Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

# Optimization of Arabic text classification using SVM integrated with word embedding models on a novel dataset

Abdulaziz M. Alayba *, Mohammed Altamimi

*Department of Information and Computer Science, College of Computer Science and Engineering, University of Ha'il, Ha'il 81481, Saudi Arabia*

ARTICLE INFO

*Article history:*
Received 10 March 2025
Received in revised form
14 July 2025
Accepted 14 August 2025

*Keywords:*
Arabic linguistics
Natural language processing
Text classification
Word embedding
Support vector machine

ABSTRACT

Arabic linguistics covers various areas such as morphology, syntax, semantics, historical linguistics, applied linguistics, pragmatics, and computational linguistics. The Arabic language presents major challenges for natural language processing (NLP) due to its complex morphological and semantic structure. In text classification tasks, effective feature selection is essential, and word embedding techniques have recently proven successful in representing textual data in a continuous vector space, capturing both semantic and morphological relationships. This study introduces a new, balanced Arabic text dataset for classification and examines the performance of combining word embedding models (Word2Vec, GloVe, and fastText) with a Support Vector Machine (SVM) classifier. The approach converts dense vector representations of Arabic text into single-value features for SVM input. Experimental results show that this method significantly outperforms the benchmark Term Frequency–Inverse Document Frequency (TF-IDF) approach, offering more accurate and reliable classification by effectively capturing Arabic contextual information.

## 1. Introduction

The Arabic language has recently emerged as one of the predominant areas of research in the field of natural language processing (NLP) (Darwish et al., 2021). This increasing consideration of NLP research interest is due to the rapid increase in digital textual data available in Arabic. Moreover, it is one of the most widely spoken languages globally because of its importance and position among Muslims, owing to the holy Quran. Many techniques and tools are being developed and explored to fortify the position of Arabic NLP as a medium of communication and automated supportive tools to process large volumes of textual data. One of the main aspects of NLP is Arabic text classification or categorization (Elarnaoty and Farghaly, 2018). This process involves the automatic assignment of each document to its predefined label or class based on the content. The benefits of text classification are crucial for various applications, including document automation (Sebastiani, 2002), large text filtering (Labani et al., 2018), sentiment analysis (Alayba et al., 2018), spam detection (Ahmed et al., 2018), abuse identification (Chou et al., 2008), topic modeling (Neogi et al., 2020), dialect identification (Altamimi and Teahan, 2019), and information retrieval (Khan et al., 2018), among various other uses.

In supervised learning (Cunningham et al., 2008), text classification is performed by converting text into numerical values using various text feature extraction methods. Subsequently, these features are input into an algorithm for training classification purposes, which assigns each document to its corresponding category (Alayba, 2019). The discerning and interpretation of the context of the text present unique challenges in computational techniques, mainly due to the complexity of the linguistic structures, the diversity of morphology, and the varying forms of a single word. Therefore, effective feature extraction that captures semantic details and a robust machine learning algorithm are essential components of successful document categorization. Text feature extraction is the intermediary layer between raw text data and the classifier, ensuring that the text is appropriately represented for machine learning algorithms.

This paper proposes a novel, largest balanced Arabic dataset comprising 90,000 articles for

* Corresponding Author.
Email Address: a.alayba@uoh.edu.sa (A. M. Alayba)

https://doi.org/10.21833/ijaas.2025.09.013

Corresponding author's ORCID profile:
https://orcid.org/0000-0003-1075-1214

classification tasks and a state-of-the-art model for Arabic text classification. This dataset is derived from existing literature (Altamimi and Alayba, 2023), adhering to specific criteria for selecting articles to construct a dataset of nine categories, each category comprising 10,000 articles. In addition, the proposed model amalgamates word embedding techniques (Selva Birunda and Kanniga Devi, 2021) and Support Vector Machine (SVM). The efficacy of the model is derived from the durability of word embedding techniques (Khanal et al., 2020) as well as the classification capabilities of SVM (Trafalis and Gilbert, 2007). Different word embedding approaches have varying advantages depending on the technique used. For example, Word2Vec effectively captures semantic information (Le and Mikolov, 2014; Mikolov et al., 2013), the GloVe algorithm considers both local context and global statistical information (Pennington et al., 2014), and fastText effectively handles subwords for morphological information (Bojanowski et al., 2017). We adopt vectors from word embedding models suitable for Arabic text and apply them to the SVM algorithm using a singular numeric value, such as mean, median (Bickel, 2003), maximum, and minimum. Moreover, many studies have proved the efficiency of the SVM algorithm in classification techniques (Trafalis and Gilbert, 2007), owing to its advantages, such as efficiency in high-dimensional spaces (Ghaddar and Naoum-Sawaya, 2018), robustness to overfitting (Han and Jiang, 2014), and a strong theoretical foundation (Wang, 2005). Furthermore, we compare the results of our proposed model with those of the word embedding models proposed in Alayba and Palade (2022).

The rest of the sections are structured as follows. Section 2 reviews all related studies on Arabic datasets and the field of Arabic text classification. In Section 3, a novel Arabic text classification dataset is proposed, and the method of collecting the data is illustrated in detail. Section 4 elaborates on the proposed model for text classification using word embedding techniques in combination with the SVM algorithm. The classification results acquired from over 70 experiments using the proposed model are discussed and compared in Section 5. A discussion of error analysis in the experiments with some case examples is detailed in Section 6. Section 7 presents the overall conclusions of the study and future work.

## 2. Literature survey

The English, French, and Spanish languages have been extensively explored for NLP in general and specifically for text classification. This review will focus on recent studies that discuss text classification for the Arabic language and explore the contributions of Arabic resources and corpora.

The initial methodologies for text classification have primarily focused on machine learning algorithms. A text classification system specifically designed for Arabic documents uses SVM and the chi-square technique for feature selection (Al-Harbi et al., 2008). This system processed a dataset comprising 1,445 Arabic online newspaper articles categorized into nine classes. It achieved the best results with an F1-measure of 90%, outperforming Naïve Bayes (NB) and K-Nearest Neighbors (KNN), which yielded 84% and 72% of F1-measure, respectively. Similarly, another method used SVM and C5.0 to investigate seven diverse Arabic corpora (Al-Harbi et al., 2008). The C5.0 classifier yielded the highest classification performance and an accuracy of 78%. A Naïve Bayes classifier for Arabic document classification was proposed in Noaman et al. (2010), and an experiment conducted on 300 documents classified into 10 categories achieved a classification accuracy of 62.23%. A comparative study examined KNN and SVM for Arabic text classification, using a corpus of news articles for training and testing. The SVM model yielded superior predictive performance (Hmeidi et al., 2008).

Furthermore, deep learning techniques for text classification were explored using the Term Frequency-Inverse Document Frequency (TF-IDF) method in conjunction with a convolutional neural network (CNN). This study conducted an experiment using 111,728 documents and achieved an accuracy of 92.94%. Alhawarat and Aseeri (2020) proposed a deep learning model called SATCDM, which combines word embedding with a Convolutional Neural Network (CNN), achieving classification accuracy ranging from 97.58% to 99.90%. Their experiments were conducted on a collection of news articles from various sources, including Abuaiadah, Al Jazeera, Al Watan, Al Khaleej, OSAC, BBC, CNN, NADA, Arabia, Khaleej, and Akhbarona. Rifai et al. (2021) performed a multi-label text classification approach for Arabic news articles collected from ten different Arabic websites. They evaluated several models, including Logistic Regression, XGBoost, CNN, and a hybrid CNN-LSTM model. The Convolutional Neural Network (CNN) achieved the best performance, with an accuracy of 94.21%, surpassing all other classifiers tested in the study. In addition, deep learning models using CNN and word2vec were employed to generate vectors for words, yielding improved classification accuracy (Abou Khachfeh et al., 2021). The performance of the classifier model was observed to improve with increasing dataset size.

Recently, Setu et al. (2024) presented a novel approach to improving Arabic news article classification by incorporating transformer-based models with data augmentation techniques. Specifically, the Bidirectional Encoder Representations from Transformers (AraBERT). The proposed model achieved 98% accuracy, addressing challenges associated with imbalanced Arabic classification. Furthermore, Akhadam and Ayyad (2024) compared various machine learning and deep learning techniques for Arabic text classification, such as Logistic Regression, Stochastic Gradient Descent (SGD), CNN, Bag of Words (BOW), and Term Frequency-Inverse Document Frequency (TF-IDF) representations. Their experiments, performed on the Al-Khaleej dataset, demonstrated that CNN models, particularly with word-level representation, achieved superior accuracy, reaching 97%.

With respect to contributions to Arabic resources, the study by Abdelali et al. (2005) presents a dataset that compiles approximately 100 articles every day from 11 Arabic publications, namely Ahram, Alraialaam, Alwatan, Aps, Assafir, Jazirah, Aorocco, Petra, Raya, Teshreen, and Uruklink,. This dataset focused on the MSA text extracted from online newspapers across various Arab countries to support researchers in machine translation, information retrieval, and other tasks related to Arabic language processing. Similarly, the Corpus of Contemporary Arabic (CCA) featured a diverse collection of online written texts encompassing arts, fiction, business, and science, complemented by spoken content sourced from radio and TV. This corpus contains over 1 million words and was specifically developed as a resource for learning Arabic as a foreign language and for related research tasks (Al-Sulaiti and Atwell, 2006). In addition, the International Corpus of Arabic (ICA) proposed 80 million words from an extensive variety of newspapers, all written in MSA (Alansary and Nagi, 2014). This corpus aims to collect articles from different sources such as newspapers, magazines, novels, books, online articles, and academic publications. Furthermore, the Open-Source Arabic Corpora (OSAC) was created by compiling approximately 18 million words across 22,429 articles gathered from various newswire websites, such as CNN and BBC, covering a wide array of topics including economics, education, astronomy, religion, sports, law, stories, health, and cooking recipes (Saad and Ashour, 2010).

Some Arabic text corpora mainly contain dialectal content. For example, the Khaleej corpus includes 5,120 articles divided into four categories: sports, local news, international news, and economy. Likewise, the Alwatan corpus contains 20,291 articles across six categories: religion, sports, culture, economy, international news, and local news. By contrast, the BBC and CNN corpora are dominated by Modern Standard Arabic (MSA). The BBC corpus consists of 4,763 articles classified into six categories: Arab news, world news, technology, sports, economics, and mixed topics. The CNN corpus follows a similar structure, with 5,070 articles divided into Arab news, world news, technology, economics, sports, and entertainment.

A more recent corpus comprised 720,000 articles from several Arabic news portals, covering eight categories: sport, art, health, economy, accidents, sciences, politics, and culture (Abou Khachfeh et al., 2021). Furthermore, the Single-labeled Arabic News Articles Dataset (SANAD) corpus contains 194,922 articles sourced from three new websites: Akhbarona, AlArabiya, and AlKhaleej (Einea et al., 2019). The articles are divided into seven categories: politics, culture, religion, medicine, sports, technology, and finance. Numerous Arabic researchers have developed Arabic corpora and performed experiments using various algorithms. However, most existing Arabic corpora are limited in size, often contain redundant articles, and exhibit class imbalances, considerably hindering classification accuracy (Abou Khachfeh et al., 2021). The following section discusses the proposed Arabic News Article Classification Dataset (ANACD) designed to address these challenges.

## 3. Arabic news article classification dataset (ANACD)

The proposed balanced dataset contains 90,000 Arabic news articles derived from the ANAD corpus (Altamimi and Alayba, 2023). The original ANAD includes over half a million Arabic articles sourced from 12 different Arab news resources and classified into 10 categories: sport, politics, economy, technology, local, art, car, health, tourism, and entertainment. The proposed ANACD is designed mainly for Arabic text classification tasks. It comprises all categories of ANAD except entertainment, with more than 10,000 articles in each category. To the best of our knowledge, ANACD is the most comprehensive and well-balanced dataset available for Arabic text classification tasks. Therefore, for the new classification dataset, a target of 10,000 articles per category was set. The following criteria were established to construct the dataset for classification purposes:

All articles should maintain an analogous length, with each article ranging from 1 KB to 5 KB. However, due to a shortfall in the number of articles in the tourism category, the range was modified to be from 0.8 KB to 6 KB.

The title of each article must satisfy two requirements: it should consist of at least three words and contain a minimum of fifteen characters.

Considering the diverse writing styles of different newspapers for each category, an analysis was conducted after counting the articles that met the first two requirements. This was achieved by building a matrix and using statistical data to ensure a balanced distribution of articles among the categories and newspapers. For example, only three newspapers contain the politics category: Alarabiya, Alwatan, and BBC. The BBC has the fewest articles, totaling 216, all of which have been incorporated into ANACD. To achieve the target of 10,000 articles per category, an additional 9,784 articles are required, which are divided equally between Alarabiya and Alwatan. Table 1 indicates the average number of words per category for each dataset. Fig. 1 shows the statistical details for the distribution of articles among the newspapers and categories in ANACD.

All articles were randomly generated after setting the target number for each category with a specific newspaper. The naming convention of the text (txt) files follows two parts: "the name of the newspaper _ the name of the txt file in ANAD" (Altamimi and Alayba, 2023). The dataset is provided in two different formats. The first format consists of nine dictionaries representing the categories, with each dictionary containing 10,000 text files.

The second format is a csv file that contains six columns: the first column denotes the category or the class; the second denotes the length of the title, which is the number of characters in the article title, including spaces; the third column is the number of tokens in the title, which is the number of words separated by a space; the fourth column is the title of the articles; the fifth is the text or the body of the article; and the last column comprises the name of the text file.

**Table 1:** Distribution of articles per category and the average number of words per category for ANACD

| Category | Number of articles | Average count of words in titles/articles | Average count of words in text/articles | Average count words in combined (title and text)/articles |
|---|---|---|---|---|
| Art | 10000 | 8.6796 | 221.6867 | 230.3663 |
| Car | 10000 | 11.2001 | 194.7877 | 205.9878 |
| Economy | 10000 | 8.8938 | 216.9192 | 225.8130 |
| Health | 10000 | 8.1783 | 251.2082 | 259.3865 |
| Local | 10000 | 8.8380 | 188.3252 | 197.1632 |
| Politics | 10000 | 8.0228 | 221.7924 | 229.8152 |
| Sport | 10000 | 7.7955 | 189.7017 | 197.4972 |
| Technology | 10000 | 8.4327 | 238.8482 | 247.2809 |
| Tourism | 10000 | 9.3156 | 246.2350 | 255.5506 |

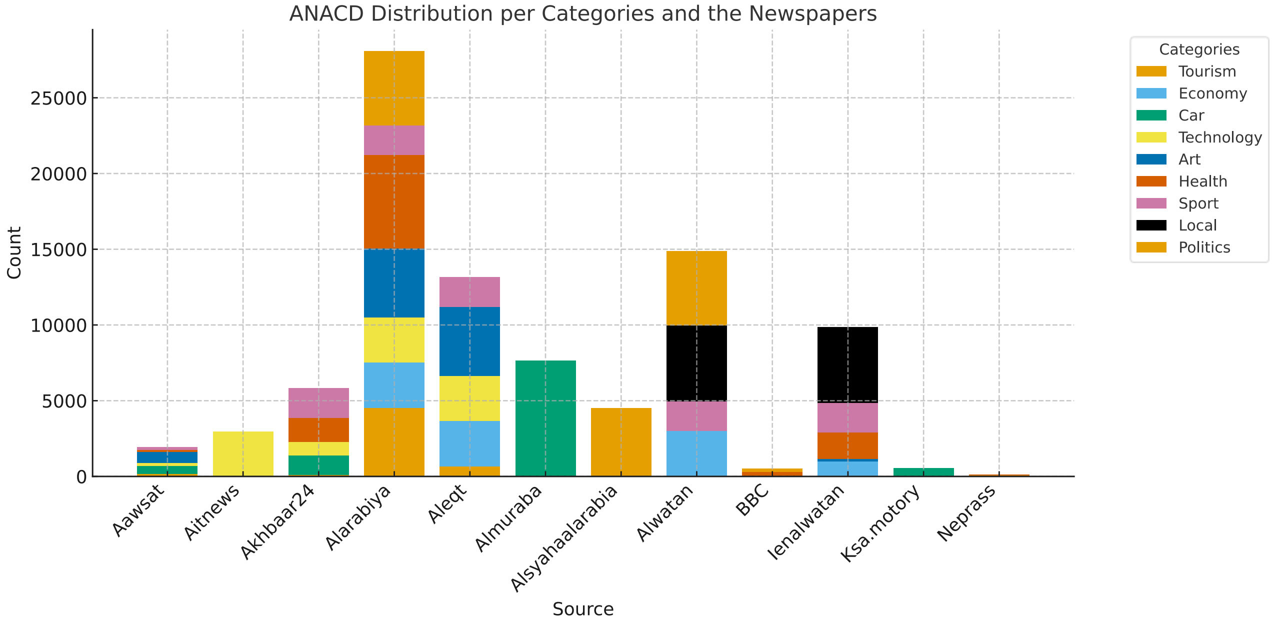

**Fig. 1:** Distribution of articles among the categories and newspapers in ANACD

The original ANACD dataset contains 20,488,357 words, including 937,333 unique words. In addition, another CSV file is prepared, where Arabic text preprocessing is applied to both the titles and the main text. This preprocessing includes removing punctuation, stop words, Arabic diacritics, non-Arabic characters, and digits, as well as normalizing Arabic characters. After cleaning and filtering, the processed ANACD dataset contains 16,744,274 words, of which 408,169 are unique.

## 4. Proposed Arabic text classification model

This section presents the various algorithms and methodologies employed in our experiments, detailing the components and configurations used in the proposed Arabic text classification model.

### 4.1. SVM

SVM is one of the most powerful machine learning algorithms. This supervised learning algorithm is commonly applicable for classification and regression tasks. It separates the datasets into classes by identifying the optimal separator hyperplane in a high-dimensional space. Moreover, it increases the margin between various points of the classes in the datasets, while the support vectors of the data points converge toward the hyperplane. The algorithm also supports different kernel functions, such as polynomial, sigmoid, Gaussian, linear, and non-linear (Ben-Hur and Weston, 2010; Alsaleem, 2011).

### 4.2. TF-IDF

TF-IDF serves as a method for text feature representation as a numeric value; it is applied in text mining and information retrieval tasks. It determines the statistical measurements of each word in a text in relation to the remaining words in the text. The weight assigned to each word is derived from a combination of term frequency (TF) and inverse document frequency (IDF), where TF is the appearance frequency of the word in the text and IDF denotes the reduction in the weight of words that appear with high frequency in the rest of the text. Therefore, a word with a low appearance frequency is assigned a higher weight, thereby identifying the uniqueness of the word in the text (Manning et al., 2008; Aizawa, 2003).

### 4.3. Word embedding

Word embedding is another technique for representing words as vectors of numeric values. This technique captures the semantic relationships between words by considering their contextual meaning. It maps words into a high-dimensional space where similar semantic words are positioned closer together and represented by adjacent vectors. These vectors are generated using an unsupervised learning technique, which uses neural networks or other statistical techniques along with large text corpora, to identify contextual relationships and word co-occurrences. Consequently, the technique represents the similarity between words based on the used corpora, which makes it an essential technique in NLP research and tasks. Many word embedding models exist, including Word2Vec (Mikolov et al., 2013), Glove (Pennington et al., 2014), fastText (Bojanowski et al., 2017), BERT (Devlin et al., 2019), ELMO (Peters et al., 2018), ULMFit (Howard and Ruder, 2018), and Transformer- XL (Joulin et al., 2017). This paper will focus on Word2Vec (Mikolov et al., 2013), Glove (Pennington et al., 2014), fastText (Bojanowski et al., 2017), and briefly describe them in the following subsections.

#### 4.3.1. Word2Vec

Word2Vec is a popular word embedding technique proposed by Mikolov et al. (2013). It comprises two main models: Continuous Bag of Words (CBOW) and Skip-gram (SG). The CBOW model predicts a target word from its neighboring context words within a specified window. This process involves three phases: the input layer, the hidden layer, and the output layer. The input layer considers the context of the word provided by the surrounding words. The hidden layer estimates the input words through a weight matrix, which is transmitted to the output layer.

The last step connects the output with the target word, which enhances word representation through error gradient backpropagation. In contrast, the SG model predicts the context words based on a specified target word. This process involves three phases: the input layer connected to the target word and the output layer that corresponds to the context. This model aims to estimate the context based on a given word, as opposed to the CBOW technique.

The final step of this model involves correlating the output with each word in the context to modify the representation through back propagation. Word2Vec generates vectors for all words in the corpus, where semantically similar words are positioned closer in the vector space. Its effectiveness and widespread application in the field of NLP tasks are due to its ability to capture complex linguistic features (Mikolov et al., 2013).

#### 4.3.2. GloVe

GloVe, which stands for Global Vectors for Word Representation, is another word embedding technique developed by a group of researchers from Stanford University led by Pennington et al. (2014). This technique generates word vectors through two primary steps. First, it aggregates global co-occurrence statistics of words from a large corpus to capture semantic meanings. Subsequently, it performs logarithmic factorization of the co-occurrence matrix to derive word vectors. Therefore, the dot product of the two vectors corresponds to the logarithm of the probability of their co-occurrence. This model efficiently integrates the advantages of matrix factorization techniques with local context, resulting in effective semantic word representations (Pennington et al., 2014).

#### 4.3.3. fastText

fastText is another word embedding model that is an extension of Word2Vec, developed by Facebook's AI Research (FAIR) team, led by Bojanowski et al. (2017). In contrast to Word2Vec, fastText considers words as bags of n-gram characters when inputting data into a neural network. This model efficiently identifies the morphological information of words by generating vectors for the sub-words through the summation of the n-gram embedding vectors, which together represent the whole word vector. fastText operates using the same two models and mechanisms as those of Word2Vec, namely CBOW and SG.

The process involves two main phases: First, each word is converted into a set of n-gram characters. Subsequently, procedures like those in Word2Vec's CBOW or SG are applied to learn embeddings by predicting context or words from these n-grams. This technique is effective in capturing morphological information, especially in rich languages (Bojanowski et al., 2017; Joulin et al., 2017).

### 4.4. BERT

It stands for Bidirectional Encoder Representations from Transformers, and it was introduced to enable bidirectional deep learning of language through transformer-based architecture (Devlin et al., 2019). It initiated an NLP revolution by considering a deep understanding of context from both directions simultaneously. It has a sufficient positive impact on a variety of language tasks, such as sentiment analysis, question answering, and other tasks. AreBERT is one of the transformer-based architectures pre-trained on a large Arabic corpus (Antoun et al., 2020). It has been trained on two large Arabic corpora, namely the 1.5 billion words Arabic Corpus (El-khair, 2016) and OSIAN: the Open-Source International Arabic News Corpus (Zeroual et al., 2019). AraBERT handles the complex morphology and syntactic in the Arabic language, and it has shown outperforming in Arabic NLP tasks compared to traditional word embeddings.

We exploit the capabilities of word embedding models for text feature extraction and the robustness of the SVM classifier. Fig. 2 illustrates the proposed model for Arabic text classification. In Fig. 2, "Text Row" data represents the filtered dataset used in the model. "Single Article" indicates the three approaches used in this study. These approaches focus on three distinct text inputs: the titles of the articles, the body texts of the articles, and a combination of both titles and body texts. "Text Feature Extraction" in Fig. 2 shows the proposed novel technique for extracting Arabic text features. We utilized word embedding models for Arabic text introduced in Alayba and Palade (2022), namely the Word2Vec CBOW model (W2V CBOW), Word2Vec SG model (W2V SG) (Mikolov et al., 2013), fastText SG model (fastText SG) (Bojanowski et al., 2017), and GloVe (Pennington et al., 2014). Each model was assessed using all three different vector sizes proposed in Alayba and Palade (2022). Furthermore, it is necessary to represent a vector using a single representative value that roughly characterizes the entire vector. Therefore, we applied various statistical measurements, including central tendency (mean, median, or mode) (Bickel, 2003), as well as the maximum and minimum values and standard deviation. The proposed methods for transforming a set of vectors into a single distinct value were tested; only two methods, i.e., mean and median, were acceptable, while the others were rejected due to inadequate feature representation. The eliminated techniques had considerably low classification accuracy because different words shared identical feature values.

"Text Feature Representation" in Fig. 2 reflects the mean or median values of the corresponding vectors in different word embedding models for each word in the Single Article. Finally, "SVM Classifier" specifies the SVM classifier used in this study, utilizing the Scikit Learn tool with a linear kernel to avoid overfitting while maintaining computational efficiency.

The classifier is trained on a part of the data and tested on the remaining part to evaluate the learning performance of the SVM model. The classifier automatically assembles the test data into one of the nine categories of ANACD.

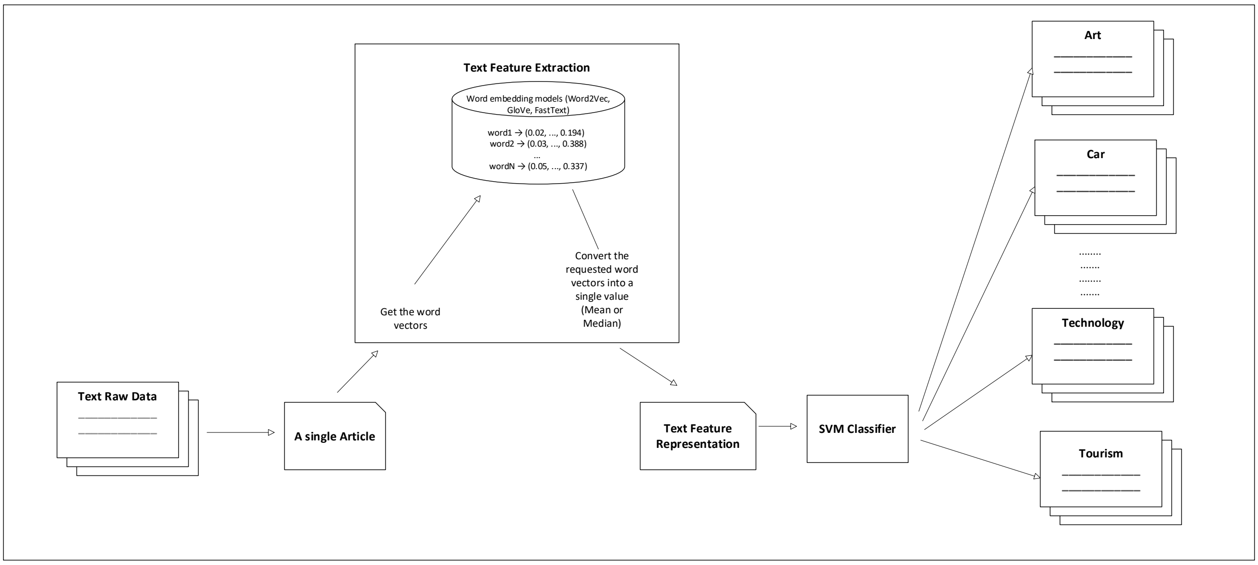
Text Feature Extraction
Word embedding models (Word2Vec, GloVe, FastText)
word1 → (0.02, ..., 0.194)
word2 → (0.03, ..., 0.388)
...
wordN → (0.05, ..., 0.337)
Get the word vectors
Convert the requested word vectors into a single value (Mean or Median)
Text Raw Data
A single Article
Text Feature Representation
SVM Classifier
Art
Car
........
Technology
Tourism

**Fig. 2:** A novel model for Arabic text feature extraction using word embedding approaches with SVM for Arabic text classification

Algorithm 1 describes in detail the classification process for ANACD. We apply the algorithms multiple times based on the following categories:

- Datasets (article title only, article body only, and combined full article title and body)
- Word embedding models (W2V CBOW, W2V SG, fastText SG, and GloVe)
- Word embedding vector sizes (100, 200, and 300)
- Corresponding single value of the vector (mean and median)

In total, 72 experiments were carried out using this algorithm, and the results are shown in Table 2 in the following section. For the dataset category, we modified the step "Extract columns" in Algorithm 1. We also adjusted the Word Embedding models and the vector sizes in the step "Load Pre-trained Word Embedding." The final change concerned the calculation of the mean or median, where we modified the method used in the steps "Compute mean vector v" and "Compute median vector v."

## 5. Experiments and results

This study proposed a feature extraction mechanism for Arabic text. It utilizes advanced text representation models of word embedding techniques and the robustness of the SVM classifier. The classification performance of Algorithm 1 was evaluated using the accuracy of the classifier (Aizawa, 2003), applying the cross-validation technique. Five-fold cross-validation was performed for all the experiments in this study, which divides the dataset into five parts (K); four out of the five parts were used for training, and the remaining part was used for validating the SVM classifier.

This process is iterated five times, ensuring that the validation parts are different in each iteration. One objective was to evaluate the practical accuracy of the classification model's performance. In addition, this approach can evaluate the quality of a fitted model and the consistency of its text features. Subsequently, the average accuracy of the five-fold cross-validation for each technique was calculated and is presented in Table 2. The accuracy was calculated using Eq. 1, followed by Eq. 2, for calculating the five-fold accuracy.

$$Accuracy_k = \frac{1}{n_k} \sum_{i=1}^{n_k} 1(\, y_i = \hat{y}_i \,) \qquad (1)$$

where, $n_k$ is the number of articles in the test set in fold $k$; $y_i$ is the true label for the $i$-th article; $\hat{y}_i$ is the predicted label for the $i$-th article; 1 (condition) represents an indicator function, which takes the value of 1 if the specified condition is true, and 0 otherwise.

**Algorithm 1:** Classifying ANACD

```
**Load Dataset:**
Load CSV file $D$
Extract columns $T=D[text]$ and $L=D[label]$
**Load Pre-trained Word Embedding Model:**
Load Pre-trained Word Embedding model $WE$
**Feature Extraction:**
Define function get_meanORmedian_$WE$($P$, $WE$, $k$):
If |P|=0:
Return zero vector $0_k$
Compute mean vector v = $\frac{1}{|P|} \sum_{\omega \epsilon P \cap WE} WE[\omega]$ OR compute median vector $v$
Return $v$
Convert each $\rho \epsilon P_T$ to feature vector $X$ using get_meanORmedian_$WE$ ($\rho$, $WE$, $k$)
**K-Fold Cross-Validation:**
Define $K$-fold cross-validation with $p\backslash l/$ folds
For each fold $i$ in 1 to $K$
Split $X$ and $L$ into training set ($X_{train}^{(i)}$ , $L_{train}^{(i)}$)and validation set ($X_{val}^{(i)}$ , $L_{val}^{(i)}$)
**Train SVM Classifier:**
Initialise SVM classifier SVM$^{(i)}$
Train SVM$^{(i)}$on ($X_{train}^{(i)}$ , $L_{train}^{(i)}$)
**Evaluate Classifier:**
Predict labels $L_{val}^{(i)}$ for $X_{val}^{(i)}$ using SVM$^{(i)}$
Calculate accuracy Ai of $L_{val}^{(i)}$
Compute average accuracy from all $K$ folds ($A_i$ )
Display *Accuracy* and classification report
```

For each fold k (where k = 1, 2, 3, 4, 5), we use fold k as the test set and the remaining four folds as the training set. Subsequently, we calculate the overall cross-validation accuracy as the average accuracy across all five folds for a multi-class classification problem with nine labels, as follows:

$$Accuracy_{cv} = \frac{1}{5} \sum_{k=1}^{5} (\, Accuracy_k \,) \qquad (2)$$

Before conducting the experiments using the proposed model, we used basic TF-IDF with SVM algorithms. The results for the three datasets (i.e., title only, body only, and combined title and body) were 0.7503, 0.8774, and 0.8783, respectively, as shown in Table 2. Table 2 also presents the performance of the SVM classifiers for the three datasets in comparison with those of different text feature extraction techniques. Fig. 3 illustrates the classification accuracy for the three datasets in comparison with that achieved using different text feature extraction techniques. The proposed model exhibited superior classification performance compared to the basic TF-IDF method.

Table 2 and Fig. 3 show that the classification accuracy for all three datasets (Title, Body, and Combined Title and Body) increased as the text length grew. The Title dataset alone provided limited textual features, resulting in higher semantic ambiguity. In contrast, the Body and Combined datasets offered richer context, which improved classification performance compared to the Title dataset.

The lowest performance across all datasets was observed when using smaller word embedding vector sizes, confirming their limitations in capturing deep semantic features. In contrast, 300-dimensional embeddings consistently outperformed other sizes across the models, as they provided richer input representations.

The results also showed that using the mean for feature extraction consistently produced higher accuracy than using the median, suggesting that the mean better preserves semantic information. The best classification performance among the three datasets (highlighted in bold and underlined in Table 2) was achieved with the W2V SG word embedding model, a 300-dimensional vector size, and mean-based feature extraction.

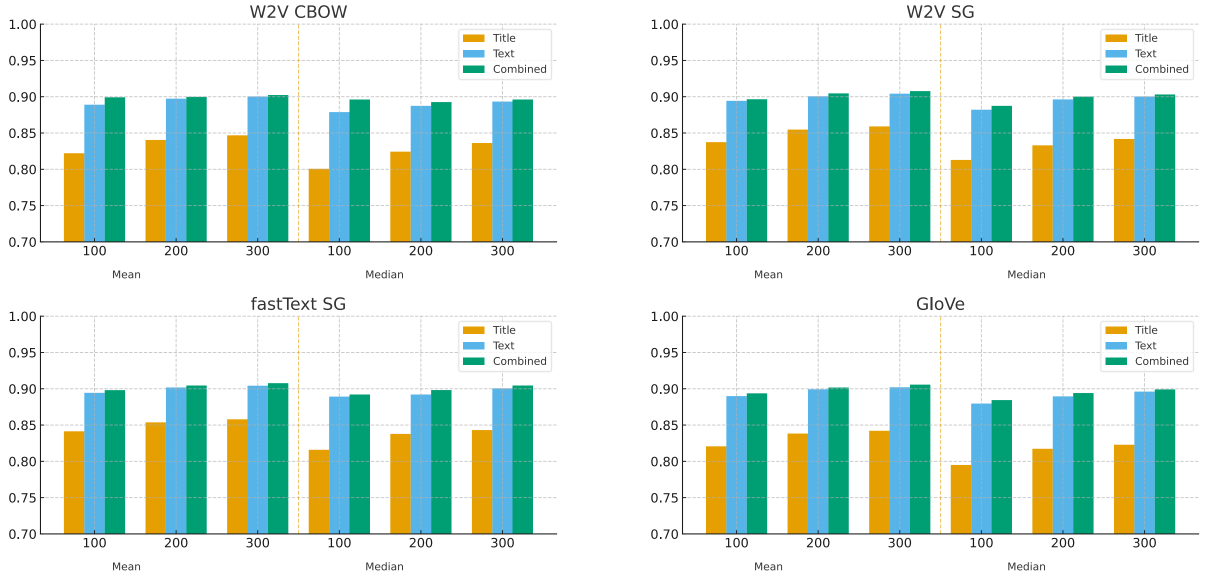

**Fig. 3:** Comparison of classification accuracy of the proposed model with different word embedding models

**Table 2:** Comparison of the classification accuracy of the proposed model and different word embedding techniques using varying word embedding sizes and mean and median values for three dataset categories

| Classifier model | | | Title | Text | Combined |
|---|---|---|---|---|---|
| BasicTFIDF | | | 0.7503 | 0.8774 | 0.8783 |
| AraBERT | | | 0.8804 | 0.9212 | 0.9268 |
| W2V CBOW | Mean | 100 | 0.8221 | 0.889 | 0.8992 |
| | | 200 | 0.8404 | 0.8973 | 0.8998 |
| | | 300 | 0.8468 | 0.9003 | 0.9023 |
| | Median | 100 | 0.8007 | 0.8788 | 0.896 |
| | | 200 | 0.8243 | 0.8874 | 0.8926 |
| | | 300 | 0.8362 | 0.8933 | 0.896 |
| W2V SG | Mean | 100 | 0.8374 | 0.8943 | 0.8965 |
| | | 200 | 0.8547 | 0.9006 | 0.9046 |
| | | 300 | **0.8591** | **0.9042** | **0.9077** |
| | Median | 100 | 0.8129 | 0.8821 | 0.8874 |
| | | 200 | 0.833 | 0.8963 | 0.9002 |
| | | 300 | 0.8418 | 0.9003 | 0.9031 |
| fastText SG | Mean | 100 | 0.8414 | 0.8944 | 0.8981 |
| | | 200 | 0.8537 | 0.9018 | 0.9045 |
| | | 300 | 0.8579 | **0.9042** | **0.9077** |
| | Median | 100 | 0.8159 | 0.8892 | 0.8921 |
| | | 200 | 0.8378 | 0.8921 | 0.8982 |
| | | 300 | 0.8431 | 0.9006 | 0.9044 |
| GloVe | Mean | 100 | 0.8207 | 0.8899 | 0.8937 |
| | | 200 | 0.8383 | 0.8993 | 0.9017 |
| | | 300 | 0.8421 | 0.9021 | 0.9057 |
| | Median | 100 | 0.795 | 0.8796 | 0.8844 |
| | | 200 | 0.8173 | 0.8895 | 0.8941 |
| | | 300 | 0.8229 | 0.8961 | 0.8991 |

The optimal classification performance among the three datasets, indicated in bold and underlined

The highest classification accuracies, 0.9042 and 0.9077, were achieved for the Body and Combined datasets using the fastText SG and W2V SG models, respectively, with a 300-dimensional vector size and mean-based extraction. These findings demonstrate the effectiveness of high-dimensional embeddings and show that combining title and body text leads to stronger Arabic text classification. They also indicate that such methods can serve as practical alternatives to transformer-based models in resource-limited settings.

Considering the state-of-the-art advancements in natural language processing, we selected AraBERT (Antoun et al., 2020) as a representative model specifically tailored for the Arabic language. The results obtained using AraBERT outperformed those of our proposed model, which is expected given AraBERT's advanced architecture and extensive pretraining. In comparison, our best-performing results were achieved using traditional word embedding techniques such as fastText SG and W2V SG. However, it is important to note that the proposed model used a pre-trained word embedding based on one corpus, which is the 1.5 billion words Arabic Corpus (El-khair, 2016), while the AraBERT (Antoun et al., 2020) was pre-trained on both corpora, which are the 1.5 billion words Arabic Corpus (El-khair, 2016) and OSIAN: the Open Source International Arabic News Corpus (Zeroual et al., 2019). Furthermore, the presented model achieved competitive results within significantly less computational time.

## 6. Error analysis

We investigated the efficiency of our proposed classification model by conducting a study including an error analysis of some inputs. Due to the large number of experiments in this paper of our proposed model, we considered applying this analysis only to high-performance models, which are the meaning of W2V SG and fastText SG.

Although the model performs a high level of accuracy in many cases, we involved only incorrectly classified instances for Arabic news articles in this section.  The procedures of analyzing misclassified Arabic news articles are comparing the predicted category with the actual one, then analyzing the occurrences of the misclassification. We observed that many error patterns in the cases were due to vocabulary overlapping, short, ambiguous titles in the Title experiments, or the absence of contextual information. Tables 3-5 reveal examples of incorrectly classified inputs of Arabic newspapers from the three different datasets. The input text in both Tables 4-5 was abridged to show the text feature that has an impact on the classification. to highlight the textual features that influenced the classification decision. Table 3 provides examples of two misclassified Arabic news titles. In the W2V SG 300 model, one title was incorrectly classified as Politics due to the presence of keywords such as "US court" and "parole," which are commonly linked to political news.

However, the actual focus of the title was on the circumstances of a singer's parole, which belongs to the Art domain. In the fastText SG 300 model, another title about launching a digital travel pass was misclassified as Technology. This error was likely caused by overlapping terms such as "app" and "Apple," even though the correct category was Tourism.

Table 4 presents two examples of misclassification based on the body of Arabic news. In the W2V SG 300 model, a news item about registering the Farasan Islands in the UNESCO MAB program, which belongs to the Art category, was incorrectly classified as Local. This error may be due to the presence of geographical terms that leaned the model toward the Local category. In the fastText SG 300 model, a report on the achievement of the Saudi team in the International Mathematical Olympiad, which should be classified as Local, was incorrectly predicted as Sport. This misclassification likely resulted from lexical similarities to sports-related content, such as references to medals, Olympiad, and teams, which misled the model.

**Table 3:** A case study for error analysis using the Title dataset with both models (W2V SG 300 and fastText SG 300) of the mean technique

| The classifier model | The input text | Actual category | Predicted category |
|---|---|---|---|
| W2V SG 300 Mean | *A US court allows singer Chris Brown to travel while he remains on parole… | Art | Politics |
| fastText SG 300 Mean | *IATA's Digital Travel Pass app launches mid-April on Apple... | Tourism | Technology |

*: Original text was in Arabic; translated into English for publication

**Table 4:** A case study for error analysis using the Text dataset with both models (W2V SG 300 and fastText SG 300) of the mean technique

| The classifier model | The input text | Actual category | Predicted category |
|---|---|---|---|
| W2V SG 300 Mean | *The National Commission for Education, Culture and Science and the Saudi Heritage Preservation Society "Turathuna" announced the registration of the Farasan Islands Reserve in the Man and the Biosphere (MAB) Programme,... | Art | Local |
| fastText SG 300 Mean | *Saudi Arabia has achieved a new accomplishment in the International Mathematics Olympiad with the Saudi team winning one silver medal and two bronze... | Local | Sport |

*: Original text was in Arabic; translated into English for publication

Table 5 outlines instances of two misclassified based on both the title and the body of Arabic news. In W2V SG 300, the text is about the closure of the airport during COVID period which correlated to Tourism. Nevertheless, the term related to "low-cost" and suspension of airport operations during the pandemic possibly misled the model toward Economic implications.

In fastText SG 300, the article contains details about Sports news linked to victory in Formula One, but it was predicted A Car. This might be because cars are central to Formula One competitions, as well as the existence of domain-specific terms like "Mercedes" or "driver" might have an impact indication of misclassification.

## 7. Conclusions and future work

This study introduced a novel dataset for Arabic text classification. It is distinguished due to the largest well-balanced Arabic dataset, containing nine categories, namely tourism, economy, car, technology, art, health, sport, local, and politics, with each category comprising 10,000 articles. The quality of the dataset can be attributed to the established criteria for filtering the articles, which include the similarity of article length and the variety of the newspaper sources. The underlying hypothesis of the proposed model is that word embedding effectively represents words in text as vectors. In contrast, some robust machine learning algorithms, such as the SVM algorithm, treat a word's feature as a single numeric value. This algorithm demonstrates strong classification and regression results in various tasks, including NLP and computer visualization. The models developed in this study are based on these two hypotheses. A novel combination technique was proposed that merged word embedding methods with the SVM algorithm by converting the word vector into a single value. This model yields impressive results for Arabic text classification. To precisely evaluate the model's performance, a benchmark model based on the basic TF-IDF technique was used. The proposed method exhibited superior accuracy compared to the basic TF-IDF technique for all the tested datasets, namely the title, body, and combined title and body datasets. Compared to the baseline results, the results for the title dataset ranged between +0.0447 and +0.1088, those for the text dataset ranged between +0.0014 and +0.0268, and those for the combined dataset increased from +0.0061 to +0.0294. The results of our experiments indicated that the W2V SG word embedding model with a vector size of 300 exhibited the highest performance for text feature extraction. These results corroborate those obtained using the W2V SG model with most of the datasets in (Alayba and Palade, 2022).

**Table 5:** A case study for error analysis using a combined dataset with both models (W2V SG 300 and fastText SG 300) of the mean technique

| The classifier model | The input text | Actual category | Predicted category |
|---|---|---|---|
| W2V SG 300 Mean | *Corona closes one of Rome's airports and limits the activities of the other Italy Rome Ciampino Airport, which normally receives low-cost flights, will close tomorrow, Friday,... | Tourism | Economy |
| fastText SG 300 Mean | *Hamilton continues his winning streak and wins the Spanish Grand Prix. British driver Lewis Hamilton, the Mercedes driver, won the Spanish Grand Prix for the fourth time in the Formula One World Championship... | Sport | Car |

*: Original text was in Arabic; translated into English for publication

Future work should consider certain transforming learning approaches such as the BERT model (Devlin et al., 2019), ELMo (Peters et al., 2018), GPT2 (Radford et al., 2019), or XLNet (Yang et al., 2019). In addition, combining certain word embedding models should be explored for feature extraction. ALLaM model (Bari et al., 2024) is remarkable.

## List of abbreviations

| | |
|---|---|
| ALLaM | Arabic large language model |
| ANACD | Arabic news article classification dataset |
| ANAD | Arabic news article dataset |
| AraBERT | Arabic bidirectional encoder representations from transformers |
| BERT | Bidirectional encoder representations from transformers |
| BOW | Bag of words |
| CBOW | Continuous bag of words |
| CCA | Corpus of contemporary Arabic |
| CNN | Convolutional neural network |
| CNN-LSTM | Convolutional neural network–long short-term memory |
| CSV | Comma-separated values |
| ELMO | Embeddings from language models |
| FAIR | Facebook AI research |
| IATA | International air transport association |
| ICA | International corpus of Arabic |
| KNN | K-nearest neighbors |
| MAB | Man and the biosphere |
| MSA | Modern standard Arabic |
| NB | Naïve Bayes |
| NLP | Natural language processing |
| OSAC | Open-source Arabic corpora |
| SANAD | Single-labeled Arabic news articles dataset |
| SATCDM | Superior Arabic text categorization deep model |
| SGD | Stochastic gradient descent |
| SG | Skip-gram |
| SVM | Support vector machine |
| TF | Term frequency |
| TF-IDF | Term frequency–inverse document frequency |
| ULMFit | Universal language model fine-tuning |
| XLNet | Generalized autoregressive pretraining for language understanding |

## Data availability

The generated datasets during the current study are available in the Mendeley Data repository at: https://data.mendeley.com/datasets/w8njshybth/1.

## Acknowledgment

This research has been funded by the Scientific Research Deanship at the University of Ha'il – Saudi Arabia through project number BA-2207.

## Compliance with ethical standards

## Conflict of interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

## References

Abdelali A, Cowie J, and Soliman H (2005). Building a modern standard Arabic corpus. In the Proceedings of the Workshop on Arabic Language Resources and Tools: Their Use in MT and IR, Asia-Pacific Association for Machine Translation, Phuket, Thailand: 25–28.

Abou Khachfeh RR, El Kabani I, and Osman Z (2021). A novel Arabic corpus for text classification using deep learning and word embedding. BAU Journal-Science and Technology, 3(1): 31–39. https://doi.org/10.54729/2959-331X.1014

Ahmed H, Traore I, and Saad S (2018). Detecting opinion spams and fake news using text classification. Security and Privacy, 1: e9. https://doi.org/10.1002/spy2.9

Aizawa A (2003). An information-theoretic perspective of tf–idf measures. Information Processing & Management, 39(1): 45–65. https://doi.org/10.1016/S0306-4573(02)00021-3

Akhadam I and Ayyad H (2024). Enhancing Arabic text classification: A comparative study of machine learning and deep learning approaches. In the IEEE 12th International Symposium on Signal, Image, Video and Communications (ISIVC), IEEE, Marrakech, Morocco: 1–6. https://doi.org/10.1109/ISIVC61350.2024.10577929

Alansary S and Nagi M (2014). The international corpus of Arabic: Compilation, analysis and evaluation. In the Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing, Association for Computational Linguistics, Doha, Qatar: 8–17. https://doi.org/10.3115/v1/W14-3602

Alayba AM (2019). Twitter sentiment analysis on health services in Arabic. M.Sc. Thesis, Coventry University Pureportal, Coventry, UK.

Alayba AM and Palade V (2022). Leveraging Arabic sentiment classification using an enhanced CNN-LSTM approach and effective Arabic text preparation. Journal of King Saud University - Computer and Information Sciences, 34(10): 9710–9722. https://doi.org/10.1016/j.jksuci.2021.12.004

Alayba AM, Palade V, England M, and Iqbal R (2018). Improving sentiment analysis in Arabic using word representation. In the Annual Conference on New Trends in Image Analysis and Processing, Computer Science and Applied Mathematics (ASAR), IEEE, Duhok, Kurdistan Region, Iraq: 13–18. https://doi.org/10.1109/ASAR.2018.8480191

Al-Harbi S, Almuhareb A, Al-Thubaity A, Khorsheed MS, and Al-Rajeh A (2008). Automatic Arabic text classification. In the Proceedings of the 9th International Conference on the Statistical Analysis of Textual Data, Presses Universitaires de Lyon, Lyon, France: 77–85.

Alhawarat M and Aseeri AO (2020). A superior Arabic text categorization deep model (SATCDM). IEEE Access, 8: 24653–24661. https://doi.org/10.1109/ACCESS.2020.2970504

Alsaleem S (2011). Automated Arabic text categorization using SVM and NB. International Arab Journal of e-Technology, 2(2): 124–128.

Al-Sulaiti L and Atwell ES (2006). The design of a corpus of contemporary Arabic. International Journal of Corpus Linguistics, 11(2): 135–171. https://doi.org/10.1075/ijcl.11.2.02als

Altamimi M and Alayba AM (2023). ANAD: Arabic news article dataset. Data in Brief, 50: 109460. https://doi.org/10.1016/j.dib.2023.109460 **PMid:37577410 PMCid:PMC10415830**

Altamimi M and Teahan WJ (2019). Arabic dialect identification of Twitter text using PPM compression. International Journal of Computational Linguistics, 10(4): 47–59.

Antoun W, Baly F, and Hajj H (2020). AraBERT: Transformer-based model for Arabic language understanding. In: Al-Khalifa H, Magdy W, Darwish K, Elsayed T, and Mubarak H (Eds.), Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, European Language Resource Association, Paris, France: 9–15.

Bari MS, Alnumay Y, Alzahrani NA et al. (2024). ALLaM: Large language models for Arabic and English. Arxiv Preprint Arxiv:2407.15390. https://doi.org/10.48550/arXiv.2407.15390

Ben-Hur A and Weston J (2010). A user's guide to support vector machines. In: Carugo O and Eisenhaber F (Eds.), Data mining techniques for the life sciences. Methods in molecular biology: 223–239. Volume 609, Humana Press, Totowa, USA. https://doi.org/10.1007/978-1-60327-241-4_13 **PMid:20221922**

Bickel DR (2003). Robust and efficient estimation of the mode of continuous data: The mode as a viable measure of central tendency. Journal of Statistical Computation and Simulation, 73(12): 899–912. https://doi.org/10.1080/0094965031000097809

Bojanowski P, Grave E, Joulin A, and Mikolov T (2017). Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5: 135–146. https://doi.org/10.1162/tacl_a_00051

Chou C-H, Sinha AP, and Zhao H (2008). A text mining approach to Internet abuse detection. Information Systems and e-Business Management, 6(4): 419–439. https://doi.org/10.1007/s10257-007-0070-0

Cunningham P, Cord M, and Delany SJ (2008). Supervised learning. In: Cord M and Cunningham P (Eds.), Machine learning techniques for multimedia: 21–49. Springer, Berlin, Germany. https://doi.org/10.1007/978-3-540-75171-7_2

Darwish K, Habash N, Abbas M et al. (2021). A panoramic survey of natural language processing in the Arab world. Communications of the ACM, 64(4): 72–81. https://doi.org/10.1145/3447735

Devlin J, Chang M-W, Lee K, and Toutanova K (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In the Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Minneapolis, USA: 4171–4186.

Einea O, Elnagar A, and Al Debsi R (2019). SANAD: Single-label Arabic news articles dataset for automatic text categorization. Data in Brief, 25: 104076. https://doi.org/10.1016/j.dib.2019.104076 **PMid:31440535 PMCid:PMC6700340**

Elarnaoty M and Farghaly A (2018). Machine learning implementations in Arabic text classification. In: Latorre Carmona P (Ed.), Industrial applications of machine learning: 295–324. Springer, Cham, Switzerland. https://doi.org/10.1007/978-3-319-67056-0_15

El-Khair IA (2016). 1.5 billion words Arabic corpus. Arxiv Preprint Arxiv:1611.04033. https://doi.org/10.48550/arXiv.1611.04033

Ghaddar B and Naoum-Sawaya J (2018). High dimensional data classification and feature selection using support vector machines. European Journal of Operational Research, 265(3): 993–1004. https://doi.org/10.1016/j.ejor.2017.08.040

Han H and Jiang X (2014). Overcome support vector machine diagnosis overfitting. Cancer Informatics, 2014: 13s1. https://doi.org/10.4137/CIN.S13875 **PMid:25574125 PMCid:PMC4264614**

Hmeidi I, Hawashin B, and El-Qawasmeh E (2008). Performance of KNN and SVM classifiers on full word Arabic articles. Advanced Engineering Informatics, 22(1): 106–111. https://doi.org/10.1016/j.aei.2007.12.001

Howard J and Ruder S (2018). Universal language model fine-tuning for text classification. Arxiv Preprint Arxiv:1801.06146. https://doi.org/10.48550/arXiv.1801.06146

Joulin A, Grave E, Bojanowski P, and Mikolov T (2017). Bag of tricks for efficient text classification. In the Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Valencia, Spain: 427–431. https://doi.org/10.18653/v1/E17-2068

Khan SUR, Islam MA, Aleem M, and Iqbal MA (2018). Temporal specificity-based text classification for information retrieval. Turkish Journal of Electrical Engineering and Computer

Sciences, 26(6): 2916–2927. https://doi.org/10.3906/elk-1711-136

Khanal J, Tayara H, and Chong KT (2020). Identifying enhancers and their strength by the integration of word embedding and convolution neural network. IEEE Access, 8: 58369–58376. https://doi.org/10.1109/ACCESS.2020.2982666

Labani M, Moradi P, Ahmadizar F, and Jalili M (2018). A novel multivariate filter method for feature selection in text classification problems. Engineering Applications of Artificial Intelligence, 70: 25–37. https://doi.org/10.1016/j.engappai.2017.12.014

Le QV and Mikolov T (2014). Distributed representations of sentences and documents. In the Proceedings of the 31st International Conference on Machine Learning, PMLR, Beijing, China: 1188–1196.

Manning CD, Raghavan P, and Schütze H (2008). Introduction to information retrieval. Cambridge University Press, Cambridge, UK. https://doi.org/10.1017/CBO9780511809071

Mikolov T, Sutskever I, Chen K, Corrado GS, and Dean J (2013). Distributed representations of words and phrases and their compositionality. In: Burges CJC, Bottou L, Welling M, Ghahramani Z, and Weinberger KQ (Eds.), Advances in neural information processing systems 26, Curran Associates, Inc., Lake Tahoe, USA: 3111–3119.

Neogi PPG, Das AK, Goswami S, and Mustafi J (2020). Topic modeling for text classification. In: Mandal JK and Bhattacharya D (Eds.), Emerging technology in modelling and graphics. Advances in intelligent systems and computing: 395–407. Volume 937, Springer, Singapore, Singapore. https://doi.org/10.1007/978-981-13-7403-6_36

Noaman HM, Elmougy S, Ghoneim A, and Hamza T (2010). Naive Bayes classifier based Arabic document categorization. In the International Conference on Intelligent Computing and Information Systems (ICICIS), IEEE, Cairo, Egypt: 757–762.

Pennington J, Socher R, and Manning CD (2014). GloVe: Global vectors for word representation. In the Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Doha, Qatar: 1532–1543. https://doi.org/10.3115/v1/D14-1162

Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, and Zettlemoyer L (2018). Deep contextualized word representations. In the Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, New Orleans, USA: 2227–2237. https://doi.org/10.18653/v1/N18-1202

Radford A, Wu J, Child R, Luan D, Amodei D, and Sutskever I (2019). Language models are unsupervised multitask learners. OpenAI Blog. Available online at: https://openai.com/blog/better-language-models/

Rifai HE, Al Qadi L, and Elnagar A (2021). Arabic multi-label text classification of news articles. In: Hassanien A-E, Chang K-C, and Mincong T (Eds.), Advanced machine learning technologies and applications. AMLTA 2021. Lecture notes in networks and systems: 431–444. Volume 1339, Springer International Publishing, Cham, Switzerland. https://doi.org/10.1007/978-3-030-69717-4_41

Saad MK and Ashour W (2010). OSAC: Open source Arabic corpora. In the 6th International Conference on Electrical and Computer Systems, European University of Lefke, Lefke, North Cyprus: 1-6.

Sebastiani F (2002). Machine learning in automated text categorization. ACM Computing Surveys, 34(1): 1–47. https://doi.org/10.1145/505282.505283

Selva Birunda S and Kanniga Devi R (2021). A review on word embedding techniques for text classification. In: Raj JS, Iliyasu AM, Bestak R, and Baig ZA (Eds.), Innovative data communication technologies and application. Lecture notes on data engineering and communications technologies: 267–281. Volume 59, Springer, Singapore, Singapore. https://doi.org/10.1007/978-981-15-9651-3_23

Setu JH, Halder N, Sikder S, Islam A, and Alam MZ (2024). Empowering multiclass classification and data augmentation of Arabic news articles through transformer model. In the International Joint Conference on Neural Networks, IEEE, Yokohama, Japan: 1–7. https://doi.org/10.1109/IJCNN60899.2024.10650716

Trafalis TB and Gilbert RC (2007). Robust support vector machines for classification and computational issues. Optimization Methods and Software, 22(1): 187–198. https://doi.org/10.1080/10556780600883791

Wang L (2005). Support vector machines: Theory and applications. In: Kacprzyk J (Ed.), Studies in fuzziness and soft computing. Springer, Berlin, Germany. https://doi.org/10.1007/b95439 **PMid:16084744**

Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov RR, and Le QV (2019). XLNet: Generalized autoregressive pretraining for language understanding. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, and Garnett R (Eds.), Advances in neural information processing systems 32 (NeurIPS 2019): 5754–5764. Curran Associates Inc., Vancouver, Canada.

Zeroual I, Goldhahn D, Eckart T, and Lakhouaja A (2019). OSIAN: Open Source international Arabic news corpus - preparation and integration into the CLARIN-infrastructure. In the Proceedings of the 4th Arabic Natural Language Processing Workshop (WANLP), Association for Computational Linguistics, Florence, Italy: 175–182. https://doi.org/10.18653/v1/W19-4619