Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.050585

ARTICLE

Relational Turkish Text Classification Using Distant Supervised Entities and Relations

Halil Ibrahim Okur[1,2,*], Kadir Tohma[1] and Ahmet Sertbas[2]

[1]Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Iskenderun Technical University, Hatay, 31200, Turkey

[2]Department of Computer Engineering, Faculty of Engineering, Istanbul University-Cerrahpasa, Istanbul, 34310, Turkey

*Corresponding Author: Halil Ibrahim Okur. Email: hibrahim.okur@iste.edu.tr

Received: 10 February 2024 Accepted: 21 March 2024 Published: 15 May 2024

ABSTRACT

Text classification, by automatically categorizing texts, is one of the foundational elements of natural language processing applications. This study investigates how text classification performance can be improved through the integration of entity-relation information obtained from the Wikidata (Wikipedia database) database and BERT-based pre-trained Named Entity Recognition (NER) models. Focusing on a significant challenge in the field of natural language processing (NLP), the research evaluates the potential of using entity and relational information to extract deeper meaning from texts. The adopted methodology encompasses a comprehensive approach that includes text preprocessing, entity detection, and the integration of relational information. Experiments conducted on text datasets in both Turkish and English assess the performance of various classification algorithms, such as Support Vector Machine, Logistic Regression, Deep Neural Network, and Convolutional Neural Network. The results indicate that the integration of entity-relation information can significantly enhance algorithm performance in text classification tasks and offer new perspectives for information extraction and semantic analysis in NLP applications. Contributions of this work include the utilization of distant supervised entity-relation information in Turkish text classification, the development of a Turkish relational text classification approach, and the creation of a relational database. By demonstrating potential performance improvements through the integration of distant supervised entity-relation information into Turkish text classification, this research aims to support the effectiveness of text-based artificial intelligence (AI) tools. Additionally, it makes significant contributions to the development of multilingual text classification systems by adding deeper meaning to text content, thereby providing a valuable addition to current NLP studies and setting an important reference point for future research.

KEYWORDS

Text classification; relation extraction; NER; distant supervision; deep learning; machine learning

1 Introduction

Natural Language Processing (NLP) represents the effort to interpret the complexity and richness of human language through computer systems. In this field, a particularly notable issue in recent years has been the extraction or classification of desired meaningful information from vast repositories

of documents. The task of extracting, analyzing, or categorizing information from texts has become increasingly challenging due to the presence of large amounts of texts or documents. In this context, text classification emerges as a pivotal approach within the ambit of Natural Language Processing, striving to systematically assign labels to textual entities such as sentences, paragraphs, and documents. A sentence or expression is formed by the combination of the syntactic and semantic features of words. Relationships between sentences create paragraphs, and relationships between paragraphs create texts. In this process, the compositional properties of documents and heterogeneous information pose a crucial challenge for Natural Language Processing. The unique structural complexity of each document and the variety of information types it contains significantly affect the ability of NLP techniques to extract meaningful information from these texts. Therefore, understanding the syntactic and semantic properties of texts, as well as how these properties come together to form a coherent meaning, is critical for developing an effective text classification system [1].

Texts, on the other hand, facilitate meaningful and regular communication in the exchange of information by enabling individuals to convey their knowledge and thoughts to others in writing. Due to the structural, regional, cultural, and temporal variations based on the unique grammar rules of each language, there may be differences in text processing steps across different languages. Therefore, the methods used should be suitable for the morphology of the language, and texts should be prepared accordingly [2].

Studies related to introducing the syntactic and semantic structure of natural language to computer systems have been conducted in the literature for many years. Information extraction, summarization, question-answering, text classification, and various other natural language processing problems can be given as examples of these studies. The first step in solving these problems is to convert texts into a format that artificial intelligence systems can use, which involves preprocessing the texts. A good text preprocessing stage can impact the performance of artificial intelligence systems [3]. However, the development of models such as GPT-4 (generative pre-trained transformer) [4] and Llama2 [5], which can directly read web pages with complex tags without the need for text preprocessing, has diminished the significance of preprocessing. Particularly, these new-generation artificial intelligence systems can process the structural and semantic complexity of natural language without the intensive preprocessing steps previously required. On the other hand, the deep cleaning and structuring provided by text preprocessing, especially for specific applications and complex language processing requirements, remains a valuable step.

Text datasets such as news articles, social media posts, and customer reviews can be utilized for text classification. Tasks such as category determination, sentiment analysis, and classification can be performed on these datasets. In recent years, deep learning-based text classifiers have shown effectiveness not only in text classification but also in natural language understanding tasks such as question-answering and natural language inference [6,7]. Deep learning (DL) models, including Deep Neural Networks (DNN), Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), Graph Neural Networks (GNN), or hybrid neural network models, can be used in text classifiers [8–10]. Additionally, various machine learning algorithms such as Support Vector Machines (SVM) and Decision Trees (DT) can also be employed as classification models [11].

In addition to various models for text classification, there are also several text vector representation models in the literature. Methods such as TF-IDF (term frequency-based), Word2Vec (word to vector) [12], and GloVe (global vectors for word representation) [13] are examples of word embedding techniques used in text classification, where the ordering and semantic features of words are taken into account [14]. In addition to that, language model-based approaches such as BERT (Bidirectional

Encoder Representations from Transformers) are widely used in text classification. BERT is a pretraining model that learns general language understanding by leveraging large amounts of text data. It can then be fine-tuned for specific tasks, including text classification, to achieve effective results [15]. By employing bidirectional attention mechanisms, BERT can capture a more comprehensive representation of texts, enabling a better understanding of word context and relationships.

There are numerous text classification studies available that offer various techniques and solutions for English texts [16,17]. Furthermore, the rapid development of pre-trained word representation models and deep neural networks has facilitated the exploration of novel approaches in NLP subtasks. Literature also includes research papers on approaches used in text preprocessing, word representation models, and text classifiers. These research publications focus on a detailed examination of text classification steps and the introduction of existing works [14,18,19].

In this study, text classification performance is improved by enriching texts with entities and their relationships. BERT-based pre-trained models are used to detect entities within the text. After entity detection, the relationships associated with these entities are queried in the Wikidata database using the SPARQL (SPARQL Protocol and RDF Query Language) query language. The obtained entity-relation information is integrated with the text dataset for classification, enriching the text data. Different machine learning algorithms (Linear SVM, Decision Tree) and deep learning algorithms (DNN, CNN, GNN) are employed to demonstrate the classification performance on Turkish and English text datasets. The results indicate that enriching texts with entity-relation information yields satisfactory performance for text classification tasks. The contributions of this study are listed below:

- Relational text classification approach: This study aims to capture relationships within texts using entity-relation information and examine their impact on classification performance. BERT-based pre-trained models are utilized for entity detection within the text, and the obtained entity-relation information is retrieved from the Wikidata database using the SPARQL query language. These pieces of information are incorporated into the text content, enriching the text data. Various machine learning and deep learning algorithms are applied to texts enriched with entity relations to demonstrate classification performance. Consequently, a novel relational text classification approach based on entity-relation information is introduced for Turkish text classification tasks.
- Database creation and distant supervision: In the scope of this study, entity-relation knowledge bases are created using the SPARQL query language and the Wikidata database. Entity-relation information for English and Turkish is collected, providing resources for relational text classification tasks through remote supervision in NLP studies. Moreover, automatic data collection and updating mechanisms can be developed to obtain larger relational knowledge bases on broader sources. This enables continuous updating of entity-relation information and the addition of new data.
- Text classification performance: The introduced relational model in this study shows an improvement in text classification performance on text datasets previously used for the same task in the literature.

When examining the contributions, it is observed that the use of entity-relation information can enhance the performance in Turkish text classification. Our motivation is to enable the enrichment of entity relations in texts through automatic and distant supervision, supporting easier and comprehensive access to information in domains where rapidly evolving text-based AI tools are employed. This study presents a new approach for Turkish studies with a relational text classification approach based

on entity-relation information. This method aims to investigate the enrichment of information in texts and the impact of relationships on text classification in a more detailed manner.

Section 2 of this study examines relevant literature. Text classification tasks conducted in Turkish and other languages, along with the results obtained from these tasks, are presented in the literature. In Section 3, the proposed relational classification method in this study is explained in detail. The utilized datasets, text preprocessing steps, entity detection, obtaining entity relations, obtaining text vectors, and classification algorithms are presented in detail. Section 4 presents the experimental results and performance metrics of this study. Finally, in Section 5, a summary and synthesis of the findings and implications of the study are provided. The section also discusses the limitations of the research and offers suggestions for future studies.

2 Literature Review

Text classification is an important research topic in the field of natural language processing and is widely used in various application domains. Specifically, the automatic assignment of texts to specific categories or labels holds significant importance in information management, sentiment analysis, spam filtering, document classification, and many other areas. Literature has proposed numerous methods and techniques in this regard. However, there are limited studies on the use of entity relationships in texts and their impact on text classification. Therefore, this study addresses the lack of utilization of entity relationships in Turkish text classification and comprehensively examines the existing literature in terms of text classification. Additionally, our study leverages machine learning and deep learning methods to enhance the performance of entity relationships in text classification. This literature review summarizes the current state of Turkish text classification and emphasizes how the use of entity relationships can contribute to improvements in this field. Within this scope, the information and performance of text classification or relational text studies in the literature have been examined. The classification method that shows the best result among the reviewed literature studies is presented in the table. Information about the studies examined in the literature is shown in Table 1.

Table 1: Text classification studies in the literature

Ref.	Dataset	Classification	Accuracy	F1-score
[20]	TTC-3600	Random forest	91.03%	–
[21]	Web News	SVM	–	90.10%
[22]	TTC-3600	Multinomial NB	90%	–
[23]	TTC-3600, 20 Newsgroup	SVM	–	78.1%, 98.0%
[24]	TTC-3600	CNN	93.3%	–
[25]	20 Newsgroup	DNN	82.6%	–
[26]	20 Newsgroup	CNN	79%	–

In a study conducted by Kılınc et al., a new dataset named TTC-3600 was developed specifically for text categorization research involving Turkish news articles. The researchers evaluated five different text classifiers and two feature selection methods on TTC-3600. The results indicated that the Random Forest classifier, combined with feature ranking-based feature selection, achieved an accuracy of 91.03% [20].

The study conducted by Flisar et al. aimed to enhance the performance of short text classification by enriching the short text document representation model with the DBpedia knowledge base. They implemented text enrichment methods using the DBpedia Spotlight framework and evaluated their effectiveness. The experiments conducted in this study demonstrated that this approach outperforms baseline methods across various datasets using different classifiers [21].

In the study conducted by Kuyumcu et al., the Turkish TTC-3600 dataset was used for text classification using the FastText word vector representation model [27] and classical machine learning algorithms. Approximately 90% accuracy was achieved using Multinomial NB classification [22].

In the study by Parlak, the contributions of preprocessing techniques to classification performance were evaluated. The effect of two preprocessing techniques on different news datasets was examined. The highest F1-score was 0.781 for the Turkish dataset (TTC-3600), while the highest F1-score was 0.980 for the English dataset (20Newsgroup) [23].

The text classification study conducted by Aci et al. was conducted on the TTC-3600 dataset using Convolutional Neural Networks and the Word2Vec method, and the texts were preprocessed using the Zemberek software. The proposed method achieved a performance accuracy of 93.3% [24].

Pittaras et al. [25] conducted a study exploring methods to enrich the input with semantic information for text classification tasks using deep neural networks (DNNs). Semantic inferences are made from texts using the WordNet semantic graph, forming weighted concept terms that constitute a semantic frequency vector. Concepts are selected using various semantic disambiguation techniques and merged with Word2Vec embeddings considering semantic relationships. Experimental results demonstrate that this semantic enrichment significantly enhances classification performance, with the concatenation method yielding the best results. The study also examines the impact of term frequency-inverse document frequency normalization on semantic vectors and the potential for dimensionality reduction.

Lezama et al. presented a novel approach based on extracting relationships from Wikipedia. They focused on extracting semantic relationships, including synonymy, hyponymy, and hyperonymy. The proposed relationship-based embedding models were evaluated by performing text classification using CNN, and a maximum accuracy of 79% was achieved on the 20-Newsgroup dataset [26].

Knowledge bases such as YAGO [28], Freebase [29], DBpedia [30], and Wikidata [31] can be used to extract entities and relationships from texts. Many studies have been conducted on these knowledge bases for various NLP tasks. These studies focus on entity and relationship extraction in texts [32] or labeling entities and relationships using distant supervision from a knowledge base [33,34]. The detection of entity and relationship information in texts through distant supervision can also be beneficial for text classification (TC) tasks [35,36].

As seen in Table 1, various text classification tasks have been performed on the Turkish TTC-3600 dataset using both traditional machine learning algorithms and deep learning models. Additionally, for performance comparison in different languages, the 20Newsgroup and BBC-News datasets consisting of English news articles are also utilized in the literature. In addition to these studies, academic publications also exist for relational text analysis.

3 Proposed Methods

This study focuses on the identification of entities within Turkish text datasets, the addition of entity relationships using distant supervision, and the classification of labeled texts using machine learning and deep learning algorithms. A knowledge base containing entity-relationship information

extracted from Wikidata, a Wikipedia-based database [31], was obtained using Python tools. The identified entities and relationships from this knowledge base were added to the texts used for classification. Turkish pre-trained BERT model [37] and English pre-trained BERT models [15] were utilized to convert the texts into vector representations. Subsequently, the text classification performances of deep learning and machine learning algorithms were demonstrated.

This work addresses the lack of entity-relationship utilization in Turkish text classification methods. The proposed classification model was also applied to English text datasets, allowing for a comparison of Turkish-English text classification performance. According to the proposed classification model shown in Fig. 1, the first step involves preprocessing Turkish texts and determining the entity relationships within the texts. In the text preprocessing stage, the texts undergo various cleaning steps to prepare them for standardized and effective use in subsequent stages. Entities within the cleaned texts are identified using a pre-trained Named Entity Recognition (NER) model, and the relationships about these entities are searched in the Wikidata database using the SPARQL query language. The identified entities and relationships are then utilized to enrich the texts. Subsequently, the cleaned and entity-relationship labeled texts are transformed into vector form using a pre-trained BERT model. The vectorized texts are then classified using deep learning techniques and machine learning algorithms, and the accuracy performances are measured. In addition to working with Turkish text datasets in this study, English text datasets from the literature were also used for model performance comparison.

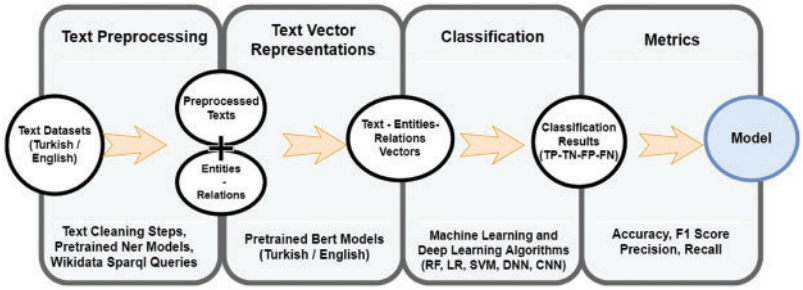

Figure 1: Proposed Turkish relational text classification model

3.1 Datasets

The text dataset can be obtained from various platforms such as social media, the web, business domains, user comments, and more. Texts may contain differences in terms of adhering to grammatical rules depending on the environment from which they are collected. The desire to adhere to grammatical rules is more prominent in written language compared to spoken language. Therefore, text collected from social media, which is closer to spoken language, is more likely to contain grammatical errors. On the other hand, texts in official documents are created with more attention to grammatical rules, resulting in fewer errors. The cleaning process of texts can also vary depending on the source from which they are obtained. In social media data, people may use abbreviations, slang, or nonsensical words that are not part of the language's characteristics. Many texts and documents may contain numerous unnecessary words. In algorithms based on statistical and probabilistic learning, noise and irrelevant features can harm system performance. Preprocessing or cleaning the text is necessary to classify such texts and correct these issues.

Text data can be extracted from the internet using web scraping methods and then sorted and listed. Texts, tags, links, and category data on web pages can be obtained through a computer program or Python libraries. Since text data obtained from news websites is more formal and likely to yield more accurate results in finding entity relationships, it was selected as the training data for the proposed model. The Turkish TTC-4900 dataset [20], which is commonly used in the literature for text classification tasks, was used for model training. Additionally, the widely used 20Newsgroups dataset [38] and the BBC dataset [39] in the literature for English text classification tasks were used for model performance comparison.

In Table 2, the document counts and category numbers for each dataset are provided. The TTC-4900 dataset consists of 4900 samples in 7 different categories. The BBC-News dataset comprises 2224 samples in 5 different categories. The 20 Newsgroups dataset, although originally consisting of 18846 samples in 20 different categories, was reduced to 6517 samples in 7 different categories for this study. Fig. 2 illustrates the distribution of categories for the datasets.

Table 2: Information on the used datasets

Datasets	Number of documents	Number of categories
TTC-4900	4900	7
BBC-News	2224	5
20 Newsgroups	6517	7

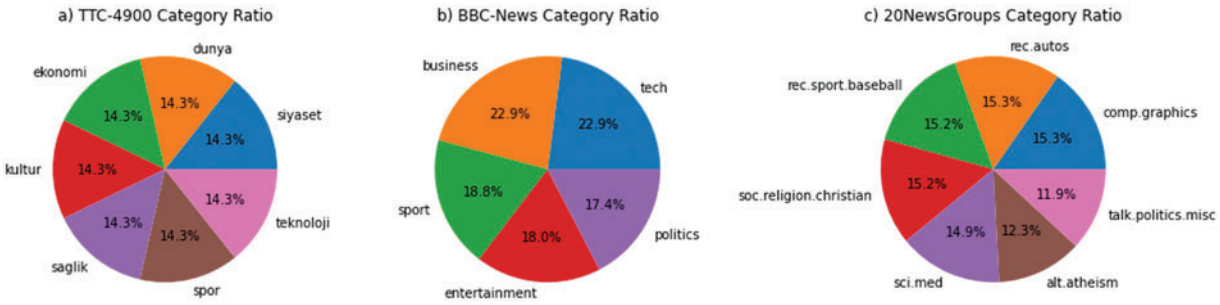

Figure 2: (a) TTC-4900, (b) TRT-Haber, (c) 20Newsgroups datasets category ratios

3.2 Text Preprocessing

In text classification, preprocessing the dataset is an important step that can enhance the performance of artificial intelligence models. Text preprocessing involves various stages such as cleaning the data from noise, correcting spelling mistakes, handling slang or abbreviations, ensuring text consistency by converting to lowercase or uppercase, tokenization (breaking down text into smaller units), and finding word roots through stemming and lemmatization. Removing frequently used words, known as stop words, is also necessary to reduce their impact on model performance. Standardizing capitalization is important for consistency in text representation. Preprocessing also addresses anomalies like slang and abbreviations by converting them to their formal counterparts. Noise removal and spelling correction may be needed to enhance human understanding of documents. Stemming and lemmatization help to unify word forms and determine the root of a word. For English texts, Snowball Stemmer [40] and NLTK library [41] are commonly used, while for Turkish texts,

the Snowball and Zemberek libraries [42] are popular choices. The preprocessing techniques have been studied and evaluated in various research works to observe their impact on text classification performance.

3.3 Entity and Relations in Texts

Named Entity Recognition (NER) is the process of finding and classifying named entities in text according to predefined entity categories. NER recognizes tagged entities in text fragments and categorizes them into predefined categories such as person, location, and organization. NER is based on a variety of natural language applications and can be used for tasks such as question answering, text summarization, machine translation, and text classification. While NER systems are successful at producing reasonable recognition accuracy, they often require a lot of human effort to carefully design rules or features. There are corpus datasets labeled for English such as OntoNotes [43], and Winer [44]. It has been introduced in the Turkish multi-layered corpus publication [45], which includes 9600 sentences from English-Penn-Treebank and includes the morphologies of words, named entities, senses, and semantic role tags. In addition, there is also a CRF-based Turkish NER approach [46]. In recent years, the transfer learning approach, in which Transformer Neural Network-based BERT model [15] and pre-trained models are used in NER applications. With this approach, a representation containing entity words can be made on a dataset by using the BERT model. Using a pre-trained NER model to obtain entities in texts saves performance and time costs.

The relation can be defined as the connection between two or more entities. For example, there can be a marriage relationship between two individuals. There can be membership or ownership relationships between a person and an organization. Relation Extraction is a method used to find the relationships between entities within a text. To identify relationships, entities such as person, location, organization, object, and verb structures need to be extracted from the text. Objects can be a place name, an organization, or a person's name. The goal is to establish connections between them.

The subject-relation-object (S-R-O) structure is used to indicate the relationship between the subject and the object [47,48]. For example, in the sentence *"Ankara is the capital of Turkey"* there is a capital relationship between Ankara and Turkey. This relationship structure can be represented as *(Ankara–Capital–Turkey)*. Obtaining a knowledge base that contains entity relationships is necessary to design a relational model.

With the help of a dataset consisting of relationships between entities, relationships can be automatically assigned to entities within the text using the distant supervised learning method. Wikipedia, which is a library that allows the extraction of entities and relationships in many languages, including Turkish, is one of the most suitable examples for this purpose. Entities and relationships can be obtained from the Wikidata database associated with Wikipedia using Python tools and the SPARQL query language. For example, information related to entities such as *Description, Occupation, Field of Work, Position Held, Hashtags, etc*., can be obtained from Wikidata.

Thus, entities obtained from the text through a pre-trained NER model and their relationship information within Wikidata can be obtained using the SPARQL query language. These entities and relationships are then added to the preprocessed text to enrich it. It may also be necessary to clean and trim relationships that are not relevant to the text. While BERT and similar models focus on specific categories for Named Entity Recognition (NER), our work transcends this limitation by linking entities identified by NER with a comprehensive knowledge base like Wikidata, thereby adding entity-linking functionality. This method provides a broader context for entities identified by NER, allowing for a more detailed analysis of entity information in texts. Through SPARQL queries, it searches for

entities in Wikidata that best match the terms (with the highest matching score), utilizing the unique identifier (ID) of the found entity. This approach not only enables a better understanding of entities within a broader context in texts but also overcomes the limitations of entity identification and linking processes, contributing to the enrichment of entity and relationship information.

Our proposed entity-relation detection and text integration method is shown in Fig. 3. First, the text tokenization and preprocessing steps described in Section 3.2 were applied to the datasets that will be used for text classification. Text cleaning processes were performed using the NLTK and Zemberek libraries. Then, the NER task was carried out to detect entities that could belong to individuals and organizations within the preprocessed texts.

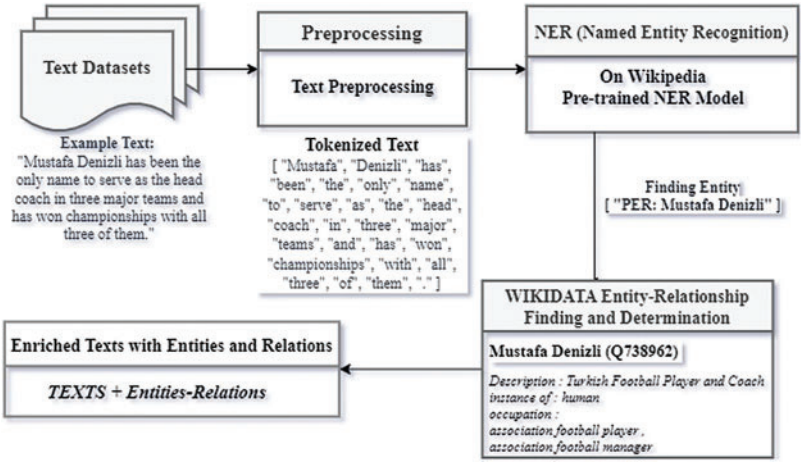

Figure 3: Finding entities in texts and Wikidata entity-relation detection

For the Named Entity Recognition (NER) task, the pre-trained transformer models **"savasy/bert-base-turkish-ner-cased"** for Turkish texts and **"Jean-Baptiste/roberta-large-ner-english"** for English texts have been utilized. These models are open-source and available for broad use. The "savasy/bert-base-turkish-ner-cased" model was trained using the WikiANN dataset, a multilingual dataset available in 176 languages, consisting of entities with LOC (location), PER (person), and ORG (organization) labels extracted from Wikipedia pages. This dataset served as a knowledge base for acquiring entities in both Turkish and English languages. Therefore, both models used are not self-trained but are existing open-source models trained with the widely used WikiANN dataset. This approach has allowed us to leverage well-known and reliable sources to achieve high accuracy and consistency in the NER task [49].

After identifying entities within the texts, we enriched them with potential relationships by creating a relationship knowledge base model using the SPARQL query language on Wikidata, a multi-language corpus owned by Wikipedia. These entities, when queried against the Wikidata database, return a Wikidata ID, which in turn provides access to a page detailing various attribute relationships, including occupation-specific ones for individuals. This process allows for the extraction of related entities via new SPARQL queries, effectively leveraging Wikidata as a knowledge base (KB) for obtaining entity relationships and enriching the detected entities with additional related entities from the Wikidata KB. Our methodology integrates structured information retrieved from Wikidata with unstructured data, encompassing entities' descriptions, related texts, and other explanatory information that goes beyond structured triples. This blend of structured and unstructured data

follows an integration approach akin to the Kepler [50] model, enabling a deeper understanding of entities and relations. The Kepler model innovates by merging knowledge embedding and language modeling techniques, facilitating the integration of structured knowledge base information with unstructured textual data, thus offering a comprehensive framework for enhancing the richness and context of entity-relationship data.

In our study, only those entities identified as 'person' through Named Entity Recognition (NER) in texts are queried in the Wikidata database using SPARQL. This targeted approach focuses on a limited set of relationships related to individuals specifically, their occupations, positions, and descriptions—thereby reducing potential noise. While this process may seem to limit the scope of data collection initially, it significantly reduces the inclusion of irrelevant information, thereby accelerating the analysis process and facilitating the interpretation of results. Specifically, the use of the 'person' tag allows for an in-depth examination of social dynamics, influence relations, and individuals' societal roles, clarifying the focus of our research and enabling more effective extraction of relevant information.

Distant supervision is utilized as a method for automatic labeling through the use of databases like Wikidata, enriching large datasets. This enables the labeling of entity relationships within texts. On the other hand, 'entity linking' refers to the process of matching entities identified in texts with their unique identifiers in external sources. In our work, entity linking has been employed to associate entities within texts with their corresponding entries in Wikidata, while distant supervision has been used to automatically label these entities and their related relationships.

Entity: Mustafa Kemal Ataturk–WikidataID: Q5152: (Mustafa Kemal Ataturk-occupation-military officer-member of an armed force or uniformed service who holds a position of authority), (Mustafa Kemal Ataturk-occupation-statesperson-civil servant or politician in high government offices), ... (Fig. 4).

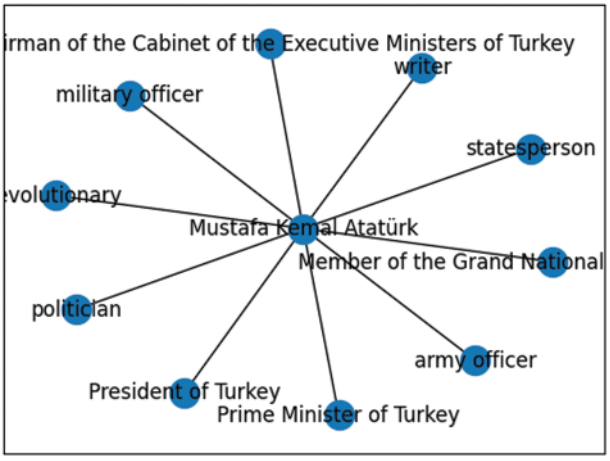

Figure 4: Entity (Mustafa Kemal Ataturk) and relations

For example, let us consider a text where the entity name "Mustafa Kemal Ataturk" or "Ataturk" is identified using a pre-trained NER model. We search for the identified entity name within Wikidata using a SPARQL query. If a matching result is found, the Wikidata ID value, "Q5152," corresponding to the entity name "Ataturk" will be obtained as the query result. Then, this entity ID value can be used in a new query to retrieve the desired attributes, i.e., relationships and descriptions, for the person

from the Wikidata page "Mustafa Kemal Ataturk". An example of the resulting entity relationships is shown in Fig. 4.

When examining these examples, the relationships obtained from Wikidata consist of the occupation, position held, and description related to individuals. These descriptive relationship information about the person in the text has a facilitating effect in determining the category of the texts. Information related to other types of relationships has not been utilized. Increasing the relationships makes the classification task more complex and challenging. As seen in Table 3, the numbers of identified entities within the datasets and the numbers of relationships obtainable from Wikidata through these entities are shown.

Table 3: Number of entities and relationships added to the datasets with the proposed model

Datasets	Number of documents	Number of all entities detected by NER (Per, Org, Loc, Misc)	Number of Wikidata relationships detected with only PER-tagged entities
TTC-4900	4900	32070	4952
BBC-News	2224	15304	16527
20Newsgroups	6517	38179	14992

As a result, the text datasets for classification have been enriched by identifying entities and their relationships within the texts. Thus, the aim is to demonstrate the positive impact on classification performance by adding new information provided by the entities within the texts. The enriched texts have been transformed into vectors to prepare them for classification.

3.4 Text Vector Representations

Text, entity, and relationship vector representations have been obtained using the pre-trained Turkish BERT (BERTurk) model [51] and the English BERT base model [15] as text vector representations. the BERT model is highlighted as a fundamental tool for obtaining vector representations of texts, entities, and relationships. The BERTurk and English BERT models offer a deep learning-based approach for Turkish and English texts, respectively. Trained on large and diverse datasets, these models produce powerful representations that capture the complex structure and semantic properties of language. Specifically, the BERTurk model was trained using a special corpus provided by Kemal Oflazer, comprising the Turkish OSCAR Corpus, Turkish Wikipedia Dump, and OPUS corpora which totals 35 GB and over 44 billion tokens. BERT models facilitate robust vector representations for texts, akin to their utility in natural language processing subtasks like part-of-speech tagging, named entity recognition, and question answering.

For each text, we utilize a BERT model to derive an n-dimensional vector representation. This representation comprehensively reflects the linguistic and semantic features of the text. Simultaneously, m-dimensional BERT vector representations for each entity-relationship derived from Wikidata are generated. This process ensures a thorough analysis of texts as well as associated entities and relationships, thereby creating a rich vector representation that includes both linguistic and ontological properties of a text.

For each text in Eq. (1), we utilize a BERT model to obtain an n-dimensional vector representation:

$$\mathbf{V}_{\text{BERT}} = [v_1, v_2, \ldots, v_n] \tag{1}$$

Additionally, for each entity and its associated relations obtained from Wikidata in each text, we derive m-dimensional BERT vector representations in Eq. (2):

$$\mathbf{V}^{i}_{\text{ER}} = \left[er^{i}_{1} + er^{i}_{2} + \ldots + er^{i}_{m}\right] \; for\, i = 1, 2, \ldots, j \tag{2}$$

The average of these m-dimensional vectors (representing the average entity-relation vector) is computed in Eq. (3):

$$\mathbf{V}_{\text{ER}-\text{avg}} = \frac{1}{j} \sum_{i=1}^{j} \mathbf{V}^{i}_{\text{ER}} \tag{3}$$

The combined vector for a single text in Eq. (4), incorporating both the text's n-dimensional BERT representation and the m-dimensional average entity-relation vector, is thus n+m dimensional:

$$\mathbf{V}_{\text{combined}} = \left[\mathbf{V}_{\text{BERT}}, \mathbf{V}_{\text{ER}-\text{avg}}\right] \tag{4}$$

This process is repeated for each of the k texts in the dataset, resulting in Eq. (5):

$$\mathbf{V}^{k}_{\text{combined}} = \left[\mathbf{V}^{k}_{\text{BERT}}, \mathbf{V}^{k}_{\text{ER}-\text{avg}}\right] \tag{5}$$

Here, $\mathbf{V}^{k}_{\text{BERT}}$ represents the n-dimensional BERT vector representation of the kth text, and $\mathbf{V}^{k}_{\text{ER}-\text{avg}}$ denotes the average m-dimensional entity-relation vector for the kth text. Each combined $\mathbf{V}^{k}_{\text{combined}}$ thus represents the $n + m$ dimensional combined vector representation for the kth text in the dataset.

Consequently, by combining the BERT representation of each text with the average entity-relationship vector, an $n + m$ dimensional combined vector representation is obtained for each text. This combined representation encompasses both the linguistic features and the entity-relationship information of the text, enabling its application in various natural language processing tasks. This approach allows for a deeper understanding of inter-textual relationships, semantic similarities, and differences. Especially when working with large datasets, such combined vector representations offer a powerful tool for modeling the complex structures and relationships of texts.

3.5 Classification

Texts enriched with entity relationships were transformed into vector forms using pre-trained BERT models. In the final stage, these vector forms undergo text classification using machine learning or deep learning classification models. For the classification process, the SVM (Support Vector Machines) and Logistic Regression classification models from the Scikit-Learn library, a Python machine learning toolkit, were utilized. Additionally, text classification was performed using Deep Neural Network (DNN) and Convolutional Neural Network (CNN) models with the help of Keras (TensorFlow), a Python-based deep learning library. The performance of the classification was measured using commonly used metrics such as Accuracy, F1-score, Precision, and Recall.

Support Vector Machine (SVM), developed by Vapnik [52], is a machine learning model used for classification and regression tasks. SVM optimizes decision boundaries in high-dimensional feature spaces by focusing on an effective representation of the dataset [53]. Logistic Regression, on the other

hand, is a classification method used to predict the probability of data belonging to a specific class. It expresses the results as probability values using the sigmoid function [54].

Deep Neural Network (DNN) is a model used in machine learning and artificial intelligence. Inspired by biological neural networks, this model consists of multi-layer structures that process input data to produce output. These networks create data-driven models by adjusting weights and bias values through the learning process. Deep neural networks can solve complex problems and often perform better on large datasets. They are used in various fields such as image and speech recognition, natural language processing, and also in text classification tasks [55].

Convolutional Neural Network (CNN) is a deep learning model used for text classification and other tasks. It is an effective type of neural network that utilizes convolutional, pooling, and fully connected layers. The convolutional layers apply filtering operations to find specific features in text data. Pooling layers reduce the feature maps while preserving important information. Fully connected layers compute the results for classification. CNN is a powerful tool for detecting and classifying patterns in texts. It achieves successful results in text-based applications such as natural language processing, sentiment analysis, and spam filtering [56].

The primary reason for preferring classification models like SVM, CNN, and DNN over deep learning models such as BERT is their suitability for specific requirements and resource constraints. Classification models like SVM are generally less complex and, in some cases, make the interpretation of the model's decisions easier. Specifically, these models have been utilized in this study for literature comparison in classification tasks. This choice is based on factors such as their good performance with small datasets, lower requirements for updates and maintenance, and more interpretable results.

4 Experimental Results

In the study, the text preprocessing steps described in Section 3.2 were carried out on the Turkish and English text classification datasets in the literature introduced in Section 3.1. Then, with the steps shown in detail in Section 3.3, the task of finding the entities and relationships in the texts was carried out. The texts in the datasets are enriched by adding the identified entities with the help of pre-trained NER models and the relationships obtained from Wikidata related to these entities. A BERT-based pre-trained model was used as the vector model, and SVM, Logistic Regression, Deep Neural Network (DNN), and Convolutional Neural Network (CNN) models were used as the classification algorithms. The accuracy performance of each classifier, along with the improvements achieved, is summarized below.

The assessment of the TTC-4900 dataset, as detailed in Table 4, underscores the significant enhancements that the integration of relational information can bring to the performance of various classifiers in classification processes. Confusion matrixes, where classification performances performed on the Turkish TTC-4900 dataset are displayed separately based on labels are shown in Fig. 5.

The Linear SVM classifier, for instance, achieved a notable accuracy rate of 91.3% without considering relations. This rate modestly increased to 92.0% upon the inclusion of relations, marking an improvement of 0.77%. In contrast, the CNN classifier saw a substantial jump in performance, from an accuracy rate of 86.0% without relations to 89.1% with them, translating into a significant gain of 3.61%. Similarly, the DNN classifier improved from an accuracy rate of 91.3% without relations to 92.1% with them, indicating an enhancement of 0.88%. A noteworthy observation was also made for

the Logistic Regression classifier. It started with an accuracy rate of 87.8% without relations and rose to 90.8% with the inclusion of relations, resulting in a notable improvement of 3.42%.

Table 4: Relational text classification results for TTC-4900 dataset

	Without relations				With relations				Accuracy
Classification	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Improvement (%)
LogisticReg.	88.2	87.8	87.9	**87.8**	90.8	90.8	90.8	**90.8**	**3.42**
LinearSVM	91.4	91.3	91.3	**91.3**	92.0	92.0	92.0	**92.0**	**0.77**
CNN	86.8	86.0	86.0	**86.0**	89.2	89.1	89.1	**89.1**	**3.61**
DNN	91.4	91.3	91.3	**91.3**	92.1	92.1	92.1	**92.1**	**0.88**

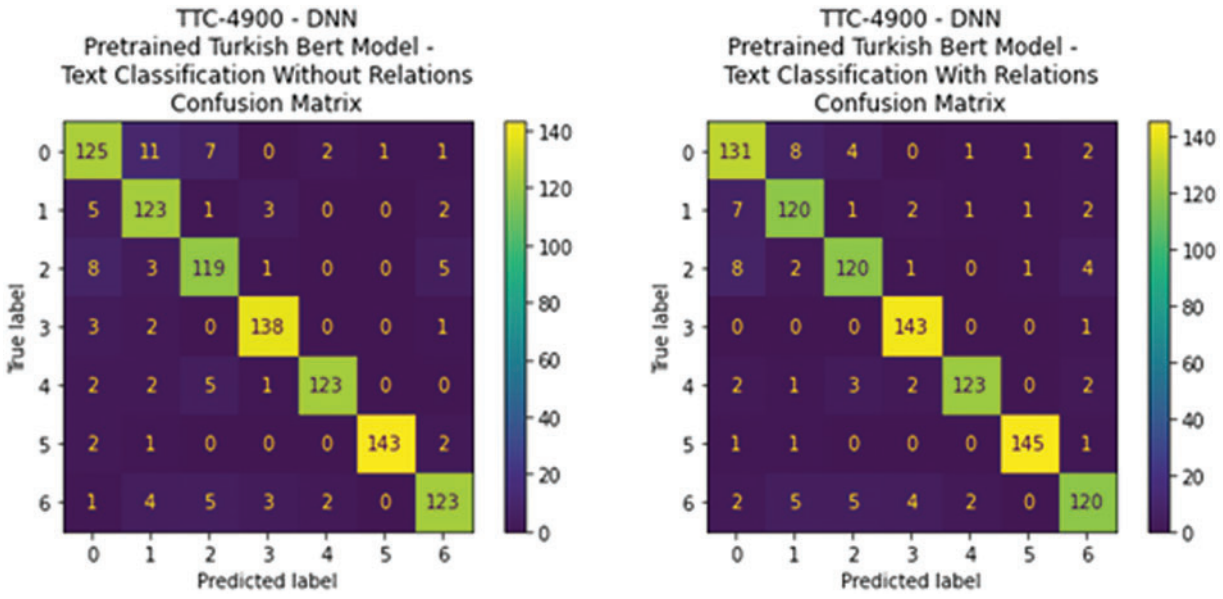

Figure 5: Text classification confusion matrices for TTC-4900 dataset

This highlights how relational information can substantially increase the accuracy and efficacy of models in classification tasks. These findings provide valuable insights into how the integration of relational information can enhance the understanding and performance of various models in classification processes.

The inclusion of relationships led to significant improvements in performance, particularly for the Linear SVM, CNN, and DNN classifiers. The Linear SVM classifier demonstrated a noteworthy increase in accuracy by incorporating relationships. Additionally, both the CNN and DNN classifiers exhibited substantial performance enhancements when relationships were considered.

In Table 5, moving on to the BBC News dataset, the Linear SVM classifier achieved an accuracy rate of 94.2% without considering relationships. However, after including relationships, the accuracy significantly increased to 98.7%, resulting in a notable improvement of 4.78%. Similarly, the CNN

classifier exhibited an accuracy rate of 89.7% without relationships, which improved to 96.7% with the inclusion of relationships, indicating a substantial performance boost of 7.81%. Unlike the previous classifiers, the DNN classifier did not exhibit any improvement despite achieving a high accuracy rate of 97.6% and 97.8% with or without relationships.

Table 5: Relational text classification results for BBC-News dataset

	Without relations				With relations				Accuracy
Classification	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Improvement (%)
LogisticReg.	83.2	77.6	75.6	**77.6**	97.7	97.6	97.6	**97.6**	**25.78**
Linear SVM	94.4	94.2	94.1	**94.2**	98.7	98.7	98.7	**98.7**	**4.78**
CNN	90.1	89.7	89.6	**89.7**	96.7	96.7	96.7	**96.7**	**7.81**
DNN	97.6	97.6	97.6	**97.6**	97.8	97.8	97.8	**97.8**	**0.21**

Lastly, in Table 6, for the 20Newsgroups dataset, the Linear SVM classifier achieved an accuracy rate of 78.8% without considering relationships. However, by incorporating relationships, the accuracy increased significantly to 96.9%, resulting in a remarkable improvement of 22.97%. The CNN classifier had an accuracy rate of 73.3% without relationships, which improved to 91.5% with the inclusion of relationships, representing a substantial increase of 24.83% in performance.

Table 6: Relational text classification results for 20 Newsgroups dataset

	Without relations				With relations				Accuracy
Classification	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Improvement (%)
LogisticReg.	80.0	76.5	73.3	**76.5**	95.9	95.8	95.8	**95.8**	**25.23**
Linear SVM	80.0	78.8	78.3	**78.8**	96.9	96.9	96.9	**96.9**	**22.97**
CNN	75.1	73.3	73.3	**73.3**	91.6	91.5	91.5	**91.5**	**24.83**
DNN	90.6	90.5	90.5	**90.5**	96.6	96.6	96.6	**96.6**	**6.75**

The study compares our findings with existing research on Turkish and English text classification datasets found in the literature. Our research followed predefined text preprocessing steps and methods for identifying entities and relationships, deeply examining the impact of enriched text information on classification performance. During this process, we utilized a BERT-based pre-trained model and various classification algorithms such as SVM, Logistic Regression, DNN, and CNN. We observed that the integration of relational information significantly enhanced classification accuracy. Experiments conducted on the TTC-4900, BBC-News, and 20Newsgroups datasets demonstrated the contribution of relational information.

These results emphasize the importance of relational information, especially in languages with limited resources. Our research proves that text classification tasks can improve model performance across different languages. When compared with literature studies on English text sets, the success in

improving accuracy and recall rates with relational information showcases the universality and applicability of this approach [16,21,22]. Overall, these results demonstrate the differences in how various classifiers process and benefit from relational information. The inclusion of relational information generally enhances performance, but the degree of improvement depends on the characteristics of the model used and how it integrates relational information. As a result, this study demonstrates that relational information plays a critical role in enhancing text classification performance across various languages, including Turkish and English.

4.1 Performance Comparison

The text classification performance of our proposed system is compared with models summarized in the literature. While Kılınc et al. [15] and Kuyumcu et al. [18] reported the highest accuracy rates obtained with Random Forest and Multinomial NB classification models as 91.03% and 90% respectively on the TTC-3600 dataset, our system surpassed this performance with a 92.1% accuracy using DNN on the same dataset. Parlak et al. [19] achieved a 78.1% F1-score on the TTC-3600 dataset, whereas our system achieved a 92.1% F1-score with DNN. Similarly, Pittaras et al. [21] and Lezama et al. [22] reported accuracy rates of 82.6% and 79% respectively for the 20Newsgroups dataset, while our system achieved a 96.6% accuracy with DNN, surpassing these results. This comparison highlights the superior performance of our proposed system across different datasets compared to the models reported in the literature. The improvements we achieved indicate that the use of relational information marks significant progress in the field of natural language processing.

5 Conclusion

The research conducted on relational text classification, according to the findings of this study, has observed that the inclusion of entity-relation information in texts enhances classification performance when using machine learning and deep learning models. The inclusion of entity relations in texts provides valuable information and context that enables the models to better understand and classify the text. The study primarily focused on person entities and selected a limited number of attribute values for these entities. This selective approach allowed for a more customized analysis of the impact of entity relations on the classification performance. The identification of relationships in the dataset led to the creation of an entity-relation knowledge base, which was utilized in the classification process.

This research, which builds on these foundational insights, demonstrates the critical role of entity-relation integration in enhancing text classification performance across various languages, including Turkish and English. Specifically, this integration led to a significant accuracy improvement of 3.42% in Turkish texts, showcasing how the morphological richness and agglutinative structure of Turkish can enable a deeper analysis of entity-relation information to boost classification performance. In English texts, the simplicity of syntactic structure and the availability of extensive natural language processing tools facilitated entity-relation integration, resulting in notable improvements up to 25.78%. These differences highlight the importance of considering the structural characteristics of languages in the development and application of relational text classification models.

In conclusion, the structural and morphological features of languages must be considered in developing and applying text classification models. Adopting modeling techniques and approaches that align with the characteristics of different languages, such as Turkish and English, is key to optimizing classification performance and increasing model accuracy. This research underscores the necessity for further research and development efforts to overcome current limitations and develop more effective and reliable relational text classification systems. Our research has concentrated on how

text classification models can be enhanced by considering linguistic features. Accordingly, our study focuses on Turkish text classification, which, due to the unique linguistic structures and complexities, requires specific approaches beyond the direct application of general-purpose generative models. In this context, how models such as Llama2 and GPT-4 can be adapted to specific domains and how the effectiveness of this adaptation can be measured is an important research topic. This work emphasizes how the integration of entity-relationship knowledge can improve the performance of models in such specific domains.

Future studies should focus on the effective integration of the morphological and syntactic characteristics of languages, the customization of entity-relation integration strategies specific to languages, and the development and application of innovative natural language processing techniques. In this context, examining the performance of relational text classification in different languages and cultural contexts, enabling models to better learn complex relational structures through enriched data sets, effectively modeling the complexity of entities and relations in texts with graph-based models and attention mechanisms, and developing interactive learning systems based on user feedback to improve the classification of ambiguous or contentious relations are crucial. These directions aim to achieve further advancements in natural language processing and text classification, contributing to the development of more effective and reliable systems.

Acknowledgement: None.

Funding Statement: The authors confirm that this study did not receive any specific funding.

Author Contributions: The authors confirm their contributions to the paper as follows: Study conception and design: Halil Ibrahim Okur, Kadir Tohma, Ahmet Sertbas; data collection: Halil Ibrahim Okur, Ahmet Sertbas; analysis and interpretation of results: Ahmet Sertbas, Kadir Tohma, Halil Ibrahim Okur; draft manuscript preparation: Halil Ibrahim Okur, Ahmet Sertbas. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data used in this study are not made publicly available due to their intended use in future research. However, access to the data can be obtained through the corresponding author upon a reasonable request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] H. Zhu, H. Peng, Z. Lyu, L. Hou, J. Li and J. Xiao, "Pre-training language model incorporating domain-specific heterogeneous knowledge into a unified representation," *Expert. Syst. Appl.*, vol. 215, no. 4, pp. 119369, 2023. doi: 10.1016/j.eswa.2022.119369.

[2] H. Ahonen, O. Heinonen, M. Klemettinen, and A. I. Verkamo, "Applying data mining techniques in text analysis" in *Report C-1997-23*. Dept. of Computer Science, University of Helsinki, Finland, 1997.

[3] C. H. Mooney and J. F. Roddick, "Sequential pattern mining–approaches and algorithms," *ACM Comput. Surv.*, vol. 45, no. 2, pp. 1–39, 2013. doi: 10.1145/2431211.2431218.

[4] J. Achiam *et al.*, "GPT-4 technical report," arXiv preprint arXiv:2303.08774, 2023.

[5] H. Touvron *et al.*, "Llama 2: Open foundation and fine-tuned chat models," arXiv preprint arXiv:2307.09288, 2023.

[6] W. Cunha, F. Viegas, C. França, T. Rosa, L. Rocha and M. A. Gonçalves, "A comparative survey of instance selection methods applied to nonneural and transformer-based text classification ," *ACM Comput. Surv.*, vol. 55, no. 265, pp. 152, 2023.
[7] P. K. Roy, S. Saumya, J. P. Singh, S. Banerjee, and A. Gutub, "Analysis of community question-answering issues via machine learning and deep learning: State-of-the-art review," *CAAI Trans. on Intel. Tech.*, vol. 8, no. 1, pp. 95–117, 2023. doi: 10.1049/cit2.12081.
[8] M. R. Hossain, M. M. Hoque, N. Siddique, and M. A. A. Dewan, "AraCovTexFinder: Leveraging the transformer-based language model for Arabic COVID-19 text identification," *Eng. Appl. Artif. Intel.*, vol. 133, no. 1, pp. 107987, 2024. doi: 10.1016/j.engappai.2024.107987.
[9] M. R. Hossain, M. M. Hoque, and N. Siddique, "Leveraging the meta-embedding for text classification in a resource-constrained language," *Eng. Appl. Artif. Intel.*, vol. 124, no. 1, pp. 106586, 2023. doi: 10.1016/j.engappai.2023.106586.
[10] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu and J. Gao, "Deep learning-based text classification: A comprehensive review," *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–40, 2021.
[11] X. Luo, "Efficient english text classification using selected machine learning techniques," *Alex. Eng. J.*, vol. 60, no. 3, pp. 3401–3409, 2021. doi: 10.1016/j.aej.2021.02.009.
[12] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," presented at the Adv. Neural Inf. Process. Syst., 2013.
[13] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proc. 2014 Conf. Emp. Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[14] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes and D. Brown, "Text classification algorithms: A survey," *Information*, vol. 10, no. 4, pp. 150, 2019. doi: 10.3390/info10040150.
[15] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.
[16] T. N. I. Fitria, "Artificial intelligence (AI) technology in OpenAI ChatGPT application: A review of ChatGPT in writing english essay," *ELT Forum: J. English Lang. Teach.*, vol. 12, no. 1, pp. 44–58, 2023. doi: 10.15294/elt.v12i1.64069.
[17] A. Gasparetto, M. Marcuzzo, A. Zangari, and A. Albarelli, "A survey on text classification algorithms: From text to predictions," *Information*, vol. 13, no. 2, pp. 83, 2022. doi: 10.3390/info13020083.
[18] F. Fkih, M. Alsuhaibani, D. Rhouma, and A. M. Qamar, "Novel machine learning-based approach for Arabic text classification using stylistic and semantic features., "*Comput. Mater. Contin.*, vol. 75, no. 3, pp. 5871–5886, 2023. doi: 10.32604/cmc.2023.035910.
[19] M. M. Mirończuk and J. Protasiewicz, "A recent overview of the state-of-the-art elements of text classification," *Expert. Syst. Appl.*, vol. 106, no. 3, pp. 36–54, 2018. doi: 10.1016/j.eswa.2018.03.058.
[20] D. Kılınc, A. Ozcift, F. Bozyigit, P. Yıldırım, F. Yucalar and E. Borandag, "TTC-3600: A new benchmark dataset for Turkish text categorization," *J. Inf. Sci.*, vol. 43, no. 2, pp. 174–185, 2017. doi: 10.1177/0165551515620551.
[21] J. Flisar and V. Podgorelec, "Document enrichment using DBPedia ontology for short text classification," presented at the Proc. 8th Int. Conf. Web Intell., Min. Semant., Novi Sad, Serbia, 2018, pp. 1–9.
[22] B. Kuyumcu, C. Aksakalli, and S. Delil, "An automated new approach in fast text classification (fastText): A case study for Turkish text classification without pre-processing," presented at the Proc. 2019 3rd Int. Conf. Natural Lang. Process. Inform. Retriev, Tokushima, Japan, Jun. 2019, pp. 1–4.
[23] B. Parlak, "The effects of preprocessing on Turkish and English news data," *Sakarya Univ. J. Comput. Inform. Sci.*, vol. 6, no. 1, pp. 59–66, 2023. doi: 10.35377/saucis...1207742.
[24] Ç. Aci and A. Çırak, "Turkish news articles categorization using convolutional neural networks and Word2Vec," (in Turkish), *Bilişim Teknolojileri Dergisi*, vol. 12, no. 3, pp. 219–228, 2019. doi: 10.17671/gazibtd.457917.
[25] N. Pittaras, G. Giannakopoulos, G. Papadakis, and V. Karkaletsis, "Text classification with semantically enriched word embeddings," *Nat. Lang. Eng.*, vol. 27, no. 4, pp. 391–425, 2021. doi: 10.1017/S1351324920000170.

[26] A. L. Lezama-Sánchez, M. T. Vidal, and J. A. Reyes-Ortiz, "An approach based on semantic relationship embeddings for text classification," *Mathematics*, vol. 10, no. 21, pp. 4161, 2022. doi: 10.3390/math10214161.
[27] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou and T. Mikolov, "Fasttext.zip: Compressing text classification models," arXiv preprint arXiv:1612.03651, 2016.
[28] T. Rebele, F. Suchanek, J. Hoffart, J. Biega, E. Kuzey and G. Weikum, "YAGO: A multilingual knowledge base from wikipedia, wordnet, and geoNames," presented at the Int. Semantic Web Conf., Cham, Springer, Oct. 2016, pp. 177–185.
[29] K. Bollacker, R. Cook, and P. Tufts, "Freebase: A shared database of structured general human knowledge," in *AAAI*, vol. 7, Jul. 2007, pp. 1962–1963.
[30] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer, "DBpedia spotlight: Shedding light on the web of documents," presented at the Proc. 7th Int. Conf. Semantic Syst., Graz, Austria, Sep. 2011, pp. 1–8.
[31] D. Vrandečić and M. Krötzsch, "Wikidata: A free collaborative knowledgebase," *Commun. ACM*, vol. 57, no. 10, pp. 78–85, 2014. doi: 10.1145/2629489.
[32] N. Zhang *et al.*, "Document-level relation extraction as semantic segmentation," arXiv preprint arXiv:2106.03618, 2021.
[33] A. Smirnova and P. Cudré-Mauroux, "Relation extraction using distant supervision: A survey," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–35, 2018.
[34] Y. M. Shang, H. Huang, X. Sun, W. Wei, and X. L. Mao, "A pattern-aware self-attention network for distant supervised relation extraction," *Inform. Sci.*, vol. 584, no. 8, pp. 269–279, 2022. doi: 10.1016/j.ins.2021.10.047.
[35] K. Yang, L. He, X. Dai, S. Huang, and J. Chen, "Exploiting noisy data in distant supervision relation classification," in *Proc. 2019 Conf. North American Chapter Assoc. Computat. Linguist.: Human Lang. Technol.*, vol. 1, pp. 3216–3225, Jun. 2019. doi: 10.18653/v1/N19-1.
[36] M. Tang, B. Yang, and H. Xu, "A multi-grained attention network for multi-labeled distant supervision relation extraction," presented at the 2021 IEEE 6th Int. Conf. Comput. Commun. Syst. (ICCCS), IEEE, Chengdu, China, Apr. 2021, pp. 110–115.
[37] S. Schweter, "BERTurk-BERT models for Turkish," *Zenodo*, Apr. 2020. doi: 10.5281/zenodo.3770924.
[38] K. Lang, "The 20 newsgroups data set," 2008. Accessed: Feb. 19, 2024. [Online]. Available: http://people.csail.mit.edu/jrennie/20Newsgroups/
[39] BBC News, "Collected by BBC News Website & Homepage," 2006, Accessed: Feb. 19, 2024. http://mlg.ucd.ie/datasets/bbc.html
[40] M. F. Porter, "Snowball: A language for stemming algorithms," 2001. Accessed: Feb. 19, 2024. [Online]. Available: http://snowball.tartarus.org/texts/introduction.html
[41] J. Perkins, "Python text processing with NLTK 2.0 cookbook," 2010. Accessed: Feb. 19, 2024. [Online]. Available: https://karczmarczuk.users.greyc.fr/TEACH/TAL/Doc/python_text_processing_with_nltk_2.0_cookbook.pdf
[42] A. A. Akin and M. D. Akin, "Zemberek, an open source NLP framework for Turkic languages," *Structure*, vol. 10, pp. 1–5, 2007.
[43] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel, "OntoNotes: The 90% solution," in *Proc. Human Lang. Technol. Conf. NAACL*, Jun. 2006, pp. 57–60.
[44] A. Ghaddar and P. Langlais, "WINER: A wikipedia annotated corpus for named entity recognition," presented at the Proc. Eighth Int. Joint Conf. Natural Lang. Process., Nov. 2017, pp. 413–422.
[45] O. T. Yıldız, K. Ak, G. Ercan, O. Topsakal, and C. Asmazoğlu, "A multilayer annotated corpus for Turkish," presented at the 2018 2nd Int. Conf. Natural Lang. Speech Process. (ICNLSP), IEEE, Apr. 2018, pp. 1–6.
[46] G. A. Şeker and G. Eryiğit, "Extending a CRF-Based named entity recognition model for Turkish well-formed text and user-generated content," *Semant. Web*, vol. 8, no. 5, pp. 625–642, 2017. doi: 10.3233/SW-170253.

[47] D. Zelenko, C. Aone, and A. Richardella, "Kernel methods for relation extraction," *J. Mach. Learn. Res.*, vol. 3, pp. 1083–1106, Feb. 2003.
[48] M. E. Z. N. Kambar, A. Esmaeilzadeh, and M. Heidari, "A survey on deep learning techniques for joint named entities and relation extraction," presented at the 2022 IEEE World AI IoT Cong. (AIoT), IEEE, Jun. 2022, pp. 218–224.
[49] X. Pan, B. Zhang, J. May, J. Nothman, K. Knight and H. Ji, "Cross-lingual name tagging and linking for 282 languages," presented at the Proc. 55th Annual Meet. Assoc. Comput. Linguist., Jul. 2017, pp. 1946–1958.
[50] X. Wang *et al.*, "KEPLER: A unified model for knowledge embedding and pre-trained language representation," *Trans. Assoc. Comput. Linguist.*, vol. 9, no. 11, pp. 176–194, 2021. doi: 10.1162/tacl_a_00360.
[51] G. Aras, D. Makaroğlu, S. Demir, and A. Çakır, "An evaluation of recent neural sequence tagging models in Turkish named entity recognition," *Expert. Syst. Appl.*, vol. 182, pp. 115049, 2021. doi: 10.1016/j.eswa.2021.115049.
[52] V. Vapnik, *The nature of statistical learning theory*. New York, NY, USA: Springer Science & Business Media, 1999.
[53] L. Gunawan, M. S. Anggreainy, L. Wihan, G. Y. Lesmana, and S. Yusuf, "Support vector machine based emotional analysis of restaurant reviews," *Procedia Comput. Sci.*, vol. 216, no. 2, pp. 479–484, 2023. doi: 10.1016/j.procs.2022.12.160.
[54] S. U. Hassan, J. Ahamed, and K. Ahmad, "Analytics of machine learning-based algorithms for text classification," *Sustain. Operat. Comput.*, vol. 3, pp. 238–248, 2022. doi: 10.1016/j.susoc.2022.03.001.
[55] D. Magalhães, R. H. Lima, and A. Pozo, "Creating deep neural networks for text classification tasks using grammar genetic programming," *Appl. Soft Comput.*, vol. 135, no. 1, pp. 110009, 2023. doi: 10.1016/j.asoc.2023.110009.
[56] R. Sarasu, K. K. Thyagharajan, and N. R. Shanker, "SF-CNN: Deep text classification and retrieval for text documents," *Intell. Automat. Soft Comput.*, vol. 35, no. 2, pp. 1799–1813, 2023. doi: 10.32604/iasc.2023.027429.