving the SDGs at national and regional level. Several existing ML approaches (e.g., [25], [38]) were mentioned by the authors as applicable in populating and analyzing the knowledge graph.

Mishra et al. [40] proposed a method that generates ontologies from text data (e.g., Wikipedia, social media, blog posts, news articles) to anticipate the impact of policymakers' decisions on climate change (SDG 13). The process consisted of entity extraction, relation extraction, and ontology formation. A fine-tuned RoBERTa model was used for entity extraction, while Graph Convolutional Networks and multi-head attention layers for relation extraction. Cho and Ackom [41] evaluated national commitments to SDGs and emissions reduction of 67 countries through comparison of their action plans reported in VNRs and Nationally Determined Contributions. TF-IDF weighting was used to represent VNRs into vector form, multidimensional scaling to reduce the vector dimension into a lower one, and cosine distance to analyze vector distribution in space in relation to economic/geographical/environmental features. Koundouri et al. [42] analyzed if the SDGs are integrated into 74 European Green Deal policy documents between 2019 and 2023. A custom dataset of 35,001 text excerpts describing the SDGs (coming from OSDG Community Dataset and SDG-Tracker among the rest) was used to fine-tune a pre-trained BERT model in classifying policy documents to their related SDGs. Benjira et al. [43] studied indicator computation using LLMs and knowledge graphs. They used rule-based filtering and LLMs for schema mapping, to find links between diverse data sources and indicator metadata about their computation. They joined the mappings in a knowledge graph to allow querying the graph topology on indicator computation. Larosa et al. [44] used LLMs (ClimateBERT and Gemini 1.0) and prompt engineering methods to process climate and sustainability policies to classify them into relevant SDGs and to find synergies and trade-offs between the SDGs. The authors state that the 80% match with expert labels was promising although the score was imbalanced among the SDGs. Koundouri et al. [45] analyzed the relatedness of 44 Human Security reports to the SDGs using keyword-based matching, TF-IDF weighting, and Random Forest classifier. The results were compared to those achieved by language models such as BERT, DistilBERT, and ELECTRA, for which the authors noted that their performance was hindered by the small-sized domain-specific training dataset. The method was made available through a web application. Li et al. [46] analyzed China's attention towards the SDGs in the Government Work Reports between 2010 and 2020. To define different explanatory variables used with econometric empirical models, among the rest, the authors combined SDG-related word and phrase frequency analysis, TF-IDF weighting, cosine-based text similarity. Strelkovskii and Komendantova [47] used the SDG Mapper tool [48] to find SDG mentions in 66 national hydrogen strategies, quantify the frequencies of SDG-related keywords, and visualize them. The SDG Mapper [48] relied on SDG and target-related keywords defined by text mining and domain experts, followed by a keyword matching procedure to identify them in text. Raman et al. [49] performed a variety of SDG-related analyses of Twitter posts. The analyses included term weighting using TF-IDF, topic modeling with Latent Dirichlet Allocation, and fine-tuning a pre-trained BERT model to classify tweets to SDGs which achieves an overall F1 score of 0.82. The fine-tuning and evaluation datasets were sampled from a dataset consisting of around 57,843 tweets. Morales-Hernández [50] compared the performance of multi-label classification models in classifying research articles to the SDGs. The Dimensions database of research articles labeled with SDGs was used for training and evaluation. For the period between 2015 and 2021, 180,852 articles from organic agriculture were selected and represented through their title and abstract. Naive Bayes, Logistic Regression, Support Vector Machines, and Random Forest classifiers were compared. Nedungadi et al. [51] analyzed research articles between 2013 and 2024 using BERTopic modeling to detect the influence of big data and AI on the SDGs. The titles and abstracts of 1,288 articles labeled with SDGs from the Dimensions database were used. The topics and their representative keywords were identified.

To visually present the topics prevalent in the related articles, we analyzed the co-occurrence of keywords in their abstracts with VOSviewer software [52] (configuration details given in Appendix B). The results are presented using a keyword co-occurrence network (Figure 1) and keyword density visualization (Figure 2). Figure 1 illustrates the keyword (node) clusters and keyword links. The clusters have different colors. The thickness of the link corresponds to the strength of the connection between the two keywords, i.e., the frequency of their simultaneous appearance in the abstracts. The size of the nodes reflects the number of abstracts in which they appear. The most common keyword is "SDG" (including all its synonyms), but other common keywords are "target", "research", "progress", "text", and "data". The green cluster (in general) refers to the areas of sustainable development covered by the articles, where in addition to keywords such as "progress", "integration", and "gap", "climate action", "health", "clean energy", "water" are also common, among the rest. The red and

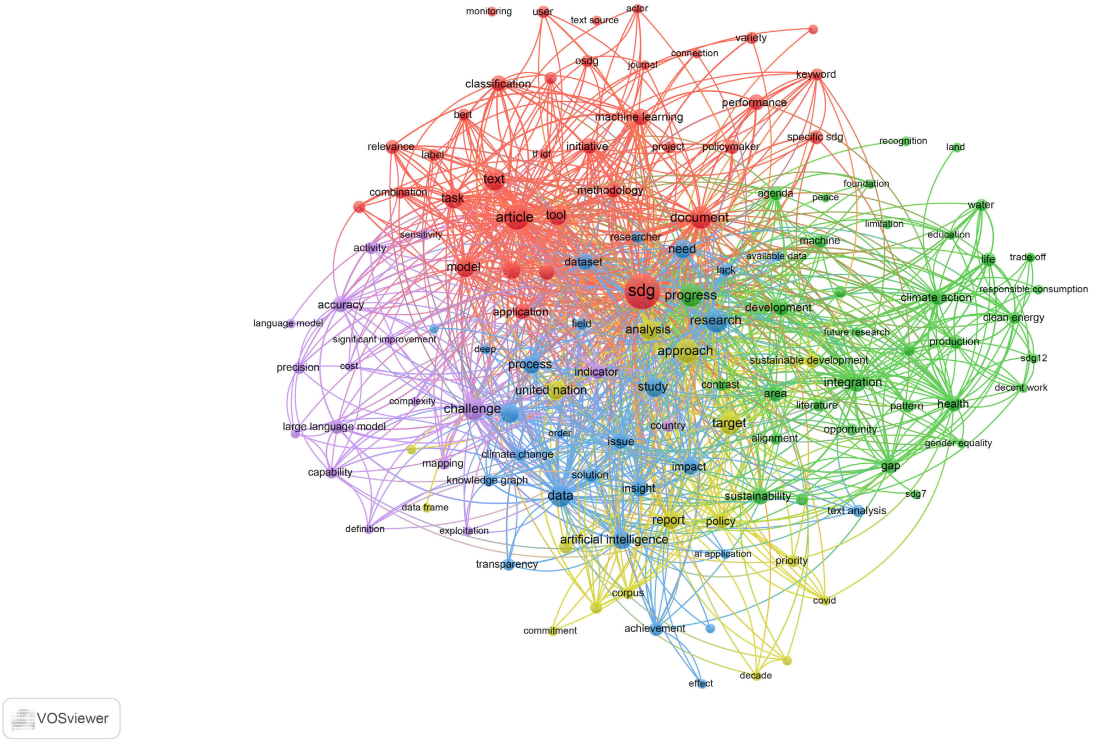

FIGURE 1. Keyword co-occurrence network created from the abstracts of the articles referenced in Section II-B. Created with the VOSviewer software [52].

purple clusters are mainly composed of methodology-related keywords. The first contains common keywords such as ''text'', ''document'', ''model'', ''task'', ''machine learning, ''classification'', as well as such specific to those fields like ''performance'', ''keyword'', ''BERT'', and ''TF-IDF''. The second contains key-phrases like ''large language model'', ''language model'' and their relevant performance-related attributes (e.g., ''accuracy'', ''capability''). Keywords such as ''transparency'', ''cost'', and ''complexity'' appear near ''large language model'' and ''artificial intelligence'' as well. While the generic keyword ''document'' is the most prevalent, specific types of documents are also mentioned (e.g., ''report'' and ''policy''). Figure 2 illustrates the density of different parts of the network. It brings additional clarity, as it displays certain keywords that do not appear in Figure 1 due to space limitations.

In the remainder of this section we discuss the identified gaps in the related work which our method tries to fill from three aspects, i.e., (1) target classes in the text classification - SDGs and/or targets, (2) text on which the methods were evaluated, and (3) comparison of state-of-the-art NLP methods. First, most of the related methods for text classification to SDGs and/or targets [24], [25], [27], [34], [35], [36], [37], [42], [44], [45], [50] only allow classification to the 17 SDGs, not addressing the more challenging problem of text classification to the 169 targets, a gap we tried to fill with our paper (in the context of indicator descriptions). Only one related paper [26] classifies text to both SDGs and targets, but it differs from our work methodologically (they use a more conventional NLP approach) and in its objective (general text classification vs. classification of indicator descriptions). Therefore, the performance of their method in multi-class and multi-label indicator description classification should be evaluated additionally. Second, while many related papers address the problem of general text classification to SDGs/targets, only few are specifically evaluated in classification of indicator descriptions [24], [25]. Compared to the work of [24], our method is mainly a representation learning method that can be used for non-parametric classification of both SDGs and targets at the same time, while capturing the relatedness between the SDGs. Additionally, our work was evaluated on a larger indicator set, offered comparison of a larger number of sentence encoders from several categories, proposed a slightly more complex fine-tuning dataset creation process (in our opinion), and offered a more thorough post-hoc analysis of the results. The method presented by [25] does not allow indicator classification to targets as our method does, is limited to Japanese input text, and focuses mainly on the problem of SDG interlink detection. Third, while several related papers use language models [24], [25], [29], [34], [37], [42], [45], large language models [24], [30], [44] and sentence encoders [27] for text classification to SDGs/targets, a comprehensive benchmarking of the

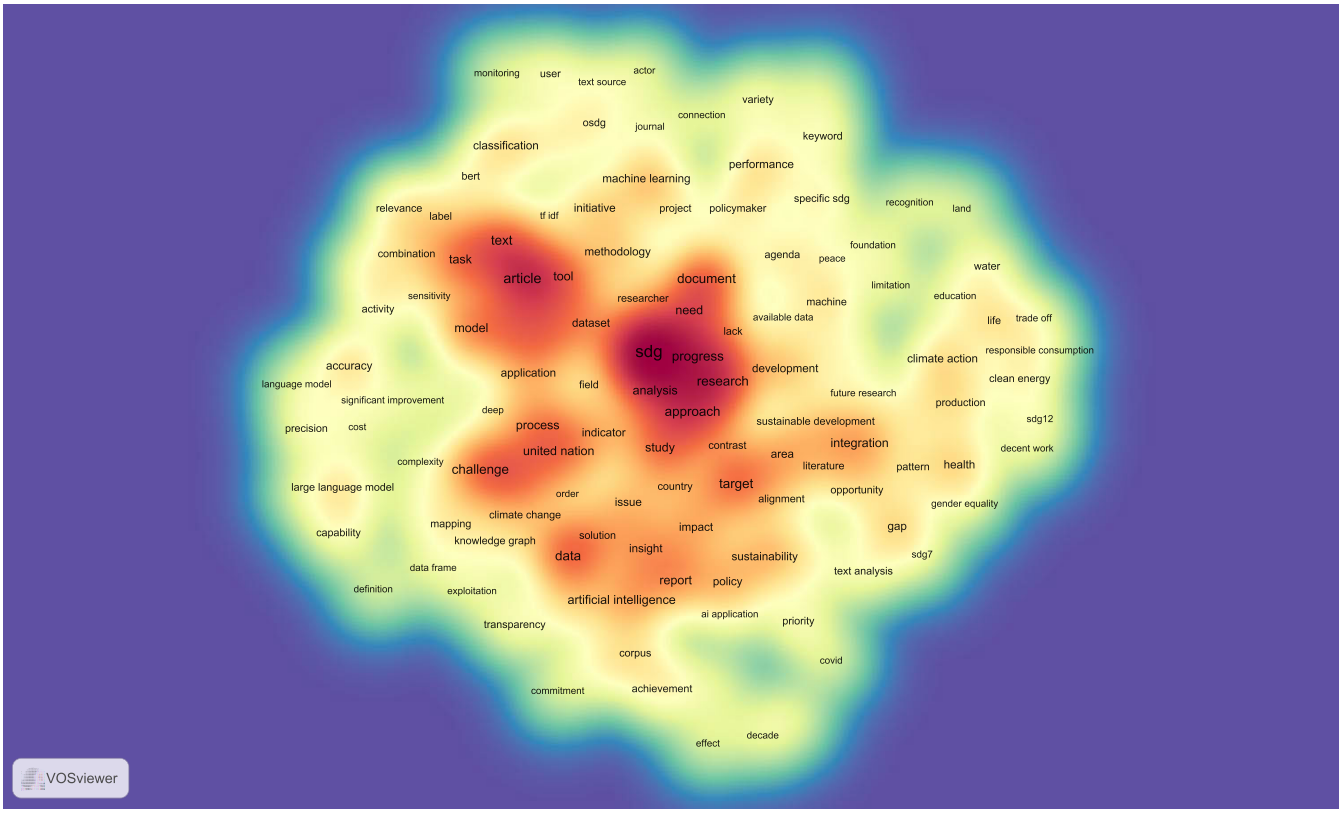

FIGURE 2. Keyword density visualization created from the abstracts of the articles referenced in Section II-B. Created with the VOSviewer software [52].

strengths and weaknesses of such models in associating indicators with SDGs and targets is lacking and is more than required. In this paper, we contributed in filling that gap in the context of sentence encoders. This paper further extends the work presented in [53] by adding preliminary similarity-based comparison of the fine-tuning and test datasets (described in Sections III-B and V-A) and adding four new real-world indicator test datasets in the testing phase (test datasets 2-5, described in Section IV-A2). As a result, this paper significantly extends the test result analysis (Section V-C) and the discussion of research questions 2 and 3 (Section VI), compared to [53].

III. METHOD

A. OVERVIEW

The model-agnostic Embed4SD framework for associating indicators with SDGs and targets based on their textual descriptions is illustrated in Figure 3. It formulates the solution to the problem as text classification, first by representing SDG, target, and indicator descriptions in a common vector space using a shared sentence encoder, then by classifying the indicator into one or multiple classes (SDGs or targets depending on the task being solved) using a classification algorithm. The framework development involved three phases briefly described below.

The first phase involved creation of custom fine-tuning/test datasets, preliminary analysis of those datasets, and benchmarking pre-trained general-purpose sentence encoders on domain-specific validation tasks formulated as (1) multi-class classification of indicator descriptions to one of the 17 SDGs they are the most associated with (abbreviated as MC-IND-SDG) and (2) multi-class classification of indicator descriptions to one of the 169 targets they are the most associated with (MC-IND-TRG), i.e., the two main tasks of interest. Based on the encoders' average performance on the validation tasks, those with the highest performance were selected for further domain-specific fine-tuning.

In the second phase, the selected sentence encoders were fine-tuned on domain-specific tasks. The fine-tuning was done on one task at a time, selected from a set of fine-tuning tasks formulated as a multi-class classification of short textual descriptions to the SDGs (abbreviated as FT-SDG) or targets (abbreviated as FT-TRG) they describe. The fine-tuning used contrastive representation learning with the triplet network architecture [54]. The fine-tuned sentence encoders (in combination with a classifier) were evaluated on two main test tasks similar to the validation tasks, as well as on two auxiliary tasks formulated as (1) multi-label classification of indicators to associated SDGs (ML-IND-SDG) and (2) multi-label classification of each SDG to related SDGs (ML-SDG-SDG). The auxiliary tasks were not seen during

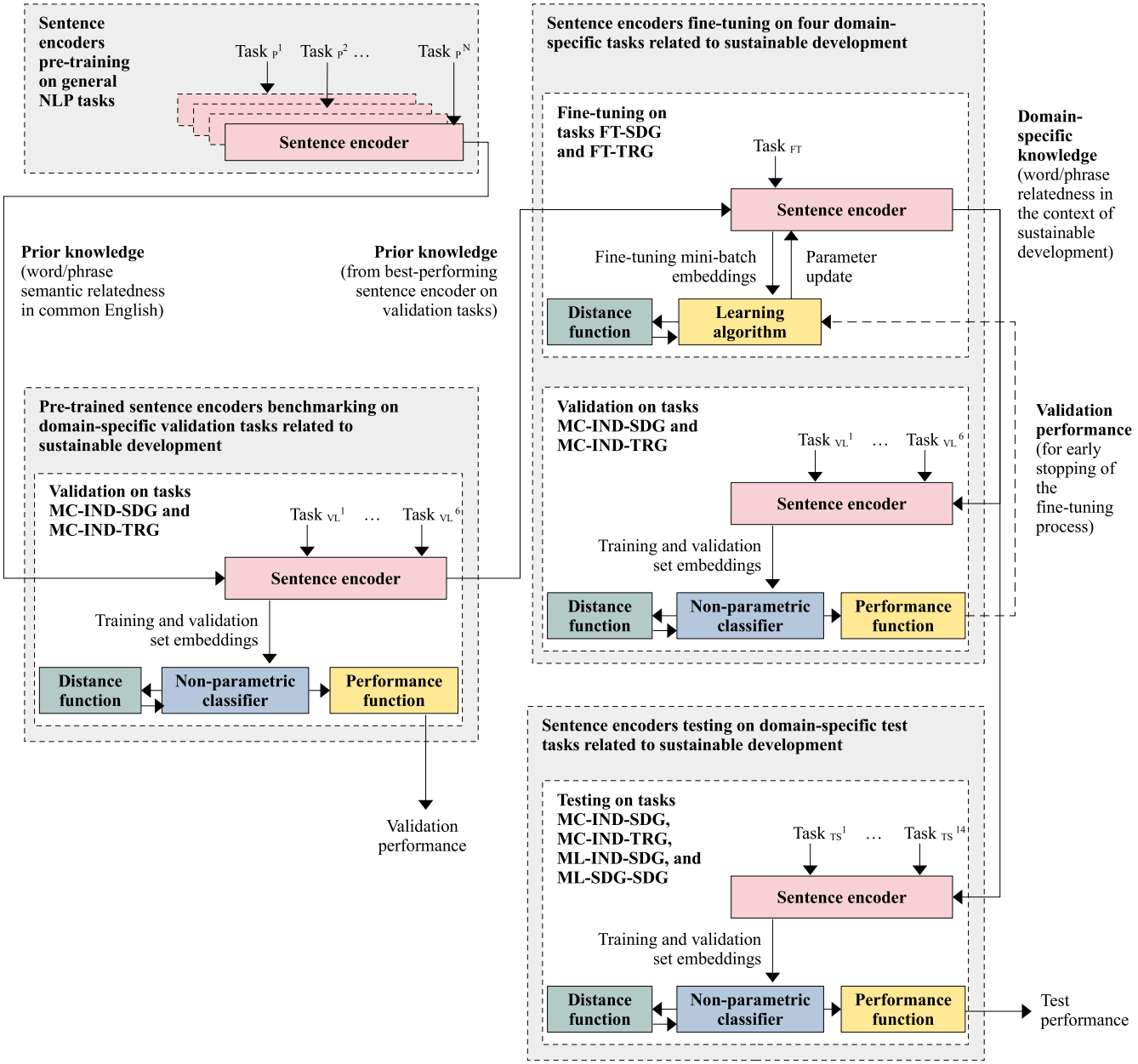

FIGURE 3. Conceptual diagram of the main components and their interactions through different development phases of the Embed4SD framework.

encoder fine-tuning and validation. Therefore, with a small modification of the classifier to allow it to do multi-label classification, in a zero-shot learning manner, we evaluated if the fine-tuned encoders had learned the mutual relations between the SDGs from the textual fine-tuning data, even though they had not been explicitly given such information in the fine-tuning process. In the testing, different indicator sets used at the national, regional, or local level of governance were used with each test task, as applicable. To have a baseline against which to measure the improvement after fine-tuning, the selected encoders were evaluated on the test tasks and indicator sets prior to their fine-tuning.

The third phase involved post-hoc analysis of the test results with xAI methods, to better understand the factors that influenced them and gain insights for future improvements of the framework. The remainder of this section describes the generic aspects of the framework, while Section IV describes the specific experimental choices.

B. DATASET CREATION PROCESS, PRELIMINARY ANALYSIS AND ENCODER BASELINE BENCHMARKING

The datasets used to fine-tune the sentence encoders consisted of short textual excerpts describing the SDGs and targets from various aspects (e.g., their main aim, definitions

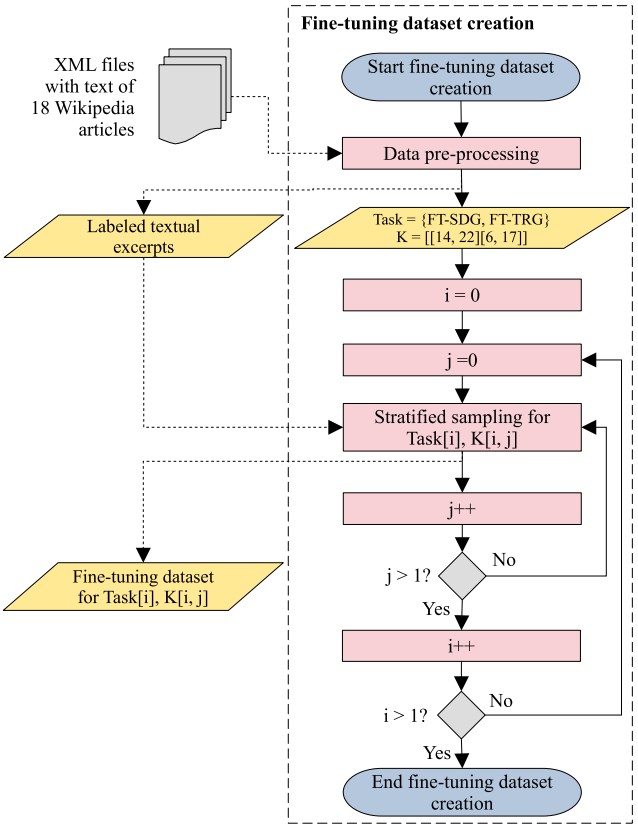

FIGURE 4. Flowchart of the fine-tuning dataset creation process.

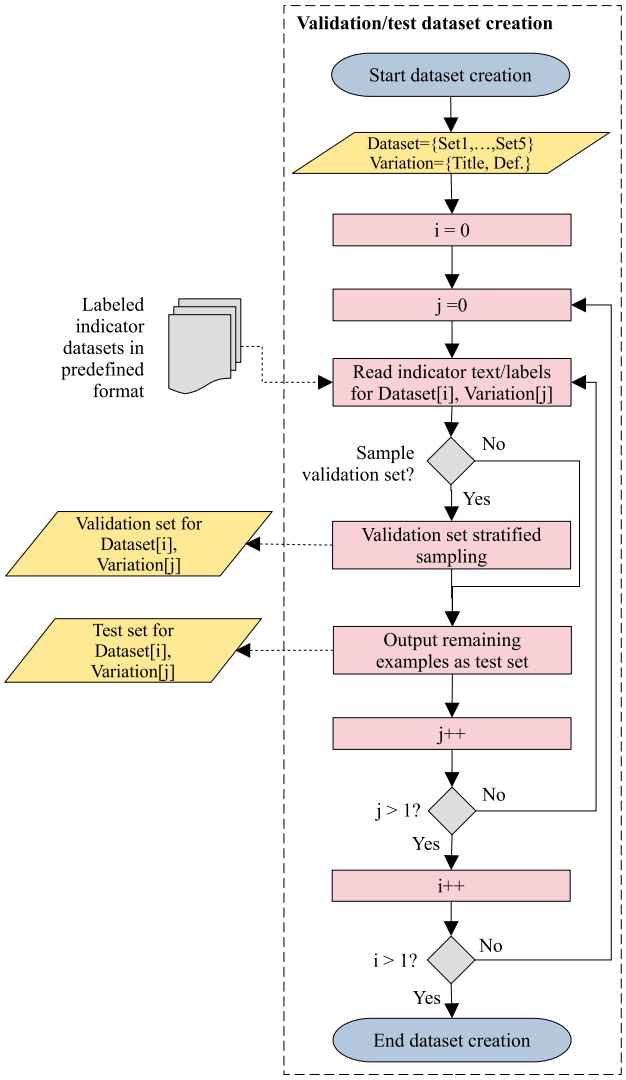

FIGURE 5. Flowchart of the validation and test dataset creation process.

of related concepts, statistical data, related challenges, related organizations). All excerpts were labeled with the SDG they describe, while those describing a particular target were labeled with both the SDG and the target label. The excerpts were also labeled with the specific aspect of the SDG they described. An initial dataset was created and used to sample several fine-tuning datasets using different stratified sampling strategies. The main idea was to analyze how the fine-tuning dataset size and structure impacted the results (experimental setup described in Section IV-A1). The fine-tuning dataset creation process is illustrated in Figure 4.

The validation and test datasets consisted of indicators taken from indicator frameworks used at the national, regional, or local level of governance, labeled with one or multiple SDGs and targets they were associated with (depending on the ground truth labels available in the indicator frameworks themselves). Two variations of the test dataset were created from each indicator framework, differing in the length of the text used to represent the indicators (experimental setup described in Section IV-A2). The process of creating the validation and test datasets is illustrated in Figure 5.

To get apriori insights in the content of the datasets and eliminate potential biases of the proposed method, prior to their use, all test set variations were subjected to similarity-based comparison with the fine-tuning examples in a completely different vector space than the ones produced by the benchmarked sentence encoders. All fine-tuning examples from the fine-tuning datasets were represented as vectors using the bag-of-words method and TF-IDF weighting. Using the same model fitted on the fine-tuning vocabulary, all descriptions of test examples were embedded in the same vector space and compared with all fine-tuning examples through cosine similarity. For each description of a test indicator, only the highest cosine similarity score was retained. The summary statistics of those similarity scores was then calculated by test set to see if there are test examples that are very similar to the fine-tuning examples (indicating that test examples appear in our fine-tuning sets). In our future work, we plan to compare the datasets in larger number of different vector spaces, produced by other pre-trained sentence encoders. The similarity-based comparison process is illustrated in Figure 6.

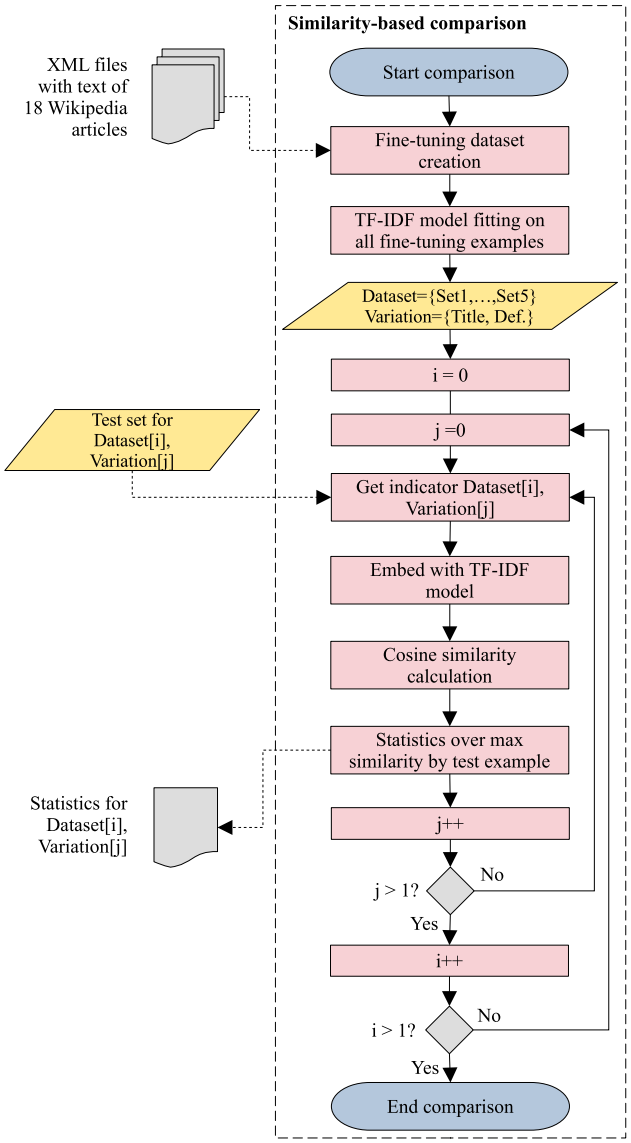

FIGURE 6. Flowchart of the similarity-based comparison of fine-tuning and test examples. The fine-tuning dataset creation is shown in Figure 4.

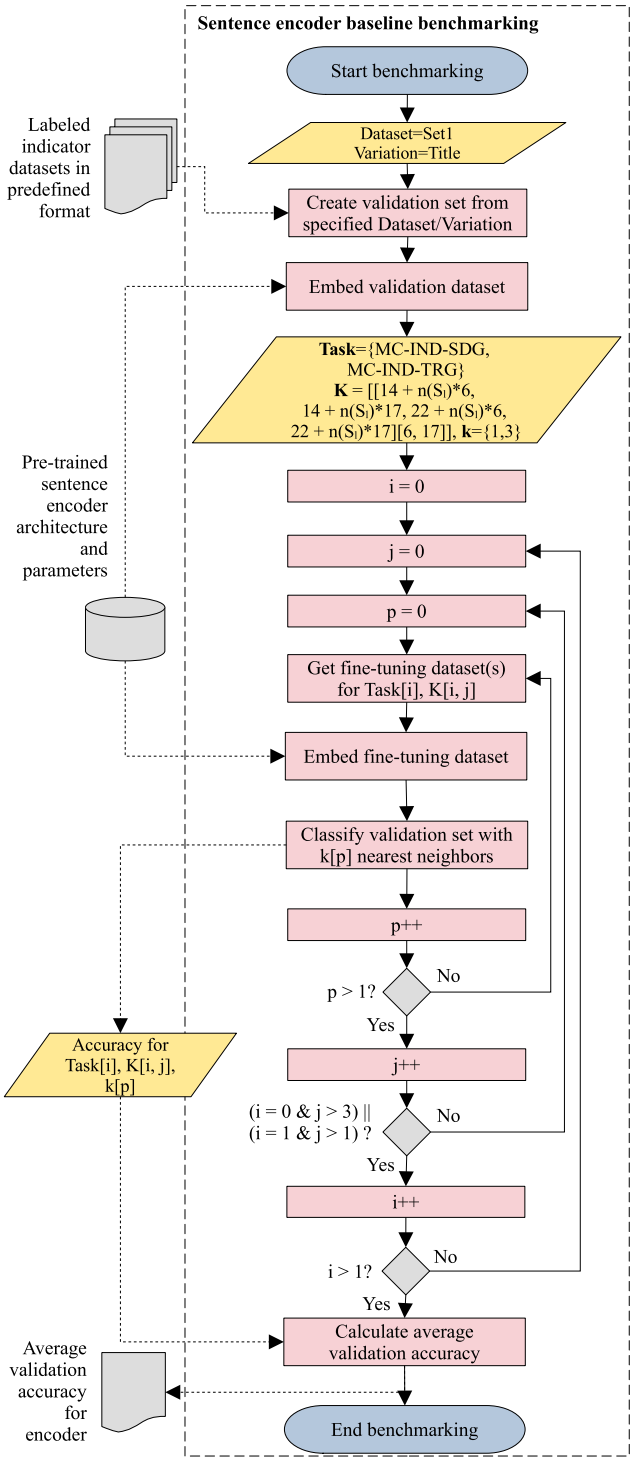

FIGURE 7. Flowchart of a sentence encoder benchmarking on all validation tasks/configurations. For validation set creation, see Figure 5.

The baseline benchmarking of the pre-trained general-purpose sentence encoders (their pre-training is outside the scope of this article) was done on the two validation tasks MC-IND-SDG and MC-IND-TRG. Although the use of language models based on DL, which capture general language characteristics (e.g., [55], [56], [57]), is a common practice today when it comes to solving NLP tasks (mainly through the concept of transfer learning), such models usually do not output ready-to-use sentence embeddings but require domain-specific fine-tuning. Pre-trained sentence encoders (e.g., [58], [59], [60]) usually fine-tune such language models (mainly through contrastive representation learning) to output meaningful sentence embeddings for straightforward sentence comparison using a distance metric. Therefore, such encoders allow for their easy clustering or classification with non-parametric algorithms, which was the main reason for our experimentation with that type of language models. Based on the encoders' average performance on the validation tasks, those with the highest performance were selected for further domain-specific fine-tuning. The process is illustrated in Figure 7.

C. SENTENCE ENCODER FINE-TUNING, VALIDATION, AND TESTING

The goal of the fine-tuning process was the adjustment of the pre-trained sentence encoder parameters to achieve improved results on the main and auxiliary test tasks over their baseline results. In each domain-specific fine-tuning task (either FT-SDG or FT-TRG), N represents the number of classes to which the examples were classified, while K represents the approximate number of examples by class in the fine-tuning set (several different values were evaluated in the experiments described in Section IV-A1). Each fine-tuning task was represented through its fine-tuning set of m examples, $\mathbf{D}_{FT}$, its distance metric measuring the distance between the embeddings of two examples in the target representation space, $d(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})$, and its objective function used to optimize the sentence encoder parameters, $J(\boldsymbol{\theta})$. Each fine-tuning example was a pair $(t^{(i)}, \mathbf{y}^{(i)}) \in \mathbf{D}_{FT},\ i = \{1, \ldots, m\}$ of short text describing an SDG or target, labeled with either the SDG or the target it describes, depending on the task. The text $t^{(i)}$ had an initial representation $\mathbf{x}^{(i)}$ belonging to a representation space $\mathbf{R}^s$, while the sentence encoder implemented a function $f : \mathbf{R}^s \rightarrow \mathbf{R}^v$ parameterized by a parameter vector $\boldsymbol{\theta}$ with the purpose of projecting the initial representation $\mathbf{x}^{(i)}$ to a new and more dense representation $\mathbf{z}^{(i)} = f(\mathbf{x}^{(i)})$, belonging to a representation space $\mathbf{R}^v$, where usually $v \ll s$. The learning process optimized the parameter vector $\boldsymbol{\theta}$ so that the representation $\mathbf{z}^{(i)}$ was useful for the validation and test tasks. In this article we use a vector notation for the label, $\mathbf{y}^{(i)}$, to represent its 1-of-N encoding.

We used the triplet network architecture, where triplets $((t^{(a)}, \mathbf{y}^{(a)}),\ (t^{(p)}, \mathbf{y}^{(p)}),\ (t^{(n)}, \mathbf{y}^{(n)}))$ were formed from fine-tuning examples and used to optimize the parameter vector $\boldsymbol{\theta}$. Each triplet had an anchor example a, a positive example p ($p \neq a$) that was related to the anchor in some human-defined way (sharing the same class in this case, $\mathbf{y}^{(a)} = \mathbf{y}^{(p)}$), and a negative example n ($n \neq a$) that was unrelated to the anchor (having a different class than the anchor in this case, $\mathbf{y}^{(a)} \neq \mathbf{y}^{(n)}$). Therefore, the objective of the learning process was to bring the projected embeddings of the anchor and the positive example closer in the target representation space $\mathbf{R}^v$ for at least a margin α than the anchor and the negative example, i.e., $(d(\mathbf{z}^{(a)}, \mathbf{z}^{(n)}) - d(\mathbf{z}^{(a)}, \mathbf{z}^{(p)})) > \alpha$. The three embeddings were combined in the triplet loss function [61], with the purpose of increasing the distance between the anchor and the negative example for a margin α, compared to the distance between the anchor and the positive example.

In the triplet network architecture, the set of all valid triplets is task-dependent. Although the number of valid triplets may be very large, not all triplets contribute to parameter improvements during training [61]. Instead of using all valid triplets, the hard triplet mining strategy forms a triplet for an anchor by searching for its most distant positive example and its closest negative example, but these hard triplets may sometimes lead to fast convergence to local minima [61]. In this work, we used the batch hard triplet loss [62], which mines the hardest positive and hardest negative examples in a mini-batch for each anchor based on a pre-specified margin α, as given in Eq. 1. For a specific anchor a, the hardest positive example p in a mini-batch ($p \neq a$) was the one belonging to the same class as the anchor ($\mathbf{y}^{(a)} = \mathbf{y}^{(p)}$) and having the largest distance from the anchor ($d(\mathbf{z}^{(a)}, \mathbf{z}^{(p)})$) in the target vector space. The hardest negative example n in a mini-batch ($n \neq a$) was the one belonging to a different class than the anchor ($\mathbf{y}^{(a)} \neq \mathbf{y}^{(n)}$) and having the smallest distance from the anchor ($d(\mathbf{z}^{(a)}, \mathbf{z}^{(n)})$) in the target vector space. The loss function included only those anchors for which the difference between the two distances exceeded the predefined margin α, i.e., resulted in a positive value. The distance metric of choice was the angular distance, defined in Eq. 2 and based on the well-known cosine similarity.

During validation and testing, we used the fine-tuned sentence encoders as feature extractors and combined them with a classification algorithm. The validation and test tasks were represented through a training set $\mathbf{D}_{TR}$, a validation $\mathbf{D}_{VL}$ or test $\mathbf{D}_{TS}$ set, accordingly, a distance metric in the target representation space $d(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})$, and a performance metric $p(\mathbf{y}^{(i)}, \hat{\mathbf{y}}^{(i)})$. Within each validation and test task (MC-IND-SDG, MC-IND-TRG, ML-IND-SDG, and ML-SDG-SDG) different experimental settings were evaluated, differing in the approximate number of training examples by class (approximate K), as described in Section IV-B. The fine-tuning process of a single sentence encoder is illustrated in Figure 8.

$$J(\boldsymbol{\theta}) = \sum_{a=1}^{64} [\max_{\substack{p=1..64 \\ a \neq p \\ \mathbf{y}^{(a)}=\mathbf{y}^{(p)}}} d(\mathbf{z}^{(a)}, \mathbf{z}^{(p)}) - \min_{\substack{n=1..64 \\ a \neq n \\ \mathbf{y}^{(a)} \neq \mathbf{y}^{(n)}}} d(\mathbf{z}^{(a)}, \mathbf{z}^{(n)}) + \alpha]_+ \quad (1)$$

$$d(\mathbf{z}^{(a)}, \mathbf{z}^{(p)}) = 1 - \frac{\mathbf{z}^{(a)}\mathbf{z}^{(p)}}{||\mathbf{z}^{(a)}||||\mathbf{z}^{(p)}||}, d(\mathbf{z}^{(a)}, \mathbf{z}^{(p)}) \in [0, 2] \quad (2)$$

D. POST-HOC ANALYSIS

To better understand the factors that influenced the evaluation results, we first tried to identify as many of them as possible and then analyzed their influence on the results using methods from xAI. These included the various decisions we made during the fine-tuning and testing processes. The idea was simple, i.e., all factors of interest were represented as input features to a meta-model, i.e., linear regression, which was then trained to predict the performance by SDG that was actually achieved by our fine-tuned sentence encoders and classifier on the test sets. The training set on which the linear regression was trained consisted of all the different fine-tuning and testing configurations, described through the mentioned factors. The contribution of each feature to the prediction made by the linear regression model for each individual training example was then calculated with the Shapley Additive Explanations (SHAP) method [63]. The

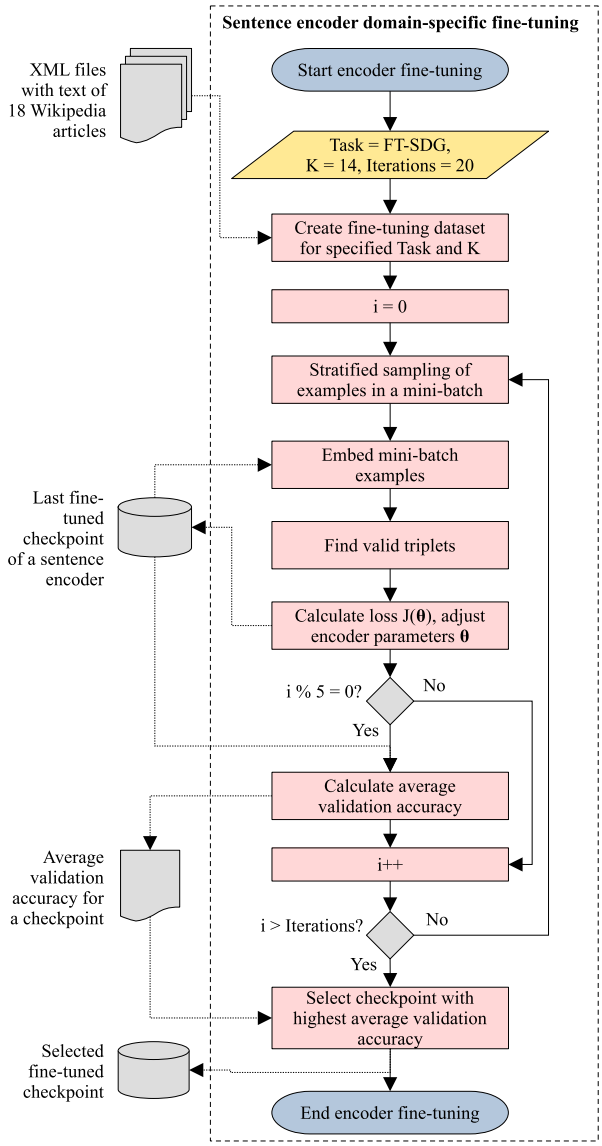

FIGURE 8. Flowchart of the fine-tuning process of a sentence encoder on the task FT-SDG, K=14. The process is the same for all fine-tuning tasks. The fine-tuning dataset creation is shown in Figure 4. The calculation of the average validation accuracy follows the process shown in Figure 7.

method is based on concepts from coalition games theory and explains feature attributions to the predictions made by an ML model for individual examples [64]. When calculating the SHAP values for each example, we wanted the algorithm to take into consideration the feature correlation and spread the credit between correlated features, as explained in [64]. The process is illustrated in Figure 9.

IV. EXPERIMENTAL DESIGN

A. DATASET CREATION PROCESS, PRELIMINARY ANALYSIS AND ENCODER BASELINE BENCHMARKING

1) FINE-TUNING DATASET CREATION PROCESS

The datasets used to fine-tune the sentence encoders were sampled from a custom-created dataset consisting of 1,815 text excerpts of similar length, extracted from

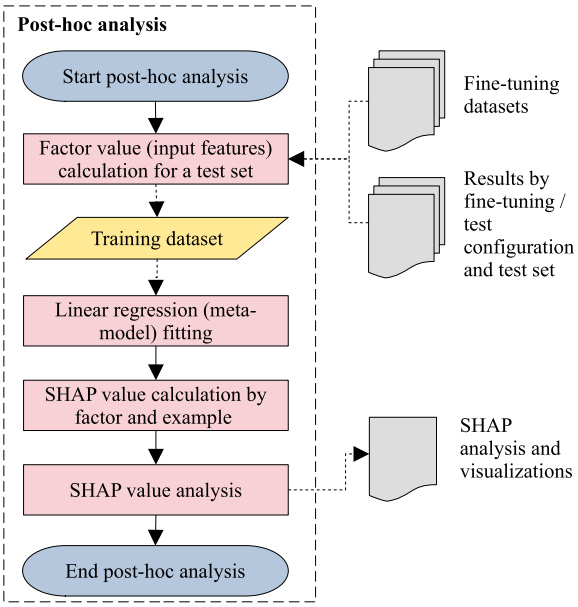

FIGURE 9. Flowchart of the post-hoc analysis.

18 English-language Wikipedia articles devoted to the SDGs. Each excerpt was labeled with the SDG and target (where applicable) it described, based on the article section it was extracted from. Those articles were (1) the article titled "Sustainable Development Goals", providing a brief description of all 17 SDGs, and (2) the articles dedicated to each of the 17 SDGs, the first called "general" article and the second "SDG-specific" articles in the sections that follow. For the exact article URLs and revision IDs see Appendix C. The text of the 18 Wikipedia articles, downloaded in XML format, was subjected to custom pre-processing consisting of four steps, i.e., (1) text extraction, (2) text cleaning, (3) sentence extraction, and (4) text excerpt extraction, as illustrated in Figure 10. The pre-processing was followed by a fifth step, i.e., a process of stratified sampling of the fine-tuning datasets from the 1,815 similar-length text excerpts. For an illustration of the process, see Figure 4.

Step 1: Text Extraction. A selected set of article sections was extracted from the XML files and cleaned from HTML and Wikipedia-specific XML markup. From the general article, the sections devoted to each SDG were extracted and labeled with the SDG they referred to. From the SDG-specific articles, the lead section and the sections with titles containing a selected set of phrases were extracted (see the GitHub repository) and labeled with the SDG the article referred to. From the section devoted to the targets of an SDG, each subsection devoted to a specific target was extracted and labeled with both the SDG and the target it referred to.

Step 2: Text Cleaning. The text extracted in the previous phase was subjected to cleaning consisting of four steps illustrated in Figure 10. The purpose of the indicator title removal step was to find all mentions of SDG indicator titles in the text and remove them, as these titles were part of

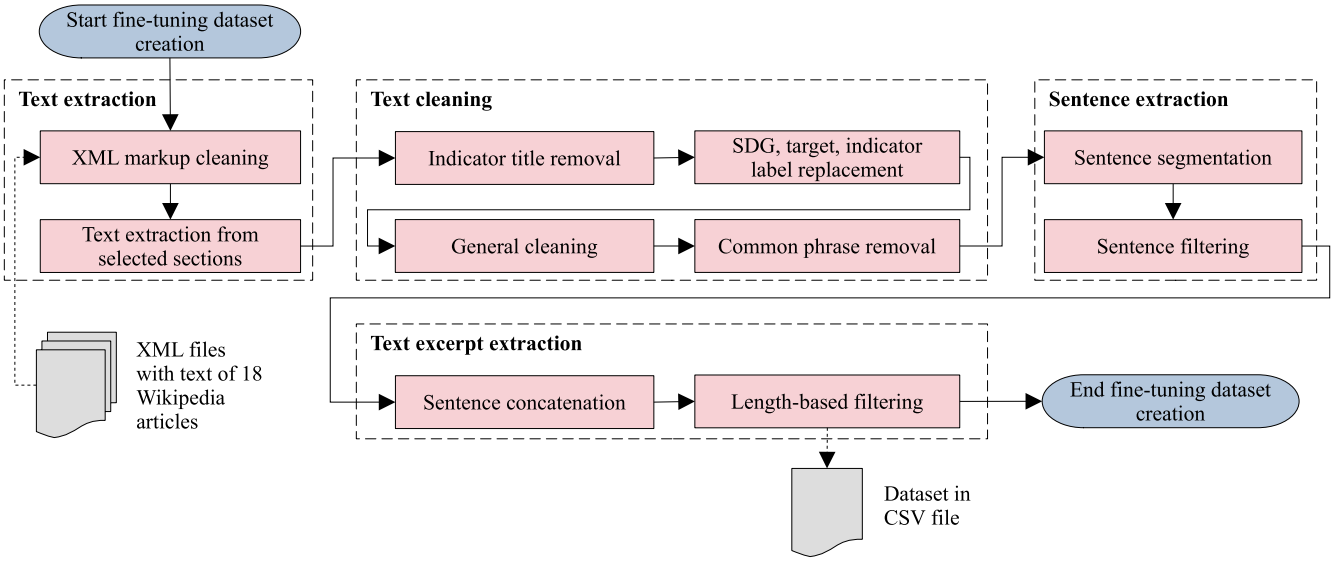

FIGURE 10. Flowchart of the pre-processing of 18 Wikipedia articles to extract candidate examples for fine-tuning datasets.

the test set. It was done by searching for common patterns, determined with prior text analysis, and their replacement with general phrases, such as "indicator", "the indicator" and similar. The second step removed mentions of SDG, target, and indicator labels from the text. Each mention of an SDG, a target, or an indicator label was replaced with a generic phrase, such as "the goal", "the target", "the indicator", or their variation, depending on the sentence context. The general cleaning removed general patterns such as list item letters or numbers from the text. The common phrase removal step removed a small set of very common phrases from the text, which did not appear as separate sentences to be filtered by the sentence filtering in the next phase. They could have negatively affected the learning process by making the excerpts that contained them appear similar, even when this was not the case.

Step 3: Sentence Extraction. In the sentence extraction phase, all paragraphs in each section were divided into their constituting sentences. The purpose was to remove common sentences that did not contribute to distinguishing the SDG or target descriptions from each other. These sentences shared common terminology, therefore, a simple bag-of-words method with TF-IDF weighing was used to represent each sentence in a common vector space. The minimum document (sentence in this particular case) frequency was set to min_df=10 documents. Only single words were weighted by the method after removing the stop words. The vectors were then clustered using the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm [65], with $\epsilon = 0.3$ and min_samples=10, to identify large clusters of similar sentences. The sentences that were retained for further processing were those labeled as outliers by the algorithm (more than 96% of the total number of sentences). Cosine similarity was used as a similarity metric, and the clustering

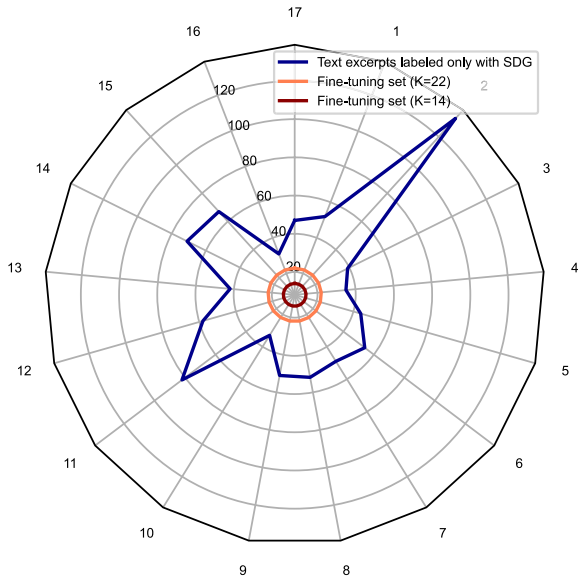

FIGURE 11. Distribution by SDG of all extracted excerpts labeled with SDG label only and the examples sampled for each of the two FT-SDG fine-tuning sets.

hyperparameters were selected experimentally. Since the distribution of the number of retained sentences by SDG and target was highly imbalanced, with significant variations in their length, combining them into text excerpts of comparable length was needed. For clarity, Algorithm 1 summarizes the sentence extraction process in a simplified form (less relevant configuration parameters are omitted).

Step 4: Text Excerpt Extraction. After the sentence extraction phase, the sentences were concatenated into short excerpts. The exact order in which the sentences

Algorithm 1 Simplified Version of the *Sentence Extraction* Algorithm (less Relevant Configuration Parameters Are omitted). The Implementation Uses the Scikit-Learn Library (abbreviated as *sklearn*) and Python Built-in Regular Expression Library (*re*). The Variable *sdg* Refers to the SDG Number Described in the Paragraph, *trg* to the Target Number (if any), *sec* Refers to the Title of the Section From Which the Paragraph Was Extracted, and *par* to the Paragraph Text. For the Exact Implementation, See the GitHub Repository [13]

```
all_sentences ← [] /* all sentences in dataset */
paragraph_sentences ← [] /* sentences by paragraph */
filtered_paragraphs ← [] /* OUTPUT: paragraphs with common sentences filtered out */
paragraphs ← [(sdg(1), trg(1), sec(1), par(1)), ..., (sdg(k), trg(k), sec(k), par(k))], sdg ∈ {1, .., 17}, trg ∈ {1, .., 169}
vectorizer ← sklearn.TfidfVectorizer(ngram_range=(1, 1), min_df=10, stop_words='english')
dbscan ← sklearn.DBSCAN(eps=0.3, min_samples=10, metric='cosine')
for (sdg, trg, sec, par) in paragraphs do
    sentences ← re.split(r'. |? |! |;; ', par) /* split paragraph to sentences */
    all_sentences.add(sentences)
    paragraph_sentences.add((sdg, trg, sec, par, sentences))
end for
tf_idf_vectors ← vectorizer.fit_transform(all_sentences).toarray() /* vectorize sentences */
clusters ← dbscan.fit(tf_idf_vectors) /* cluster sentence vectors */
start_index ← 0
for (sdg, trg, sec, par, sentences) in paragraph_sentences do
    retained_sentences ← []
    span ← len(sentences)
    end_index ← start_index + span
    current_clusters = clusters[start_index:end_index] /* get labels for sentences in current paragraph */
    for (sentence, cluster) in zip(sentences, current_clusters) do /* DBSCAN outliers are labeled with -1 */
        if cluster = -1 then
            retained_sentences.add(sentence) /* if the sentence is an outlier, retain it */
        end if
    end for
    start_index ← end_index
    filtered_paragraphs.add((sdg, trg, sec, par, retained_sentences))
end for
```

appeared in the original text and the paragraph breaks were preserved during concatenation, while trying to achieve an approximate value of 30(±10) words per excerpt. Those excerpts with less than 5 and more than 55 words were filtered out. In such a way, sufficient context was captured in the excerpts while keeping their number of words similar. Each excerpt was labeled with the SDG and target labels of the section it was extracted from (see Text Extraction phase).

Step 5: Stratified Sampling of Fine-Tuning Datasets. Four different fine-tuning datasets were sampled from the extracted dataset using stratified sampling strategies. In each fine-tuning set, based on the task (FT-SDG or FT-TRG), the number of classes N corresponded to either the number of SDGs (N=17) or the number of targets (N=169). We experimented with the number of examples by class K, to see if a larger number of examples by class in the fine-tuning set improved the performance or, on the contrary, made it worse. We hypothesized that in FT-SDG, the examples extracted from the more general article sections, e.g., the sections of the general SDG article or lead section of the SDG-specific articles, would actually result in a less noisy fine-tuning set and, consequently, sentence encoder that would perform better on the test tasks. That also applied to FT-TRG, where we hypothesized again that extracting examples from the first sentences/paragraphs describing each target was better. Therefore, in the two FT-SDG datasets, K was selected according to the mean and maximum number of excerpts by SDG, extracted from the general article and the SDG-specific article lead sections, i.e., K=14 in the first and K=22 in the second. The first fine-tuning set (K=14) was composed mainly of textual excerpts extracted from (1) the general article, (2) the lead section, and (3) the "background" section of the SDG-specific articles. The second fine-tuning set (K=22) extended the first with excerpts from the remaining sections of the SDG-specific article. In FT-TRG, K was selected according to the mean and maximum number of examples extracted by target, i.e., K=6 in the first task and K=17 in the second. The examples were those extracted from the section that described the targets in the SDG-specific articles. The distribution of fine-tuning examples by SDG and target in each of the four fine-tuning sets is illustrated in Figure 11 and Figure 12, while the distribution of examples by Wikipedia article section and

SDG in the two FT-SDG fine-tuning sets is illustrated in Figure 13 and Figure 14.

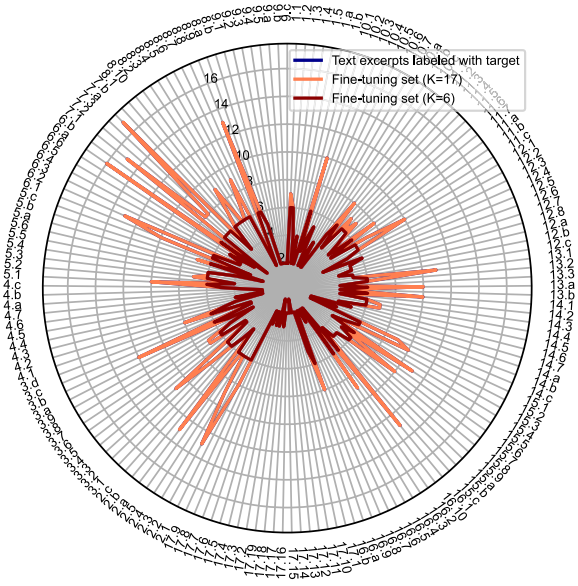

FIGURE 12. Distribution by target of all extracted excerpts labeled with target label and examples in the two FT-TRG fine-tuning sets. The overlap of the two lines indicates that the second fine-tuning set (K=17) includes all extracted excerpts.

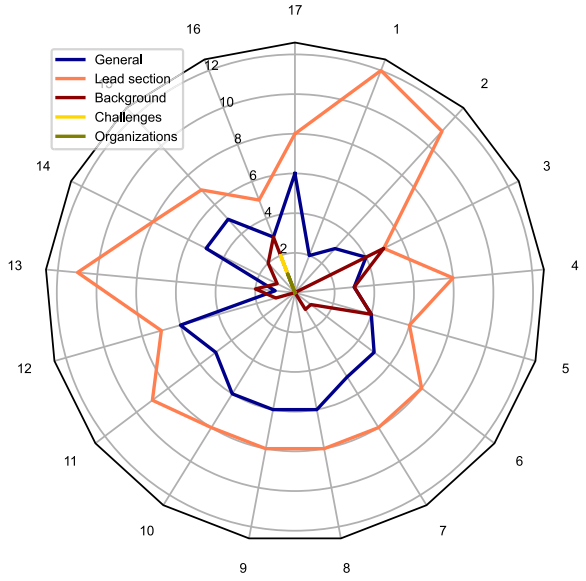

FIGURE 13. Distribution by SDG and Wikipedia article section of the examples in the FT-SDG fine-tuning set N=17, K=14.

2) VALIDATION AND TEST DATASET CREATION PROCESS

Due to the lack of ready-to-use datasets for validation/testing of the proposed framework, we had to adjust several existing indicator frameworks for their use in our main and

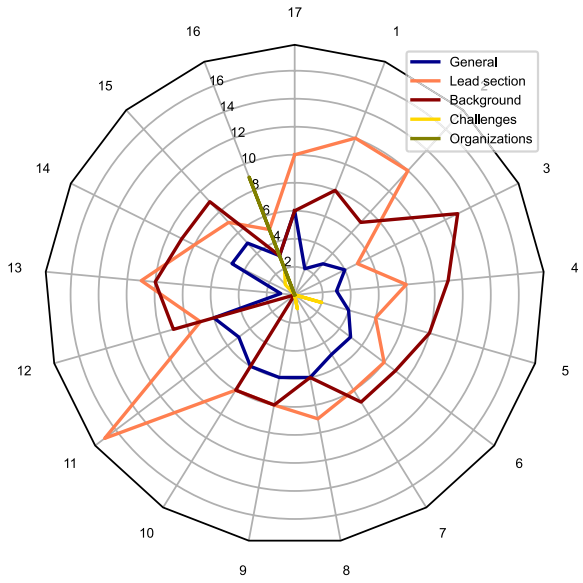

FIGURE 14. Distribution by SDG and Wikipedia article section of the examples in the FT-SDG fine-tuning set N=17, K=22.

auxiliary validation/test tasks. The validation set consisted of indicators sampled from the Global indicator framework of the 2030 Agenda, labeled with SDGs and targets. Five indicator test sets were created from (1) the Global indicator framework of the 2030 Agenda (validation/test dataset 1), (2) the EUROSTAT's EU SDG indicators (test dataset 2), (3) the World Bank's World Development Indicators (test dataset 3, also abbreviated as "WDI SDG" indicators in the remaining text), (4) indicators presented in the European Handbook for SDG VLRs (test dataset 4, also abbreviated as "EU Local SDG" indicators), and (5) the initial set of European regional SDG indicators (test dataset 5, also abbreviated as "EU Regional SDG" indicators). While the indicators from the Global indicator framework of the 2030 Agenda (validation/test dataset 1) and EU Regional SDG indicators were represented only through their titles, two variations were created from the other three test sets. In the first, the indicators were represented through their titles, and in the second, through the indicator title concatenated as a first sentence with an excerpt from the indicator definition containing approximately 30($\pm$10) words. The content and order of the sentences was preserved. The indicators were labeled with one SDG, multiple SDGs, or one target depending on the information available in the data source. Consequently, they were used only in the appropriate main or auxiliary test tasks. A custom sixth dataset for the ML-SDG-SDG task was created as well. For an illustration of the process, see Figure 5.

Validation/Test Dataset 1: Global Indicator Framework of the 2030 Agenda. The validation and test set 1 were sampled from the global indicators of the 2030 Agenda. The refinement in March 2021 was used, consisting of 247 indicators, including the repeating ones. Each indicator

was represented by its title and labeled with the SDG and target it monitored. The repeating indicators were labeled with the multiple SDGs and targets they monitored. In the multi-class classification validation/testing tasks, a prediction that matched any of those multiple labels was considered a correct prediction. The validation-test set ratio was 25%-75%. To ensure representative sets, two criteria were taken into consideration in the stratified sampling, i.e., (1) the SDGs which the indicators measure, and (2) their titles' word count. Based on word count, the indicators under each SDG were divided into three categories, i.e., (1) less than 10 words, (2) between 10 and 20 words, (3) more than 20 words. This division worked fine for most of the SDGs, but for several SDGs, one of the categories contained only one example, so a representative split was impossible. For those SDGs, the indicators were divided into two categories, i.e., (1) less than 15 words and (2) more than 15 words. Finally, based on this categorization, the indicators in the validation and test set were sampled. For the multi-label classification of an indicator to SDGs, the test set examples were sampled in the same way, but only those indicators that belonged to multiple different classes (repeating indicators that monitored multiple different SDGs) were retained.

Test Dataset 2: EU SDG Indicator Set. As a second source of test indicators, the EU SDG indicator set consisting of 100 indicators was used (the version from 2023[4]). Indicator title and definition given in the Monitoring report on the progress towards the SDGs in an EU context, 2023 edition [17], were used. The single or multiple SDGs (for multi-purpose indicators), which the indicators monitor, were used as labels in the test tasks (1) MC-IND-SDG and (2) ML-IND-SDG.

Test Dataset 3: WDI SDG Indicator Set. WDI are the World Bank's collection of indicators that monitor different economies on global development. A set of 408 indicators, classified under an SDG and a target, was downloaded from the World Bank's data portal.[5] The indicators available under license other than CC-BY, as well as those that were not classified under a specific SDG and target in the data source, were excluded from the dataset, which resulted in a set of 368 indicators. The title and long description taken from the indicator metadata were used as a source of text. The indicators were used in the test tasks (1) MC-IND-SDG and (2) MC-IND-TRG.

Test Dataset 4: EU Local SDG Indicator Set. Test indicator set 4 consists of 72 indicators presented in the European Handbook for SDG Voluntary Local Reviews, 2022 Edition [10]. The title and definition of the indicators given in the document were prepared as described at the beginning of this section. The multiple SDG and target each indicator belongs to were used as labels, therefore, the two variations of this test set were used in (1) MC-IND-SDG, (2) MC-IND-TRG, and (3) ML-IND-SDG (only those indicators labeled with multiple SDGs).

Test Dataset 5: EU Regional SDG Indicator Set. The test indicator set 5 consists of the initial 83 European regional SDG indicators of the project REGIONS2030 [20]. Only the title of the indicators was available in the document, so this indicator set had one representation only. The SDG and target each indicator belonged to were used as labels, therefore, this test set was used in (1) MC-IND-SDG and (2) MC-IND-TRG.

Test Dataset 6: SDG Relatedness Dataset. For the multi-label classification of an SDG to multiple related SDGs (ML-SDG-SDG), the test set consisted of the titles of SDGs 1 to 16, labeled with the SDGs they link to. That information was extracted from the section ''Links to other SDGs'' of each SDG-specific Wikipedia article. All mentioned SDGs were considered as linked to the SDG the article referred to. SDG 17 was not included, as it was related to all other SDGs.

3) SENTENCE ENCODER BASELINE BENCHMARKING

Twelve state-of-the-art sentence encoders at the time of writing, belonging to four diverse categories, were compared on the validation tasks, of which the ones with the highest average accuracy over all validation task experimental configurations were selected for further fine-tuning. The first three were variations of the Universal Sentence Encoder (USE), i.e., the standard model based on the Transformer architecture [66] (USE-TRANSFORMER,[6] $v = 512$) and two multilingual models [67], of which the first based on a convolutional neural network (USE-MILTILINGUAL-CONVOLUTION,[7] $v = 512$) and the second on the Transformer architecture (USE-MILTILINGUAL-TRANSFORMER,[8] $v = 512$). The second group included the Sentence BERT (SBERT) models. The original models [58] were based on pre-trained BERT and RoBERTa models, fine-tuned through Siamese and triplet network architectures. The sentence encoder based on the BERT base architecture (SBERT-BERT-BASE,[9] $v = 768$) was used in this article. On the SBERT website,[10] the authors pointed to a new set of fine-tuned sentence encoders that had outperformed original SBERT encoders. Four that have the highest performance, as reported on that website,[11] were used, (1) fine-tuned MiniLM [68] model with 6 hidden layers (SBERT-MINILM-L6,[12] $v = 384$), (2) fine-tuned MiniLM model with 12 hidden layers (SBERT-MINILM-L12,[13] $v = 384$), (3) fine-tuned DistilRoBERTa [69] model (SBERT-DISTILROBERTA,[14] $v = 768$), and

[4]https://ec.europa.eu/eurostat/web/sdi/information-data

[5]https://databank.worldbank.org/source/sustainable-development-goals-(sdgs)

[6]https://tfhub.dev/google/universal-sentence-encoder-large/5

[7]https://tfhub.dev/google/universal-sentence-encoder-multilingual/3

[8]https://tfhub.dev/google/universal-sentence-encoder-multilingual-large/3

[9]https://huggingface.co/sentence-transformers/bert-base-nli-stsb-mean-tokens

[10]https://www.sbert.net/

[11]https://www.sbert.net/docs/pretrained_models.html

[12]https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[13]https://huggingface.co/sentence-transformers/all-MiniLM-L12-v2

[14]https://huggingface.co/sentence-transformers/all-distilroberta-v1

(4) fine-tuned MPNET [70] model (SBERT-MPNET-BASE,[15] $v = 768$). The third group of encoders included those based on the Simple contrastive sentence embedding framework (SimCSE) [59], which fine-tuned pre-trained BERT and RoBERTa models through contrastive learning in unsupervised and supervised settings. BERT base model fine-tuned in both settings was used (SIMCSE-UNSUP-BERT-BASE[16] and SIMCSE-SUP-BERT-BASE,[17] $v = 768$). The final group included the Sentence T5 (ST5) models [60] - fine-tuned T5 models through contrastive learning, optimized for sentence encoding. Two models were used (ST5-BASE[18] and ST5-LARGE,[19] $v = 768$). As new sentence encoders are constantly being proposed in the literature, the selected set is not exhaustive and will be expanded in our future work.

B. SENTENCE ENCODER FINE-TUNING, VALIDATION, AND TESTING

At each fine-tuning iteration, the examples in a mini-batch of size 64 were sampled using a stratified sampling strategy, which ensured a balanced distribution across the 17 SDGs in the mini-batch, i.e., 3-4 randomly selected examples by SDG. The SDGs that have either 3 or 4 examples were also randomly sampled. The same strategy was used for both fine-tuning tasks. The Adam optimization algorithm was used to fine-tune the network, with a learning rate $\eta = 2e-5$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e-8$. The margin α was set to 0.4 in the first fine-tuning task (FT-SDG) while 0.2 in the second (FT-TRG).

For each pre-trained sentence encoder selected for further fine-tuning, there was one baseline checkpoint, and 20 checkpoints fine-tuned on each of the four fine-tuning datasets and five random seeds. Early stopping was used as a regularization strategy. Validation tasks were used to evaluate the checkpoint parameters after each 5 consecutive fine-tuning iterations, and the average performance on all validation experimental settings (described in Table 1) was calculated. The model was fine-tuned on each task for 20 iterations and the checkpoint with the highest average validation accuracy was selected. The process is illustrated in Figure 8.

During the validation and testing, we were using the fine-tuned sentence encoders as feature extractors and combined them with a non-parametric learning algorithm k Nearest Neighbors (kNN). Within tasks MC-IND-SDG, ML-IND-SDG, and ML-SDG-SDG, four different experimental settings were defined, differing in the approximate number of training examples by class (approximate K). In task MC-IND-TRG, two different experimental settings were defined, again differing in the approximate K. Six validation experimental settings in total were defined and 14 test experimental settings. Table 1 gives the details on each.

The training sets $\mathbf{D}_{TR}$ of the validation and test tasks were created from those used in fine-tuning. In tasks MC-IND-SDG, ML-IND-SDG, and ML-SDG-SDG, combinations of examples from the fine-tuning sets were created. If S_l, $l = \{1, \ldots, 17\}$, is the set of all targets under SDG l, with cardinality $n(S_l)$, then the approximate number of examples for SDG l in the combined training sets would be $K \in \{14 + n(S_l) * 6,\ 14 + n(S_l) * 17,\ 22 + n(S_l) * 6,\ 22 + n(S_l) * 17\}$. In ML-SDG-SDG, while the ''training'' examples were initially sampled in the same manner, the actual training set used by the kNN classifier consisted of the centroids of the ''training'' examples by class, i.e., one training example by class. These centroids, one per class, were then used to classify the titles of the first 16 SDGs in 16 classes, excluding SDG 17.

During validation and testing, the parameters of the sentence encoder were fixed and it was only used to output embeddings of the training and validation $\mathbf{D}_{VL}$ (test $\mathbf{D}_{TS}$) set examples. The training set embeddings and labels were used by the kNN classifier to predict the validation (test) examples classes based on a weighted sum of their k nearest neighbors' labels (subset of k examples represented as $\mathbf{D}_{TR}^{(i)} \subset \mathbf{D}_{TR}$), as given with Eq. 3. In MC-IND-SDG and MC-IND-TRG, the performance metric was the accuracy, while in ML-IND-SDG and ML-SDG-SDG the Normalized Discounted Cumulative Gain (NDCG) where the predictions given with Eq. 3 were ranked in descending order, and the five predicted classes with the highest score were compared to the actual labels.

$$\hat{\mathbf{y}}^{(i)} = \sum_{j=1}^{k} [1 - d(\mathbf{z}^{(i)}, \mathbf{z}^{(j)})]\mathbf{y}^{(j)}, (\mathbf{z}^{(j)}, \mathbf{y}^{(j)}) \in \mathbf{D}_{TR}^{(i)} \quad (3)$$

C. POST-HOC ANALYSIS

To better understand the factors that influenced the test results, all factors of interest were represented as input features to a meta-model, i.e., linear regression, which was then trained to predict the accuracy or NDCG by SDG that was actually achieved by our fine-tuned sentence encoders and kNN classifier on the test datasets from (1) the Global indicator framework of the 2030 Agenda (task MC-IND-SDG) and (2) SDG relatedness (task ML-SDG-SDG). The training set on which the linear regression was trained consisted of all different fine-tuning and test experimental configurations. The same process was repeated twice, once for MC-IND-SDG and ML-SDG-SDG, resulting in a total of 5,440 labeled examples for the first and 2,560 for the second. The input features are described in Table 2. Different linear regression algorithms were compared, i.e., regular linear regression without regularization, Ridge regression (varying regularization hyperparameter α), Lasso regression (varying α), and ElasticNet (varying α and r), where $\alpha, r \in \{1e-6, 5e-6, 1e-5, 5e-5, 1e-4, 5e-4, 1e-3, 5e-3, 1e-2, 5e-2, 0.1, 0.5, 1.0\}$. The contribution of each feature to the prediction made by the linear regression model

[15] https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[16] https://huggingface.co/princeton-nlp/unsup-simcse-bert-base-uncased
[17] https://huggingface.co/princeton-nlp/sup-simcse-bert-base-uncased
[18] https://tfhub.dev/google/sentence-t5/st5-base/1
[19] https://tfhub.dev/google/sentence-t5/st5-large/1

TABLE 1. Description of the validation and test tasks, together with all experimental settings. Note: In the task ML-SDG-SDG, one centroid by SDG is calculated from the training examples (6th column).

		D_{TR} settings				D_{VL} / D_{TS} settings				kNN settings	
Phase	Task	Text	Label	N	Approx. K	Text	Label	N	K	k	Metric
Validation	MC-IND-SDG	Wiki. excrpt.	SDG	17	$14+n(S_l)*6$ $14+n(S_l)*17$ $22+n(S_l)*6$ $22+n(S_l)*17$	Indicator title	SDG	17	Section IV-A2	1, 3	Accuracy@1
Validation	MC-IND-TRG	Wiki. excrpt.	Target	169	6 17	Indicator title	Target	169	Section IV-A2	1, 3	Accuracy@1
Test	MC-IND-SDG	Wiki. excrpt.	SDG	17	$14+n(S_l)*6$ $14+n(S_l)*17$ $22+n(S_l)*6$ $22+n(S_l)*17$	Indicator title/definition	SDG	17	Section IV-A2	3	Accuracy@1
Test	MC-IND-TRG	Wiki. excrpt.	Target	169	6 17	Indicator title/definition	Target	169	Section IV-A2	20	Accuracy@5
Test (zero-shot learning)	ML-IND-SDG	Wiki. excrpt.	SDG	17	$14+n(S_l)*6$ $14+n(S_l)*17$ $22+n(S_l)*6$ $22+n(S_l)*17$	Indicator title/definition	SDGs	17	Section IV-A2	20	NDCG@5
Test (zero-shot learning)	ML-SDG-SDG	Wiki. excrpt.	SDG	16	$14+n(S_l)*6$ $14+n(S_l)*17$ $22+n(S_l)*6$ $22+n(S_l)*17$	SDG title	Linked SDGs	16	1	16	NDCG@5

for each individual training example was then calculated with the SHAP method (see Figure 9).

V. RESULTS

A. PRELIMINARY ANALYSIS

For each indicator test set and two variations of the indicator descriptions (title or concatenated title with definition excerpt), Table 3 first gives the average number of words in the specific indicator set and then the average similarity of all indicators to their most similar fine-tuning example. The results indicate that this average similarity is rather low for most indicator sets, i.e., between 0.20 and 0.25. It is the highest for the test indicator set sampled from the Global indicator framework of the 2030 Agenda, i.e., 0.35. The results further show that this average similarity is slightly higher when the indicators are represented by their titles (which are quite short in most cases), compared to the average similarity when they are represented through their concatenated title with definition excerpt (which are longer – around 30 words). Therefore, it can be concluded that the test examples do not show much similarity to the fine-tuning examples, i.e., test examples do not appear among the fine-tuning examples.

B. SENTENCE ENCODER BASELINE BENCHMARKING

The validation accuracy of the twelve pre-trained sentence encoders averaged over the six validation experimental settings and two values of kNN $k \in \{1, 3\}$, is summarized in Table 4. The average accuracy varies between the different categories of sentence encoders and within the categories themselves. Two sentence encoders achieving the highest average accuracy were selected for further domain-specific fine-tuning, i.e., SBERT-MINILM-L6 and SBERT-MINILM-L12. In general, the encoders from the SBERT category, i.e., SBERT-MINILM-L6, SBERT-MINILM-L12, SBERT-MPNET-BASE, SBERT-DISTILROBERTA, (with one exception – SBERT-BERT-BASE presented in a separate research article [58], prior to the remaining four encoders), have the highest average accuracy, followed by the encoders from the USE and SimCSE categories, i.e., the USE-TRANSFORMER and SIMCSE-SUP-BERT-BASE. The ST5 encoders are among those with the lowest average accuracy, but it is comparable to that of some of the encoders from the USE and SimCSE categories, as well as to the SBERT-BERT-BASE encoder. We believe that such validation accuracy may be a result of the encoders' pre-training tasks/datasets and their similarity to the main tasks solved in this article. Solving the main tasks requires sentence encoders that capture the differences between the topics covered by the SDGs/targets in a common vector space, i.e., a very diverse set of topics. The four best-performing encoders have been pre-trained on a large and diverse set of tasks and datasets (for more details see the SBERT website[20]), which has probably enabled them to better capture the differences between the SDGs and targets. It should be noted that all the aforementioned conclusions apply solely to the tasks solved in this article and should not be generalized.

C. SENTENCE ENCODER TEST RESULTS AFTER FINE-TUNING

The test results of the two selected sentence encoders in the 14 test experimental settings, (1) MC-IND-SDG (4 settings), (2) MC-IND-TRG (2 settings), (3) ML-IND-SDG (4 settings), and (4) ML-SDG-SDG (4 settings) are presented in Tables 5, 6, 7, and 8, appropriately. The tables first give the highest baseline test results for any of the two

[20]https://www.sbert.net/

TABLE 2. Description of the input features of the linear regression. "FTn" refers to the examples from the fine-tuning set of the fine-tuning task experimental setting, while "kNN" refers to the training set of the test task experimental setting used by the kNN classifier.

Input Feature	Description
[Section] Example Num [FTn | kNN]	Number of examples in the [fine-tuning set | kNN training set] of the [fine-tuning task FT-SDG | test task experimental setting], by Wikipedia article section and SDG. This applies to all sections except the "targets" section, which is addressed separately in the two input features that follow.
Target Mean Example Num [FTn | kNN]	Mean of the total number of examples by target in the [fine-tuning | training] set of [fine-tuning | test] task experimental setting (extracted from the SDG-specific Wikipedia articles' "targets" section), grouped by SDG.
Target Std Example Num [FTn | kNN]	Standard deviation of the total number of examples by target in the [fine-tuning | training] set of [fine-tuning | test] task experimental setting (extracted from the SDG-specific Wikipedia articles' "targets" section), grouped by SDG.
[Section] Mean Word Num [FTn | kNN]	Mean number of words in the [fine-tuning | training] examples text of [fine-tuning | test] task experimental setting by section and SDG.
[Section] Std Word Num [FTn | kNN]	Standard deviation of the number of words in the [fine-tuning | training] examples text of [fine-tuning | test] task experimental setting by section and SDG.
Task FTn	Fine-tuning task. Value equals 1 for datasets of task FT-SDG experimental settings, and value equals 2 for datasets of task FT-TRG experimental settings.
Num Neighbors kNN	Number of nearest neighbors used by the kNN classifier in the test task experimental setting.

TABLE 3. Average word count and similarity to a fine-tuning example (max value) of the different indicator sets and indicator representations, (1) through their title and (2) through their title concatenated with definition excerpt.

Test set	2030 Agenda Indicators	EU SDG Indicators		WDI SDG Indicators		EU Local SDG Indicators		EU Regional SDG Indicators
Indicator representation	Title	Title	Title + Definition	Title	Title + Definition	Title	Title + Definition	Title
Avg words	15.86 (±8.98)	6.01 (±2.89)	31.28 (±10.85)	9.32 (±3.48)	31.28 (±10.85)	4.21 (±2.00)	31.79 (±11.33)	5.48 (±3.07)
Avg sim.	0.35 (±0.16)	0.25 (±0.11)	0.23 (±0.09)	0.26 (±0.11)	0.23 (±0.09)	0.25 (±0.09)	0.20 (±0.06)	0.26 (±0.11)

TABLE 4. Average validation accuracy of the twelve pre-trained sentence encoders on the six validation experimental settings and two values of nearest neighbors $k \in \{1, 3\}$. The two selected for further fine-tuning are given in bold.

Sentence Encoder	Average Validation Accuracy
USE-TRANSFORMER	0.63
USE-MULTILINGUAL-CONVOLUTION	0.61
USE-MULTILINGUAL-TRANSFORMER	0.59
ST5-BASE	0.59
ST5-LARGE	0.58
SBERT-BERT-BASE	0.60
SBERT-MINILM-L6	**0.73**
SBERT-MINILM-L12	**0.69**
SBERT-DISTILROBERTA	0.66
SBERT-MPNET-BASE	0.68
SIMCSE-UNSUP-BERT-BASE	0.61
SIMCSE-SUP-BERT-BASE	0.63

encoders (prior to their fine-tuning) by test task and test dataset combination. Then, they give the highest average result over five random seeds after the encoders fine-tuning with all fine-tuning datasets, again by test task and test dataset combination. These average results are accompanied by the best test results of that same fine-tuning/test configuration but with one specific random seed. This result is given along with the improvement over the highest baseline result for the same test task and dataset combination.

In the first test task, MC-IND-SDG, the highest kNN classifier accuracy@1 (k=3) on all datasets is above 80% or very close to it (in the case of EU Regional SDG test set), as given in Table 5. The accuracy@1 is 90% for the test set sampled from the Global indicator framework from the 2030 Agenda. For the test sets having two indicator description variations (title and concatenated title with definition excerpt), it is visible that the classifier accuracy is higher when the indicators are represented through their title, both before and after the encoder fine-tuning. However, the fine-tuning makes the classifier accuracy less sensitive to changes in the indicator description length. A decrease in the accuracy of the baseline classifiers due to an increase in the length of the indicator description is present for all test sets, ranging from more than 15% for the EU SDG and EU Local SDG test sets to 3% for WDI SDG. However, after the sentence encoder fine-tuning, for the EU SDG and EU Local SDG test sets, the accuracy decreases by 5% and 3% appropriately, which is much less than its decrease with the baseline classifiers. For the WDI SDG test sets, there is an increase of 1%.

In the second test task MC-IND-TRG, which requires distinguishing between 169 highly interconnected targets, the kNN classifier accuracy@5 (k=20) is around 80% or above in most cases, as given in Table 6. Measuring the accuracy@1 was a rather strict test criterion, considering the high level of inter-relatedness between the targets and our aim to find even the non-obvious associations of the indicators with the targets. In that sense, we expected to have more than one associated target with the test indicators. As in the first task,

TABLE 5. Highest baseline result in test task MC-IND-SDG and highest average result over five random seeds after encoders fine-tuning by test set. The best test results of that same fine-tuning/test configuration are given, along with the improvement over the highest baseline result for that test set.

Configuration			Accuracy@1 (kNN k=3)	
Test Dataset	**Indicators Representation**	**Fine-Tuning**	**Highest Average over 5 Random Seeds (St.Dev.)**	**Best Checkpoint (Improvement Over Baseline)**
2030 Agenda Indicators	Title	No (Baseline) Yes	/ 0.89 (0.006)	0.84 0.90 (+0.061)
EU SDG Indicators	Title	No (Baseline) Yes	/ 0.86 (0.009)	0.82 0.87 (+0.050)
EU SDG Indicators	Title + Definition	No (Baseline) Yes	/ 0.82 (0.005)	0.65 0.82 (+0.170)
WDI SDG Indicators	Title	No (Baseline) Yes	/ 0.82 (0.007)	0.76 0.83 (+0.068)
WDI SDG Indicators	Title + Definition	No (Baseline) Yes	/ 0.82 (0.016)	0.73 0.84 (+0.106)
EU Local SDG Indicators	Title	No (Baseline) Yes	/ 0.87 (0.015)	0.83 0.89 (+0.056)
EU Local SDG Indicators	Title + Definition	No (Baseline) Yes	/ 0.85 (0.010)	0.68 0.86 (+0.181)
EU Regional SDG Indicators	Title	No (Baseline) Yes	/ 0.76 (0.015)	0.73 0.77 (+0.036)

TABLE 6. Highest baseline result in test task MC-IND-TRG and highest average result over five random seeds after encoders fine-tuning by test set. The best test results of that same fine-tuning/test configuration is given, along with the improvement over the highest baseline result for that test set.

Configuration			Accuracy@5 (kNN k=20)	
Test Dataset	**Indicators Representation**	**Fine-Tuning**	**Highest Average over 5 Random Seeds (St.Dev.)**	**Best Checkpoint (Improvement Over Baseline)**
2030 Agenda Indicators	Title	No (Baseline) Yes	/ 0.88 (0.009)	0.84 0.89 (+0.050)
WDI SDG Indicators	Title	No (Baseline) Yes	/ 0.78 (0.009)	0.73 0.79 (+0.057)
WDI SDG Indicators	Title + Definition	No (Baseline) Yes	/ 0.77 (0.007)	0.73 0.77 (+0.041)
EU Local SDG Indicators	Title	No (Baseline) Yes	/ 0.79 (0.006)	0.78 0.79 (+0.014)
EU Local SDG Indicators	Title + Definition	No (Baseline) Yes	/ 0.78 (0.014)	0.63 0.79 (+0.167)
EU Regional SDG Indicators	Title	No (Baseline) Yes	/ 0.83 (0.009)	0.78 0.84 (+0.060)

TABLE 7. Highest baseline result in test task ML-IND-SDG and highest average result over five random seeds after encoders fine-tuning by test set. The best test results of that same fine-tuning/test configuration are given, along with the improvement over the highest baseline result for that test set.

Configuration			NDCG@5 (kNN k=20)	
Test Dataset	**Indicators Representation**	**Fine-Tuning**	**Highest Average over 5 Random Seeds (St.Dev.)**	**Best Checkpoint (Improvement Over Baseline)**
2030 Agenda Indicators	Title	No (Baseline) Yes	/ 1.00 (0.000)	0.97 1.00 (+0.026)
EU SDG Indicators	Title	No (Baseline) Yes	/ 0.90 (0.007)	0.90 0.91 (+0.011)
EU SDG Indicators	Title + Definition	No (Baseline) Yes	/ 0.86 (0.001)	0.79 0.86 (+0.073)
EU Local SDG Indicators	Title	No (Baseline) Yes	/ 0.70 (0.007)	0.70 0.71 (+0.012)
EU Local SDG Indicators	Title + Definition	No (Baseline) Yes	/ 0.70 (0.007)	0.64 0.71 (+0.067)

the fine-tuning makes the classifier less sensitive to changes in the description length, particularly for the EU Local SDG test set.

The third task, ML-IND-SDG, appears to be the most challenging for the EU Local SDG test set compared to the rest which have a relatively high NDCG@5 score both before and after the fine-tuning. Similarly to the previous two test tasks, there is performance degradation as the length of the indicator description increases for test sets EU SDG and EU Local SDG. However, such performance degradation is either not present or much lower with a fine-tuned encoder, as given in Table 7.

The improvement of the NDCG@5 value in the test task ML-SDG-SDG is around 7.9% after fine-tuning, as given

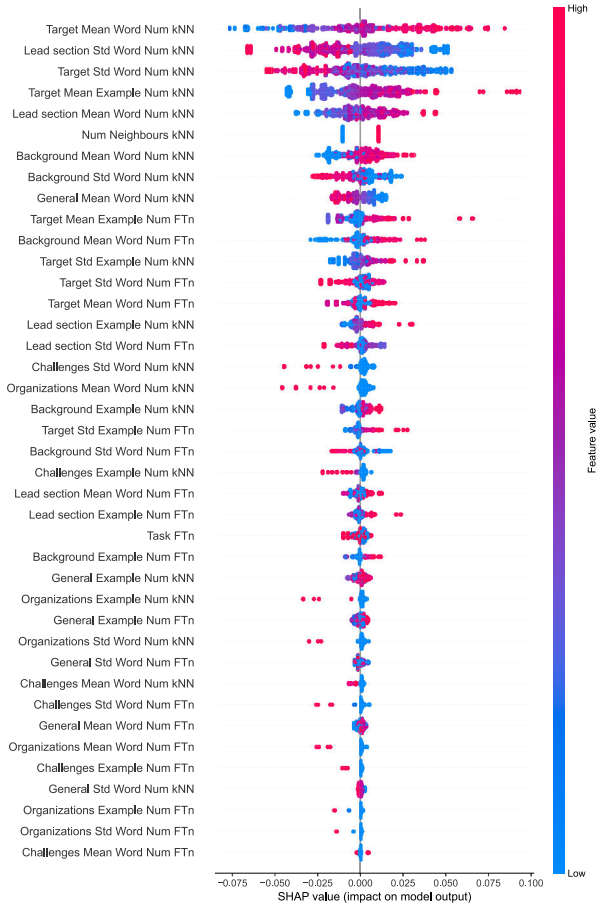

FIGURE 15. SHAP value distribution by input feature (test task MC-IND-SDG).

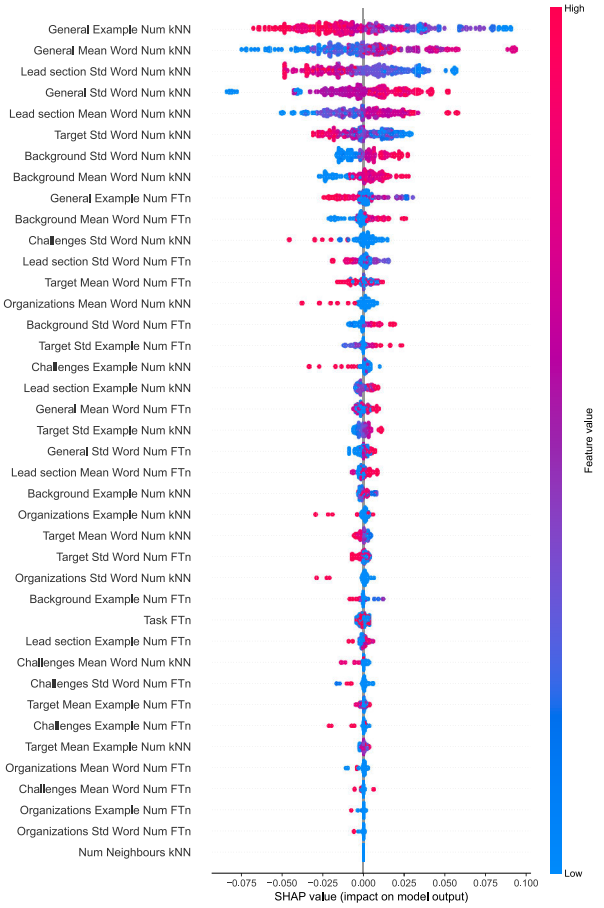

FIGURE 16. SHAP value distribution by input feature (test task ML-SDG-SDG).

in Table 8. The improvements in ML-IND-SDG and ML-SDG-SDG indicate that although information on the mutual SDG relations was not explicitly provided during fine-tuning, it was still learned from the text in the fine-tuning datasets.

D. POST-HOC ANALYSIS

In task MC-IND-SDG, the lowest mean squared error (MSE) of 0.0073 was achieved with Ridge regression ($\alpha = 1e-6$), as in task ML-SDG-SDG, where the lowest MSE was 0.0059. The distribution of SHAP values by feature in task MC-IND-SDG is illustrated in Figure 15, while in task ML-SDG-SDG in Figure 16. The suffix FTn or kNN simply indicates that the feature refers to the examples in the fine-tuning set of the fine-tuning tasks or to the examples in the kNN training set of the test tasks. In both tasks, the structure of the kNN training set had a larger influence. In the task MC-IND-SDG, the large number of words in the text excerpts extracted from the section devoted to the targets, the ''background'' section, and the lead section had the largest positive influence, as well as the mean number of examples by target and the low standard deviation of the number of words in the lead section excerpts. During fine-tuning, the large mean number of examples for the SDG targets in the fine-tuning set had the greatest positive influence, as well as the large number of words in the excerpts from the ''background'' section. Contrary to our expectations, the large number of excerpts from the general Wikipedia article or their large number of words did not have a positive influence on the accuracy by SDG. The presence of excerpts extracted from sections such as ''organizations'' or ''challenges'' did not affect accuracy positively, and this observation also applies to the task ML-SDG-SDG. However, in this task the large number of examples from the general article in both the fine-tuning and kNN training set had negative influence, while the large mean number of words in the excerpts from the ''background'' section in both the fine-tuning and kNN training set a positive one. The large mean number of words in the excerpts from the lead section, which are part of the kNN training set, had positive influence as well.

VI. DISCUSSION

This article proposed a model-agnostic method for finding associations between sustainable development indicators used at the national or subnational level of governance with the SDGs and targets from the UN 2030 Agenda. By relying on textual descriptions of SDGs, targets, and indicators,

TABLE 8. Highest baseline result in test task ML-SDG-SDG and highest average result over five random seeds after encoders fine-tuning by test set. The best test results of that same fine-tuning/test configuration are given, along with the improvement over the highest baseline result for that test set.

Configuration			NDCG@5 (kNN k=16)	
Test Dataset	**Indicators Representation**	**Fine-Tuning**	**Highest Average over 5 Random Seeds (St.Dev.)**	**Best Checkpoint (Improvement Over Baseline)**
SDG Relatedness	Title	No (Baseline)	/	0.67
		Yes	0.75 (0.005)	0.75 (+0.079)

the dependence on indicator statistical data quantity and quality was removed. The article proposed a new formulation of the problem in the ML domain – text classification and divided the problem into two main and two auxiliary tasks to make the evaluation easier. The proposed method used (1) short text to describe the SDGs, targets, and indicators, (2) general-purpose pre-trained sentence encoders to represent those descriptions in a common vector space, (3) contrastive representation learning and domain-specific datasets to fine-tune their parameters, and (4) kNN classifier in the experimental setting, to ''associate'' indicators with SDGs and targets by comparing their embeddings outputted by the shared encoder with a distance metric. The method was evaluated on five real-world indicator sets used at different levels of governance. This section analyzes the results in the context of the three research questions and summarizes the limitations of the method.

Regarding the potential of textual data and general-purpose pre-trained sentence encoders in solving the main and auxiliary tasks (RQ1), the test results showed that text is a promising type of data for solving the tasks. The validation results showed that certain sentence encoders are particularly well suited to solving the tasks even before the domain-specific fine-tuning, i.e., they have already captured useful knowledge to solve the tasks even during their pre-training on general NLP tasks. This particularly refers to the SBERT category of sentence encoders. Furthermore, in all of the test configurations, their performance was further improved by fine-tuning with domain-specific datasets, even though the datasets were quite limited in size. That suggests the possibility of even greater performance improvements with larger fine-tuning datasets and better selected examples based on the insights from Section V-D.

In terms of performance improvements through domain-specific fine-tuning of the pre-trained encoders (RQ2), the presented results by test task category and dataset showed that the domain-specific fine-tuning improved the baseline results in all test task categories for all test datasets. Although the fine-tuning datasets were quite limited in size, their use with a contrastive representation learning method still improved the baseline results. The best-performing fine-tuning and kNN configuration was specific to the test indicator dataset, which can be attributed to the different purposes of the datasets, resulting in different writing styles, as well as length of the titles and definitions of their indicators. A more diverse validation set sampled from different indicator sets may help in selecting a model that better fits diverse indicator sets, a possible research direction for future work. Furthermore, fine-tuning with other contrastive representation learning methods is possible as well and should be considered in future work.

The results also showed that the textual data used in describing the SDGs, targets, and indicators has a significant influence on the test results when using the proposed method (RQ3). This especially applies to the structure of the fine-tuning and kNN training set, as shown by the SHAP analysis on the Global indicator framework test set in Section V-D. Those results showed that the differences between the topics covered by the SDGs were better captured in the excerpts extracted from the abstract and ''background'' section of the SDG-specific Wikipedia articles, but not so well in the excerpts extracted from sections such as ''challenges'' or ''organizations''. On the contrary, the sections of the articles devoted to the targets were more useful in associating indicators with the SDGs and targets, compared to associating SDGs to their related SDGs. The length of the indicator descriptions also influenced the results, but that influence was less pronounced when using fine-tuned sentence encoders. When using fine-tuned encoders, there was either no performance degradation or it was much smaller compared to the case of encoders which were not fine-tuned. Therefore, it can be concluded that the benefit of domain-specific fine-tuning of the sentence encoders was twofold, i.e., (1) it improved the predictive performance of the evaluated classifiers over the baseline and (2) it resulted in classifiers that were less sensitive to changes in indicator description length.

Most of the limitations of this study were mainly a result of the experimental choices we had to make to keep the study within a reasonable scope, given the many challenges that had to be addressed. These challenges included (1) a large number of SDGs and targets with complex mutual interlinks which the method had to learn to distinguish, (2) the lack of ready-to-use fine-tuning datasets for the problem, i.e., the need to create them ourselves, and, finally, (3) the non-trivial representative text extraction process from the Wikipedia articles. Therefore, the limitations and directions for overcoming them in future work are given in Table 9. Table 9 shows that the insights from this study have opened several promising research directions to improve the presented results. Those research directions range from benchmarking more advanced ML models on the problem, such as more recently published sentence encoders or LLMs, to creation of larger and more diverse text datasets for model fine-tuning and evaluation, definition of additional fine-tuning and evaluation tasks to be combined with those

TABLE 9. Limitations of the study and directions for future work.

Limitation	Future Work Direction
Benchmarking sentence encoders from categories that were state-of-the-art at the time of writing, potentially missing newly proposed encoders.	Benchmarking an extensive set of newly proposed state-of-the-art encoders which are highly ranked in general benchmarks like the Massive Text Embedding Benchmark (MTEB) [71]. Benchmarking state-of-the-art open-source LLMs on the tasks, after proper adjustment of the tasks to LLM context.
Using a single source of text data for the fine-tuning datasets, i.e., Wikipedia, which may be insufficient to attribute the different writing styles and title/description expressiveness inherent to different real-world indicator sets.	Creating fine-tuning datasets with textual descriptions of the SDGs and targets that come from diverse publicly available sources.
Hypothesizing a positive influence (1) of the sections from the general Wikipedia article and (2) of sections such as "organizations" and "challenges" (used to compensate for the shorter general sections).	The classification of indicators to SDGs generally requires descriptions of the SDGs that are much more similar to those of their targets.
Experimentation with a single contrastive representation learning method, i.e., the triplet network architecture.	Evaluating additional contrastive representation learning methods and adjusting their fine-tuning process to the findings outlined in Sections V and VI.
No fine-tuning task that considers the mutual links between the SDGs/targets.	Obtaining data that captures certain links between the SDGs/targets as part of an additional fine-tuning task, under the assumption that such a task would improve the test results.

TABLE 10. Titles of the 17 SDGs [1].

		Targets	
Goal	Title	Outcome	Means of Implementation
1	End poverty in all its forms everywhere	1.1 - 1.5	1.a, 1.b
2	End hunger, achieve food security and improved nutrition and promote sustainable agriculture	2.1 - 2.5	2.a - 2.c
3	Ensure healthy lives and promote well-being for all at all ages	3.1 - 3.9	3.a - 3.d
4	Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all	4.1 - 4.7	4.a - 4.c
5	Achieve gender equality and empower all women and girls	5.1 - 5.6	5.a - 5.c
6	Ensure availability and sustainable management of water and sanitation for all	6.1 - 6.6	6.a, 6.b
7	Ensure access to affordable, reliable, sustainable and modern energy for all	7.1 - 7.3	7.a, 7.b
8	Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all	8.1 - 8.10	8.a, 8.b
9	Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation	9.1 - 9.5	9.a - 9.c
10	Reduce inequality within and among countries	10.1 - 10.7	10.a - 10.c
11	Make cities and human settlements inclusive, safe, resilient and sustainable	11.1 - 11.7	11.a - 11.c
12	Ensure sustainable consumption and production patterns	12.1 - 12.8	12.a - 12.c
13	Take urgent action to combat climate change and its impacts	13.1 - 13.3	13.a, 13.b
14	Conserve and sustainably use the oceans, seas and marine resources for sustainable development	14.1 - 14.7	14.a - 14.c
15	Protect, restore and promote sustainable use of terrestrial ecosystems, sustainably manage forests, combat desertification, and halt and reverse land degradation and halt biodiversity loss	15.1 - 15.9	15.a - 15.c
16	Promote peaceful and inclusive societies for sustainable development, provide access to justice for all and build effective, accountable and inclusive institutions at all levels	16.1 - 16.10	16.a, 16.b
17	Strengthen the means of implementation and revitalize the global partnership for sustainable development	17.1 - 17.19	

proposed in this paper, as well as use of more advanced model fine-tuning methods.

In addition to scientific contribution, the proposed method has practical significance as well. The results achieved on several real-world indicator sets showed that the method could be useful in complementing and facilitating the non-trivial process of manual mapping of indicator sets used at different levels of governance to multiple SDGs and targets with which those indicators are associated, which was the main goal of this work. The method can provide human experts with data-driven suggestions for such associations, which is particularly useful when associating indicators with the 169 highly interlinked targets from the Agenda. In that context, one particularly relevant use case that could benefit from the proposed method is the association of locally relevant indicator sets used in different geographic regions with the SDGs and targets, in order to facilitate the monitoring of the effects that local policies have on the Agenda.

VII. CONCLUSION

Monitoring progress toward achieving the 2030 Agenda for Sustainable Development of the United Nations is of the utmost priority in the current decade. Sustainable development policies/actions are taken at different levels of governance, and monitoring of their effects on the Sustainable Development Goals (SDGs) and targets is essential due to SDGs/targets complex and not always obvious interactions. Indicators relevant to a specific context (e.g., level of governance, geographic region) are commonly used for this purpose, but the associations of such indicators with the SDGs may not always be easy to determine. This article presents a model-agnostic framework (Embed4SD) to associate indicators with SDGs and targets by comparing their textual descriptions. In that way, it removes the dependence on the variable indicator statistical data quantity/quality and facilitates human experts' manual mapping process with data-driven insights. Our experiments include a comprehensive

TABLE 11. 18 English-language Wikipedia articles and their revision IDs used as data source for the fine-tuning datasets.

No.	Wikipedia article URL	Revision ID
1	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_1	1058009356
2	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_2	1060913868
3	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_3	1059877068
4	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_4	1058007755
5	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_5	1061200438
6	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_6	1061708983
7	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_7	1060845341
8	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_8	1062437298
9	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_9	1059594243
10	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_10	1058008474
11	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_11	1060946438
12	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_12	1060940446
13	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_13	1058011032
14	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_14	1061877687
15	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_15	1060901217
16	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_16	1058009669
17	https://en.wikipedia.org/wiki/Sustainable_Development_Goal_17	1056725992
18	https://en.wikipedia.org/wiki/Sustainable_Development_Goals	1060949206

TABLE 12. Baseline and fine-tuned configurations achieving the highest result in test task MC-IND-SDG (Table 5).

Test Dataset		Config.		Fine-Tuning Config.		Test Config.
Dataset	**Indicator Rep.**	**Fine-Tuning**	**Encoder**	**Task**	**K**	**K**
2030 Agenda Indicators	Title	No (Baseline)	SBERT-MINILM-L12	/	/	$14 + n(S_l) * 6$
		Yes	SBERT-MINILM-L12	FT-SDG	22	$22 + n(S_l) * 17$
EU SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L12	/	/	$22 + n(S_l) * 6$
		Yes	SBERT-MINILM-L12	FT-SDG	22	$14 + n(S_l) * 6$
EU SDG Indicators	Title + Definition	No (Baseline)	SBERT-MINILM-L12	/	/	$22 + n(S_l) * 6$
		Yes	SBERT-MINILM-L6	FT-SDG	14	$22 + n(S_l) * 6$
WDI SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L6	/	/	$22 + n(S_l) * 6$
		Yes	SBERT-MINILM-L12	FT-TRG	17	$14 + n(S_l) * 6$
WDI SDG Indicators	Title + Definition	No (Baseline)	SBERT-MINILM-L12	/	/	$14 + n(S_l) * 6$
		Yes	SBERT-MINILM-L6	FT-SDG	14	$14 + n(S_l) * 6$
EU Local SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L6	/	/	$22 + n(S_l) * 6$
		Yes	SBERT-MINILM-L12	FT-TRG	6	$14 + n(S_l) * 17$
EU Local SDG Indicators	Title + Definition	No (Baseline)	SBERT-MINILM-L12	/	/	$14 + n(S_l) * 17$
		Yes	SBERT-MINILM-L12	FT-SDG	14	$22 + n(S_l) * 17$
EU Regional SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L6	/	/	$14 + n(S_l) * 6$
		Yes	SBERT-MINILM-L12	FT-SDG	22	$22 + n(S_l) * 17$

TABLE 13. Baseline and fine-tuned configurations achieving the highest result in test task MC-IND-TRG (Table 6).

Test Dataset		Config.		Fine-Tuning Config.		Test Config.
Dataset	**Indicator Rep.**	**Fine-Tuning**	**Encoder**	**Task**	**K**	**K**
2030 Agenda Indicators	Title	No (Baseline)	SBERT-MINILM-L6	/	/	6
		Yes	SBERT-MINILM-L12	FT-TRG	6	6
WDI SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L6	/	/	6
		Yes	SBERT-MINILM-L12	FT-TRG	17	17
WDI SDG Indicators	Title + Definition	No (Baseline)	SBERT-MINILM-L6	/	/	6
		Yes	SBERT-MINILM-L6	FT-TRG	6	6
EU Local SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L12	/	/	17
		Yes	SBERT-MINILM-L12	FT-SDG	14	6
EU Local SDG Indicators	Title + Definition	No (Baseline)	SBERT-MINILM-L6	/	/	6
		Yes	SBERT-MINILM-L6	FT-SDG	14	17
EU Regional SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L6	/	/	6
		Yes	SBERT-MINILM-L6	FT-SDG	14	6

domain-specific benchmarking of 12 sentence encoders, fine-tuning of the best ones on a newly created dataset, evaluation with five real-world indicator sets consisting of around 800 indicators in total, and measuring the influence of 40 factors on the results using explainable artificial intelligence (xAI). The results show that certain sentence encoders are better suited to solving the task than others, potentially due to the diversity of their pre-training datasets. Furthermore, not only does fine-tuning improve predictive performance over baselines, it also reduces the sensitivity to changes in indicator description length, i.e., while the performance drops even by up to 17% for baseline models as length increases, it remains comparable for fine-tuned models. We believe that Embed4SD makes a step in filling the current gap of comprehensive benchmarking of AI models on the problem and opens a promising research direction, that is, solving the problem through textual data.

TABLE 14. Baseline and fine-tuned configurations achieving the highest result in test task ML-IND-SDG (Table 7).

Test Dataset		Config.		Fine-Tuning Config.		Test Config.
Dataset	**Indicator Rep.**	**Fine-Tuning**	**Encoder**	**Task**	**K**	**K**
2030 Agenda Indicators	Title	No (Baseline)	SBERT-MINILM-L12	/	/	$14 + n(S_I) * 6$
		Yes	SBERT-MINILM-L12	FT-SDG	22	$14 + n(S_I) * 6$
EU SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L12	/	/	$22 + n(S_I) * 6$
		Yes	SBERT-MINILM-L12	FT-SDG	14	$14 + n(S_I) * 6$
EU SDG Indicators	Title + Definition	No (Baseline)	SBERT-MINILM-L6	/	/	$14 + n(S_I) * 6$
		Yes	SBERT-MINILM-L12	FT-SDG	14	$14 + n(S_I) * 6$
EU Local SDG Indicators	Title	No (Baseline)	SBERT-MINILM-L12	/	/	$14 + n(S_I) * 17$
		Yes	SBERT-MINILM-L12	FT-SDG	14	$14 + n(S_I) * 17$
EU Local SDG Indicators	Title + Definition	No (Baseline)	SBERT-MINILM-L12	/	/	$22 + n(S_I) * 6$
		Yes	SBERT-MINILM-L6	FT-SDG	14	$22 + n(S_I) * 17$

TABLE 15. Baseline and fine-tuned configurations achieving the highest result in test task ML-SDG-SDG (Table 8).

Dataset		Config.		Fine-Tuning Config.		Test Config.
Test Dataset	**Indicator Rep.**	**Fine-Tuning**	**Encoder**	**Task**	**K**	**K**
SDG Relatedness	Title	No (Baseline)	SBERT-MINILM-L12	/	/	$14 + n(S_I) * 6$
		Yes	SBERT-MINILM-L12	FT-SDG	22	$14 + n(S_I) * 6$

APPENDIX A
SUSTAINABLE DEVELOPMENT GOALS

Each of the first 16 SDGs is devoted to a specific area, while SDG 17 is devoted to the means of implementation of the other 16 SDGs and global partnership. For each SDG, there are targets that describe what needs to be realized by 2030, i.e., (1) outcome targets and (2) means of implementation targets, both types of equal importance [1]. A brief description of the 17 SDGs in terms of their title and targets is given in Table 10. For further details, see the UN 2030 Agenda [1].

APPENDIX B
KEYWORD CO-OCCURRENCE ANALYSIS

The keyword co-occurrence analysis of abstracts of related articles referenced in Section II-B was performed using the VOSviewer software,[21] version 1.6.20. To construct the co-occurrence network, binary counting was used. It only considered whether a keyword appeared in an abstract, not the number of times the keyword appeared there. Keywords that appeared in at least two abstracts were included in the analysis. To remove irrelevant keywords from the text (e.g., ''use'', ''type'', ''number'', ''link'') and map synonyms to one same keyword (e.g., ''(UN) sustainable development goal(s)'' or ''sdgs'' were mapped to ''sdg''), a thesaurus file was used. The link weights were normalized using the co-occurrence counts. The clustering used the default parameters suggested by the tool, resulting in six clusters. For a more detailed description of the parameters, see the VOSviewer manual.[22]

APPENDIX C
WIKIPEDIA ARTICLES USED AS DATA SOURCE

The 18 English-language Wikipedia articles used as a source of text for fine-tuning datasets are listed in Table 11. The table contains the article URLs and revision IDs.[23] The text of the articles was downloaded using Wikipedia's export page[24] which allows download of a specific set of articles in XML format. The date of the download was 2021-12-28. Only the current revision (the most recent version) of the articles at the specified date was downloaded, without their full history (all versions of the article). For details on Wikipedia article history, see Wikipedia pages on the topic.[25] ,[26]

APPENDIX D
SELECTED CHECKPOINT DETAILS

The fine-tuning and test configuration details of the selected checkpoints referenced in Tables 5, 6, 7, and 8 in Section V-C are given in Tables 12, 13, 14, and 15, accordingly. The four tables given in this section are organized in the same way as the tables given in Section V-C.

ACKNOWLEDGMENT

The work of Ana Gjorgjevikj was done in part when she was Ph.D. student at the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje. She is now with the Computer Systems Department, Jožef Stefan Institute in Ljubljana, where the work was completed.

REFERENCES

[1] *Transforming Our World: The 2030 Agenda for Sustainable Development, A/RES/70/1*, United Nations Gen. Assem., New York, NY, USA, 2015.
[2] *Global Indicator Framework for the Sustainable Development Goals and Targets of the 2030 Agenda for Sustainable Development, A/RES/71/313, Annex*, United Nations Gen. Assem., New York, NY, USA, 2017.
[3] M. Nilsson, D. Griggs, and M. Visbeck, ''Policy: Map the interactions between sustainable development goals,'' *Nature*, vol. 534, no. 7607, pp. 320–322, Jun. 2016.
[4] M. Nilsson, E. Chisholm, D. Griggs, P. Howden-Chapman, D. McCollum, P. Messerli, B. Neumann, A.-S. Stevance, M. Visbeck, and M. Stafford-Smith, ''Mapping interactions between the sustainable development goals: Lessons learned and ways forward,'' *Sustainability Sci.*, vol. 13, no. 6, pp. 1489–1503, Nov. 2018.

[21] https://www.vosviewer.com/
[22] https://www.vosviewer.com/getting-started
[23] https://en.wikipedia.org/wiki/Wikipedia:Revision_id
[24] https://en.wikipedia.org/wiki/Special:Export
[25] https://en.wikipedia.org/wiki/Help:Page_history
[26] https://en.wikipedia.org/wiki/Help:Permanent_link

[5] T. Bennich, N. Weitz, and H. Carlsen, "Deciphering the scientific literature on SDG interactions: A review and reading guide," *Sci. Total Environ.*, vol. 728, Aug. 2020, Art. no. 138405.

[6] *The Sustainable Development Goals Report 2023: Special Edition*, United Nations Dept. Econ. Social Affairs, New York, NY, USA, 2023.

[7] *The Sustainable Development Goals Report 2022*, United Nations Dept. Econ. Social Affairs, New York, NY, USA, 2022.

[8] A. Huovila, P. Bosch, and M. Airaksinen, "Comparative analysis of standardized indicators for smart sustainable cities: What indicators and standards to use and when?" *Cities*, vol. 89, pp. 141–153, Jun. 2019.

[9] A. Merino-Saum, P. Halla, V. Superti, A. Boesch, and C. R. Binder, "Indicators for urban sustainability: Key lessons from a systematic analysis of 67 measurement initiatives," *Ecological Indicators*, vol. 119, Dec. 2020, Art. no. 106879.

[10] A. Siragusa, I. Stamos, C. Bertozzi, and P. Proietti, "European handbook for SDG voluntary local reviews—2022 edition," Publications Office Eur. Union, Luxembourg, Luxembourg, Tech. Rep. KJ-NA-31-111-EN-N, 2022, doi: 10.2760/355330.

[11] L. Lella, N. Osés-Eraso, I. Stamos, and R. Manfredi, "Monitoring the SDGS at regional level in Eur. Regions2030 pilot project final report," Publications Office Eur. Union, Luxembourg, Luxembourg, Tech. Rep. KJ-09-23-520-EN-N, 2023, doi: 10.2760/02404.

[12] *Localizing the Post-2015 Development Agenda: Dialogs on Implementation*, United Nations Develop. Group, 2014. [Online]. Available: https://unhabitat.org/dialogues-on-localizing-the-post-2015-development-agenda

[13] A. Gjorgjevikj, K. Mishev, D. Trajanov, and L. Kocarev. (2023). *Embed4sd*. [Online]. Available: https://github.com/gjorgjevik/embed4sd

[14] A. Breuer, H. Janetschek, and D. Malerba, "Translating sustainable development goal (SDG) interdependencies into policy advice," *Sustainability*, vol. 11, no. 7, p. 2092, Apr. 2019.

[15] *Handbook for the Preparation of Voluntary National Reviews*, United Nations Dept. for Social Econ. Affairs, New York, NY, USA, 2023.

[16] *Guidelines for Voluntary Local Reviews: Vol. 1, a Comparative Analysis of Existing VLRS*, United Cities Local Governments (UCLG) UN HABITAT, Barcelona, Spain, 2020.

[17] *Sustainable Development in the European Union—Monitoring Report on Progress Towards the SDGs in an EU Context*, Publications Office Eur. Union, Luxembourg, Luxembourg, 2023.

[18] A. Siragusa, P. Proietti, C. Bertozzi, E. Coll Aliaga, S. Foracchia, A. Irving, S. Monni, M. P. Oliveira, R. Sisto, A. Siragusa, P. Proietti, and C. Bertozzi, "Building urban datasets for the SDGS. Six European cities monitoring the 2030 agenda," Publications Office Eur. Union, Luxembourg, Luxembourg, Tech. Rep. KJ-NA-30855-EN-N, 2021, doi: 10.2760/510439.

[19] A. Siragusa, M. Vizcaino, P. Proietti, and C. Lavalle, "European handbook for SDG voluntary local reviews," Publications Office Eur. Union, Luxembourg, Luxembourg, Tech. Rep. KJ-NA-30067-EN-N, 2020, doi: 10.2760/670387.

[20] M. V. Rapun, I. Stamos, P. Proietti, and A. Siragusa, "Regions2030—European regional SDG indicators," Eur. Union, Luxembourg, Luxembourg, Tech. Rep. KJ-NA-31-326-EN-N, 2022, doi: 10.2760/850788.

[21] C. Yeh, C. Meng, S. Wang, A. Driscoll, E. Rozi, P. Liu, J. Lee, M. Burke, D. B. Lobell, and S. Ermon, "SustainBench: Benchmarks for monitoring the sustainable development goals with machine learning," in *Proc. 35th Conf. Neural Inf. Process. Syst. Datasets Benchmarks Track*, Jan. 2021, pp. 1–11.

[22] R. Vinuesa, H. Azizpour, I. Leite, M. Balaam, V. Dignum, S. Domisch, A. Felländer, S. D. Langhans, M. Tegmark, and F. F. Nerini, "The role of artificial intelligence in achieving the sustainable development goals," *Nature Commun.*, vol. 11, no. 1, pp. 1–10, Jan. 2020.

[23] N. Tomašev et al., "AI for social good: Unlocking the opportunity for positive impact," *Nature Commun.*, vol. 11, no. 1, p. 2468, May 2020.

[24] M. A. Soriano, R. Berlanga, and I. Lanza-Cruz, "On the problem of automatically aligning indicators to SDGs," in *Proc. Eur. Semantic Web Conf.*, Jan. 2023, pp. 138–142.

[25] T. Matsui, K. Suzuki, K. Ando, Y. Kitai, C. Haga, N. Masuhara, and S. Kawakubo, "A natural language processing model for supporting sustainable development goals: Translating semantics, visualizing nexus, and connecting stakeholders," *Sustainability Sci.*, vol. 17, no. 3, pp. 969–985, May 2022.

[26] Y. Li, V. F. Frans, Y. Song, M. Cai, Y. Zhang, and J. Liu, "SDGdetector: An R-based text mining tool for quantifying efforts toward sustainable development goals," *J. Open Source Softw.*, vol. 8, no. 84, p. 5124, Apr. 2023.

[27] F. Sovrano, M. Palmirani, and F. Vitali, "Deep learning based multi-label text classification of UNGA resolutions," in *Proc. 13th Int. Conf. Theory Pract. Electron. Governance*, Sep. 2020, pp. 686–695.

[28] D. S. Meier, R. Mata, and D. U. Wulff, "Text2sdg: An r package to monitor sustainable development goals from text," 2021, *arXiv:2110.05856*.

[29] D. U. Wulff, D. S. Meier, and R. Mata, "Using novel data and ensemble models to improve automated labeling of sustainable development goals," 2023, *arXiv:2301.11353*.

[30] A. Hajikhani and C. Cole, "A critical review of large language models: Sensitivity, bias, and the path toward specialized AI," 2023, *arXiv:2307.15425*.

[31] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. Int. Conf. Mach. Learn.*, vol. 4, Jun. 2014, pp. 1188–1196.

[32] L. Pukelis, N. Bautista Puig, M. Skrynik, and V. Stanciauskas, "OSDG—Open-source approach to classify text data by UN sustainable development goals (SDGs)," 2020, *arXiv:2005.14569*.

[33] L. Pukelis, N. Bautista-Puig, G. Statulevičiūtė, V. Stančiauskas, G. Dikmener, and D. Akylbekova, "OSDG 2.0: A multilingual tool for classifying text data by UN sustainable development goals (SDGs)," 2022, *arXiv:2211.11252*.

[34] M. Angin, B. Taşdemir, C. A. Yılmaz, G. Demiralp, M. Atay, P. Angin, and G. Dikmener, "A RoBERTa approach for automated processing of sustainability reports," *Sustainability*, vol. 14, no. 23, p. 16139, Dec. 2022.

[35] D. F. Hsu, M. T. LaFleur, and I. Orazbek, "Improving SDG classification precision using combinatorial fusion," *Sensors*, vol. 22, no. 3, p. 1067, Jan. 2022.

[36] L. M. Fonseca, J. P. Domingues, and A. M. Dima, "Mapping the sustainable development goals relationships," *Sustainability*, vol. 12, no. 8, p. 3359, Apr. 2020.

[37] J. E. Guisiano, R. Chiky, and J. D. Mello, "SDG-meter: A deep learning based tool for automatic text classification of the sustainable development goals," in *Proc. Asian Conf. Intell. Inf. Database Syst.*, Jan. 2022, pp. 259–271.

[38] T. B. Smith, R. Vacca, L. Mantegazza, and I. Capua, "Natural language processing and network analysis provide novel insights on policy and scientific discourse around sustainable development goals," *Sci. Rep.*, vol. 11, no. 1, pp. 1–10, Nov. 2021.

[39] E. Fotopoulou, I. Mandilara, A. Zafeiropoulos, C. Laspidou, G. Adamos, P. Koundouri, and S. Papavassiliou, "SustainGraph: A knowledge graph for tracking the progress and the interlinking among the sustainable development goals' targets," *Frontiers Environ. Sci.*, vol. 10, p. 2175, Oct. 2022.

[40] P. Mishra, S. K. Narayanasamy, and K. Srinivasan, "Context-aware embedded language transformers for evaluating climate change-based sustainable development goals," *IEEE Access*, vol. 13, pp. 65757–65775, 2025.

[41] H. Cho and E. Ackom, "Artificial intelligence (AI)-driven approach to climate action and sustainable development," *Nature Commun.*, vol. 16, no. 1, p. 1228, Jan. 2025.

[42] P. Koundouri, A. Alamanos, A. Plataniotis, C. Stavridis, K. Perifanos, and S. Devves, "Assessing the sustainability of the European green deal and its interlin kages with the SDGs," *NPJ Climate Action*, vol. 3, no. 1, p. 23, Mar. 2024.

[43] W. Benjira, F. Atigui, B. Bucher, M. Grim-Yefsah, and N. Travers, "Automated mapping between SDG indicators and open data: An LLM-augmented knowledge graph approach," *Data Knowl. Eng.*, vol. 156, Mar. 2025, Art. no. 102405.

[44] F. Larosa, S. Hoyas, H. A. Conejero, J. Garcia-Martinez, F. F. Nerini, and R. Vinuesa, "Large language models in climate and sustainability policy: Limits and opportunities," 2025, *arXiv:2502.02191*.

[45] P. Koundouri, P.-S. Aslanidis, K. Dellis, A. Plataniotis, and G. Feretzakis, "Mapping human security strategies to sustainable development goals: A machine learning approach," *Discover Sustainability*, vol. 6, no. 1, p. 96, Feb. 2025.

[46] C. Li, Z. Chen, Q. Jiang, M. Yue, L. Wu, Y. Bao, B. Huang, A. B. Wang, Y. Tan, and Z. Xu, "Impacts of government attention on achieving sustainable development goals: Evidence from China," *Geography Sustainability*, vol. 6, no. 2, Apr. 2025, Art. no. 100233.

[47] N. Strelkovskii and N. Komendantova, "Integration of UN sustainable development goals in national hydrogen strategies: A text analysis approach," *Int. J. Hydrogen Energy*, vol. 102, pp. 1282–1294, Feb. 2025.
[48] S. Borchardt, G. Barbero Vignola, D. Buscaglia, M. Maroni, and L. Marelli, "Mapping EU policies with the 2030 agenda and SDGS—Fostering policy coherence through text-based SDG mapping," Publications Office Eur. Union, Luxembourg, Luxembourg, Tech. Rep. KJ-NA-31-347-EN-N, 2022, doi: 10.2760/110687.
[49] R. Raman, P. Singh, V. K. Singh, R. Vinuesa, and P. Nedungadi, "Understanding the bibliometric patterns of publications in IEEE access," *IEEE Access*, vol. 10, pp. 35561–35577, 2022.
[50] R. C. Morales-Hernández, J. G. Jagüey, and D. Becerra-Alonso, "A comparison of multi-label text classification models in research articles labeled with sustainable development goals," *IEEE Access*, vol. 10, pp. 123534–123548, 2022.
[51] P. Nedungadi, S. Surendran, K.-Y. Tang, and R. Raman, "Big data and AI algorithms for sustainable development goals: A topic modeling analysis," *IEEE Access*, vol. 12, pp. 188519–188541, 2024.
[52] N. J. Van Eck and L. Waltman, "Software survey: VOSviewer, a computer program for bibliometric mapping," *Scientometrics*, vol. 84, no. 2, pp. 523–538, Aug. 2010.
[53] A. Gjorgjevikj, "Knowledge transfer in deep learning with small text datasets," Ph.D. dissertation, Fac. Comput. Sci. Eng., SS. Cyril Methodius Univ. Skopje, Skopje, North Macedonia, 2023.
[54] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *Proc. Int. Workshop Similarity-Based Pattern Recognit.*, Dec. 2014, pp. 84–92.
[55] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[56] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.
[57] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.
[58] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," 2019, *arXiv:1908.10084*.
[59] T. Gao, X. Yao, and D. Chen, "SimCSE: Simple contrastive learning of sentence embeddings," 2021, *arXiv:2104.08821*.
[60] J. Ni, G. Hernández Ábrego, N. Constant, J. Ma, K. B. Hall, D. Cer, and Y. Yang, "Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models," 2021, *arXiv:2108.08877*.
[61] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.
[62] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*.
[63] S. Lundberg and S. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2017, pp. 1–11.
[64] H. Chen, J. D. Janizek, S. Lundberg, and S.-I. Lee, "True to the model or true to the data?" 2020, *arXiv:2006.16234*.
[65] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Kdd*, Jan. 1996, pp. 226–231.
[66] D. Cer, Y. Yang, S.-Y. Kong, N. Hua, N. Limtiaco, R. St. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, and R. Kurzweil, "Universal sentence encoder," 2018, *arXiv:1803.11175*.
[67] Y. Yang, D. Cer, A. Ahmad, M. Guo, J. Law, N. Constant, G. H. Abrego, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, and R. Kurzweil, "Multilingual universal sentence encoder for semantic retrieval," 2019, *arXiv:1907.04307*.
[68] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2020, pp. 5776–5788.
[69] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," 2019, *arXiv:1910.01108*.
[70] K. Song, X. Tan, T. Qin, J. Lu, and T. Liu, "MPNet: Masked and permuted pre-training for language understanding," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2020, pp. 16857–16867.
[71] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers, "MTEB: Massive text embedding benchmark," 2022, *arXiv:2210.07316*.

ANA GJORGJEVIKJ received the Ph.D. degree in computer science and engineering from Ss. Cyril and Methodius University in Skopje, in 2023, with a particular focus on natural language processing and machine learning. She is currently a Postdoctoral Researcher at the Jožef Stefan Institute, Ljubljana, Slovenia, and was awarded a Horizon Europe MSCA Postdoctoral Fellowship in the 2024 call. She also has 12 years of professional experience as a Software Engineer. Her research interests include data science, machine learning, natural language processing, representation learning, multi-task learning, and meta-learning.

KOSTADIN MISHEV received the Ph.D. degree in computer science and engineering from the Ss. Cyril and Methodius University in Skopje, in 2023. He is currently an Assistant Professor with the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje. He has published more than 50 scientific articles and has coordinated more than 20 research and industry projects. His expertise spans data science, speech technologies, VoiceBots, and the application of artificial intelligence in healthcare.

DIMITAR TRAJANOV (Member, IEEE) is currently a Full Professor with the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, and a Visiting Research Professor with Boston University. He serves on the National Higher Education Accreditation Board and as an AI Consultant for leading companies. Previously, he was the Founding Dean of the Faculty of Computer Science and Engineering, from 2011 to 2015, establishing it as the largest technical faculty in the nation. He also led the Department of Information Systems and Network Technologies, from 2015 to 2022, and directs the Regional Social Innovation Hub, a collaboration with UNDP. He has authored over 230 articles and seven books and has led more than 40 of the 70+ research and industry projects he has participated in. His research focuses on AI, data science, machine learning, NLP, and their applications in various domains. His research interests include data science, machine learning, NLP, FinTech, semantic web, e-commerce, technology for development, ESG, and climate change. In 2023, he received the "Best Scientist Award" from Ss. Cyril and Methodius University in Skopje.

LJUPCO KOCAREV (Fellow, IEEE) is a member with the Macedonian Academy of Sciences and Arts, retired professor at the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, and the Funding Director of the Research Center for Computer Science and Information Technologies, Macedonian Academy of Sciences and Arts. He was the President of the Macedonian Academy of Sciences and Arts, from January 2020 to December 2023. His work has been supported by the Macedonian Ministry of Education and Science, Macedonian Academy of Sciences and Arts, NSF, AFOSR, DoE, ONR and ONR Global, NIH, STMicroelectronics, NATO, TEMPUS, FP6, FP7, Horizon 2020, Alliance of National and International Organizations for the Belt and Road Regions, DAAD, and DFG. His scientific interests include networks, nonlinear systems and circuits, dynamical systems and mathematical modeling, machine learning, and computational biology. He has co-authored more than 200 journal articles and has been granted eight patents worldwide.