Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.035910
Article

Novel Machine Learning–Based Approach for Arabic Text Classification Using Stylistic and Semantic Features

Fethi Fkih[1,2,*], Mohammed Alsuhaibani[1], Delel Rhouma[1,2] and Ali Mustafa Qamar[1]

[1]Department of Computer Science, College of Computer, Qassim University, Buraydah, Saudi Arabia
[2]MARS Research Lab LR17ES05, University of Sousse, Sousse, Tunisia
*Corresponding Author: Fethi Fkih. Email: f.fki@qu.edu.sa
Received: 09 September 2022; Accepted: 30 November 2022

Abstract: Text classification is an essential task for many applications related to the Natural Language Processing domain. It can be applied in many fields, such as Information Retrieval, Knowledge Extraction, and Knowledge modeling. Even though the importance of this task, Arabic Text Classification tools still suffer from many problems and remain incapable of responding to the increasing volume of Arabic content that circulates on the web or resides in large databases. This paper introduces a novel machine learning-based approach that exclusively uses hybrid (stylistic and semantic) features. First, we clean the Arabic documents and translate them to English using translation tools. Consequently, the semantic features are automatically extracted from the translated documents using an existing database of English topics. Besides, the model automatically extracts from the textual content a set of stylistic features such as word and character frequencies and punctuation. Therefore, we obtain 3 types of features: semantic, stylistic and hybrid. Using each time, a different type of feature, we performed an in-depth comparison study of nine well-known Machine Learning models to evaluate our approach and used a standard Arabic corpus. The obtained results show that Neural Network outperforms other models and provides good performances using hybrid features (F1-score = 0.88%).

Keywords: Arabic text classification; machine learning; stylistic features; semantic features; topics

1 Introduction

1.1 Context and Problem Statement

The rapid increase in the amount of data has necessitated the development of automated classification systems [1]. The data is mostly available in the form of text and images. Text classification entails classifying texts in one or more pre-defined classes or topics. It is a technique for categorizing open-ended text into a collection of predetermined categories [2]. Text classifiers can organize, arrange, and categorize almost any type of text, including documents, medical research, and files, as well as text found on the internet. For instance, scientific articles in computer science domain can be

categorized by their topics: Artificial intelligence, software engineering, Cybersecurity, etc. Therefore, text classification is a basic problem in Natural Language Processing (NLP) that has a wide range of applications.

NLP is a sub-field of Artificial Intelligence (AI) that helps machines to understand human language [3]. NLP research dates back to 1950 when Alan Turing wrote the paper titled "Computing Machinery and Intelligence". NLP systems are diverse and use technologies such as sentiment analysis, speech recognition, text summarization, and question-answer systems [4]. Many NLP systems employ machine learning (ML) techniques. The most common ML methods include Naïve Bayes (NB) [5], Support Vector Machines (SVM) [6], and k-Nearest Neighbor (kNN) [6]. Quite recently, researchers have applied deep learning (DL) algorithms, such as Convolutional Neural Networks (CNN) [7], auto-encoders, Recurrent Neural Networks (RNN), Gated Recurrent Units (GRU), and Long Short-Term Memory (LSTM) [8] for text classification.

Arabic, a morphologically rich language, is the official language of 22 countries and has more than 422 million native and non-native speakers [9]. As such, Arabic text classification has been at center stage for NLP researchers. For example, Muaad et al. [5] performed Arabic text classification using various ML algorithms, such as multinomial NB, Bernoulli NB, and SVM. They used five publicly available Arabic corpora, including BBC, CNN, Open Source Arabic Corpus, Al-Khaleej, and an Arabic COVID-19 dataset. Muaad et al. [7] compared the performance of ML algorithms with CNN for Arabic document classification. They also used the Al-Khaleej dataset. CNN got the best accuracy of 0.98, slightly better than the state-of-the-art approaches. Similarly, Ibrahim et al. [6] studied Arabic theses and dissertations solely based on their titles. They used multi-classification, where NB was the most accurate algorithm (0.88) compared to SVM, kNN, and Random forest. Similarly, Setyanto et al. [8] used word embedding methods, including GloVe and fastText, along with LSTM for Arabic language opinion mining. The dataset includes 55,000 tweets tagged as positive, negative, or neutral. LSTM achieved an accuracy of 0.909. In another related research, Boussakssou et al. [10] developed an Arabic chatbot using seq2seq models. The results were satisfactory and the chatbot was able to properly communicate with humans.

Even though many research works have tried to handle the task of the Arabic documents' classification, the proposed models remain incapable to supply good performances and are insufficient to cover the majority of knowledge fields. In this context, we propose an efficient model for Arabic document classification.

1.2 Contributions

The main contributions of this work can be summarized in the following points:

- Using automatic translation in order to exploit English semantic resources for Arabic.
- Using semantic topics as features for the classification.
- Conducting a comparative study of a set of nine Machine Learning algorithms.
- Carrying out a comparative study of three types of features: stylistic, semantic, and hybrid.
- Proposing a generic model that can be applied to any knowledge field (sport, medicine, art, etc.).

1.3 Paper Organization

In this paper, we aim to develop an approach for Arabic text classification while employing ML and a combination of stylistic and semantic features. Although a lot of research is available for Arabic

text classification, to the best of our knowledge, we know of no previous research involving topics-based features to improve classification accuracy. Therefore, an English dataset comprising 110 diverse topics is used. The rest of the paper is organized as follows: Section 2 presents a thorough literature review. The proposed model is provided in Section 3, whereas the experiments are provided in Section 4. Section 5 discusses the results and Section 6 concludes the paper along with providing some future directions.

2 Related Work

Automatic classification of texts assembles documents that resemble each other according to specific criteria. There are two types of criteria: observable (the type of document, year, discipline, and edition) and content (topics and writing style). The task of text classification has been experiencing a strong resurgence of interest due to the increasing growth of available digital documents and data and the need to classify them effectively. Moreover, there is a gigantic Arabic text available online. Given the fact that Arabic is the fifth most spoken language in the world with more than 6.0% of the people in the world speaking Arabic [11], the necessity for organizing and classifying these texts has also emerged. Text classification is generally a task that assigns one or more categories from a predefined list to a document. These categories or labels can be selected from a predefined list or automatically extracted by the system.

Arabic text classification is a lively field of research, and automating this task has become a challenge for the Arabic NLP scientific community. Consequently, various research works have evolved considerably over the past few years [12–14], and several new models have appeared. In the following paragraphs, we provide an overview of Arabic text classification.

The initialization of work directed at Arabic text classification might date back to the pioneer works proposed by El-Kourdi et al. [15], Sawaf et al. [16], El Halees [17], and Abdeen et al. [18]. In particular, in the work by El Kourdi et al. [15], a Naïve Bayes classifier has been trained on non-vocalized Arabic web documents (300 in number) for classification to a specific class from a set of five pre-defined categories. Sawaf et al. [16] have looked into Arabic text classification using statistical methods for text analysis for document clustering on the Arabic newswire corpus. However, the morphology of Arabic is a critical issue in any statistical document processing work. When employing classification or clustering on a word level, morphological analyzers are crucial. The sparse data problem is something we have to cope with when employing full-form terms. Morphological analysis can help to solve this problem. Full morphological analysis based on language information and a complicated set of rules is one feasible technique. To deal with morphological processing, Sawaf et al. [16] took a different technique using character n-grams or the use of sub-word units for developing a morphological analyzer. El-Halees [17] has introduced a comprehensive comparative study on the classification of Arabic text. Various well-known classifiers such as Naïve Bayes, SVM, Maximum Entropy, Artificial Neural Networks (ANN), and kNN were compared with unified datasets and experimental settings applying feature selection on Arabic datasets.

The majority of work that followed the aforementioned three pieces of initial research tended to employ a built-in Arabic dataset with various sizes and contents, with the Internet websites as their major source of data [19,20], and then utilize a known ML algorithm as a classifier. For example, Abdeen et al. [21] classified the Arabic text's Internet content using 40 Gigabytes of data divided into five categories, namely culture, economics, religion, sports, and politics. The Naïve Bayes and kNN methods were then used for training the models. Similarly, Alsaleem [22] tested NB and SVM algorithms on various Arabic data indexes. The experimental results revealed that SVM outperformed

the NB. Duwairi et al. [23] proposed three feature reduction approaches for Arabic text, namely, stemming, light stemming, and word clusters. Stemming is the process of reducing words to their stems. By contrast, light stemming eliminates common affixes from words without reducing them to their stems. Word clusters divide synonyms into groups, each represented by a single word. The impact of the aforementioned three strategies were explored and analyzed on the kNN classifier. The goal of using prior methods is to reduce the size of document vectors while maintaining classifier accuracy. Three Feature Subset Selection (FSS) metrics were examined in another investigation done by Mesleh [24]. The author examined the impact of component selection measurements on classification accuracy. Overall, it was demonstrated that the Odd Ratio (OR) performed better than others. Several studies then looked at alternative approaches, including n-gram and separate distance measurements, and their effect on Arabic text classification. Elhassan et al. [25] discussed the paucity of freely available Arabic corpora. They also focus on many kinds of research on the subject of Arabic text classification and depict a logical picture of its approach as well as a camper the evaluation of text classification strategies that were used.

Recently, Arabic text classification using DL techniques has become the main theme for dealing with Arabic text classification challenges. For instance, on different sizes of datasets, Boukil et al. [11] used the Term Frequency-Inverse Document Frequency (TF-IDF) with CNN. Moreover, El-Alami et al. [26] proposed a DL-based technique that uses a deep-stacked autoencoder using word-count vectors as input. In the pre-training step, they used Restricted Boltzmann Machines (RBM), then unrolled the model to create the deep network, before using backpropagation in the fine-tuning stage. They employed various traditional ML algorithms such as NB and SVM to find that deep autoencoders performed well in Arabic text categorization, particularly for SVM. For sentiment analysis of Arabic tweets utilizing feature weighting, Altaher [27] presented a hybrid technique based on deep learning. They chose the most frequently occurring terms in tweets using TF-IDF for feature selection, and then utilized features weighting to choose the most significant characteristics. The results showed that their hybrid technique was feasible and outperformed SVM and other classifiers in terms of accuracy, precision, and efficiency. Galal et al. [12] focused on categorizing Arabic text using CNN as well as taking into account how well it performed in many NLP tasks. They also proposed a novel technique GStem that grouped similar Arabic words based on additional Arabic letters and word embedding distances. Using GStem as a pre-processing stage enhances the accuracy of the CNN model because the number of separate terms is reduced. Moreover, Elnagar et al. [13] proposed two new large corpora, SANAD and NADiA. The authors investigated the impact of utilizing the word embedding models [28,29] to improve the efficiency of categorization tasks. The results indicated that the Convolutional-Gated Recurrent Unit (GRU) had the lowest efficacy and the attention-GRU had the greatest.

Most recently, a supervised feed-forward DL technique was proposed by Sundus et al. [14]. The TF-IDF of datasets is sent into the first layer of the deep network. They then adopted a supervised logistic regression. The experimental investigations revealed a significant improvement in classification efficiency and time spent creating the DL model compared to logistic regression. The findings revealed that using DL classification models to solve the problem of Arabic text categorization is quite promising. Moreover, Alhawarat et al. [30] proposed a CNN multi-kernel architecture along with word embedding to classify Arabic news items. In particular, a novel model named Superior Arabic Text Categorization Deep Model (SATCDM) was introduced. Their approach achieves promising results with various publicly available datasets on Arabic text classification. Furthermore, Gwad et al. [31] presented a Long Short-Term Memory (LSTM), a common type of Recurrent Neural Network (RNN) to evaluate Arabic tweets. They demonstrated that LSTM outperforms classic pattern

recognition techniques in terms of lower parameter calculations, reduced working time, and improved efficiency. Likewise, a Bag-of-Concepts and deep autoencoder representations were presented by El-Alami et al. [32] for Arabic text categorization. It uses Chi-Square methods to identify the most informative characteristics and includes explicit semantics based on Arabic WordNet. To create high-level representations, multiple stacks of Restricted Boltzmann Machines (RBMs) were applied to text vectors. The features are then fed into a deep autoencoder for classification.

In the same context, many researches were oriented towards improving feature selection techniques for Arabic text classification. Any improvement in feature selection techniques will necessarily improve classifier performance. In order to enhance the classification process, the authors in [33] used an improved Chi-square for reducing the huge number of possible features that can be used for classifying Arabic documents. For the same purpose, Chantar et al. [34] proposed an approach based on the binary Grey Wolf Optimizer (GWO) to decrease the higher complexity of the feature selection task. The findings show that combining the proposed features selection approach with an SVM provides good performance. Furthermore, Marie-Sainte et al. introduced in [35] a new firefly algorithm for feature selection dedicated to Arabic documents. The new algorithm was applied with an SVM on an Arabic documents corpus and achieved good results.

To improve the classification quality, many researchers investigated many types of approaches' combinations in order to find the most suitable for the Arabic language. Abuhaiba et al. [36] built four classification models using different combination techniques. The obtained results show that combining many classification algorithms can enhance the Arabic text classification task. In the same context, authors in [37,38] combined CNN, KNN, SVM, and many other approaches using efficient feature representation techniques. In fact, they used light stemming [37] and the word frequency [38] for the features selection task.

Despite the success in Arabic text classification achieved by the aforementioned proposed research on various datasets, to the best of our knowledge, no prior work has ever examined the introduction of semantic indicators for better classification and topic modeling. As such, the proposed work in this paper addresses the problem from this perspective.

3 Proposed Model

In this section, we introduce our proposed approaches for Arabic text classification. To achieve this objective, we propose three models: a stylistic model using statistical features, a semantic model using topics-based features, and a hybrid model that combines stylistic and semantic features. Since there is no Arabic linguistic resource for extracting semantic features, we started with translating Arabic documents into English using online Google Translate. Practically, the original text undergoes a sequence of processes till providing the decision by the system.

3.1 Automatic Translation to English

Currently, translation tools have become performant and provide good results since they rely on semantic resources that make translation tasks more relevant and reflect the true meaning of the original language. Nevertheless, some information will be lost in this stage, especially for such a rich language as Arabic. Table 1 shows an example of two documents extracted from the dataset and translated from Arabic to English using an online Google Translator implemented with Python.

As shown in Table 1, the translation quality is good and mostly maintains the real meaning of the original text. Moreover, it avoids, as much as possible, any information loss that can affect the quality of any further process. After translating the Arabic document to English, a sequence of preprocessing tasks will be applied to the text.

Table 1: Samples of documents translated from Arabic to English using online Google translator

Arabic document	English translation	Class
الأمريكية أوبرا وينفري ألا يقتصر عملها على الفن بل عملت مع أحد المتخصصين لإطلاق نوع جديد من الشاي سيصبح متوفرا ابتداء من الشهر المقبل في سلسلة مقاهي ستاربكس	American star Oprah Winfrey decided not to limit her work to art but worked with a specialist to launch a new type of tea that will become available starting next month in the Starbucks chain of cafes.	Culture
أفادت تقارير إعلامية بريطانية أن نادي مانشستر يونايتد الإنجليزي يضع نصب عينيه التعاقد مع الفرنسي أنطوان غريزمان مهاجم أتلتيكو مدريد الإسباني على الرغم من امتداد تعاقد الأخير مع النادي المدريدي حتى عام	According to British media reports, Manchester United have set their sights on contracting Atletico Madrid striker Antoine Griezmann, despite the extension of the latter's contract with the Madrid club until a year & Sports	Sports

3.2 Text Preprocessing

Since our proposed approach is mainly articulated around a matching process between the document to classify and the topics' dataset, cleaning and preprocessing processes should be applied to the target English document. This process aims to restore words to their standard forms and passes through three main operations:

- Make lower: converts the text to the lower case (not capitalized).
- Text cleaning: removes emojis and emoticons, numbers, and stop words.
- Text lemmatization: groups together the variations and the inflected forms of a given lexical unit so they can be treated as a single token.

Table 2 shows an example of a preprocessed sentence transformed from a raw state to a cleaned and lemmatized text.

Table 2: Example of text preprocessing

Steps	Sentence
Raw text	1 Arsenal striker, olivier giroud, said that the option to leave the club still exists amid interest from AC milan west ham everton and marseille in his services.

(Continued)

Table 2: Continued

Steps	Sentence
Make lower	1 Arsenal striker, olivier giroud, said that the option to leave the club still exists amid interest from ac milan west ham everton and marseille in his services.
Text cleaning	Arsenal striker olivier giroud said option leave club still exists amid interest ac milan west ham everton marseille services.
Lemmatization	Arsenal striker olivier giroud say option leave club still exist amid interest ac milan west ham everton marseille service.

3.3 Features Extraction

After cleaning the text and reducing words to their standard (base) form using a lemmatization process, we extract the features that will be learned from the ML models.

3.3.1 Stylistic Features

Our first proposed model is purely based on statistical (stylistic) features. In fact, using such types of features ensures many advantages for the model. The first advantage is that the model is independent of any language and corpus. Moreover, extracting stylistic features remain a simple task that does not need complex treatment. Table 3 presents the list of stylistic features used in this work.

Table 3: Stylistic features

Feature	Description
Characters-count	Number of characters per document
Words-count	Number of words per document
Short-words-count	Number of short words (having only 3 characters)
Long-words-count	Number of long words (having 6 or more characters)
Hapax-leg-count	Number of words that occur only once (hapax legomena)
Hapax-disleg-count	Number of words occurring twice (hapax dislegomena)
Punctuation-count	Number of punctuation marks (? ! ; , ” ” ’ ’ . . .)

3.3.2 Semantic Features

Basically, our idea is to classify documents according to a set of stylistic and linguistic indicators that can determine their topics. To this purpose, we used an available English dataset of topics, named SEMCAT dataset[1], that contains 6500 words semantically grouped under 110 topics (such as animal, art, baseball, car, and school) [39]. Each topic of this dataset contains a set of words that describe its category. Then, documents were cleaned and lemmatized in order to restore them to their standard forms. After preparing the document text, we generate a 110-dimensional vector containing the weight of each topic in this document. The weight of a given topic corresponds to the sum of occurrences of common words between the document and the file describing the topic in the dataset. Finally, the ML model classifies the document based on its vector.

[1] https://github.com/avaapm/SEMCATdataset2018.

The stepwise implementation of the proposed model is provided in Algorithm 1. As explained in the algorithm, we extract for each document a 110-tuple of numbers denoting the weight of each SEMCATdataset2018's topic. For each word of the document, we compute its frequency in the file representing the topic. Subsequently, the weight of this topic in the processed document will be the sum of all frequencies of all words.

Algorithm 1: Computing topics' scores in a document

Data:
1. $D = \{W_1, \ldots, W_n\}$: a textual document of n words
2. $\text{Top}_{110} = \{f_1, \ldots, f_{110}\}$: dataset of 110 files containing words related to a topic
3. **Result:** V_{110}: 110-tuple (vector) of numbers
4. **begin**
5. $V_{110} \leftarrow 0$
6. **for** $i = 1$ to 110 **do**
7. $W_{freq} \leftarrow 0$
8. **for** $j = 1$ to n **do**
9. **if** W_j in f_i **then**
10. count $\leftarrow 0$
11. **for** $k = 1$ to $|f_i|$ **do**
12. **if** $W_j = tf_k$ **then**
13. count $\leftarrow$ count $+ 1$
14. **end if**
15. **end for**
14. $W_{freq} \leftarrow W_{freq} +$ count
15. **end if**
16. **end for**
17. $V^i_{110} \leftarrow W_{freq}$
18. **end for**
19. **return** V_{110}
20. **end**

4 Experiments and Results

This section describes the experimental study and the obtained results.

4.1 Dataset Description

In order to evaluate our proposed model, we carried out our experimental study on SANAD[2], a standard Arabic dataset containing seven different classes labelled from 0 to 6 [40], as described in Table 4. To ensure the balance among the seven classes, we used 850 documents for each one (5950 documents in total).

[2] https://data.mendeley.com/datasets/57zpx667y9/2 (accessed on 20 May 2022).

Table 4: Distribution of classes in the SANAD dataset

Class label	Class description	Number of documents
0	Culture	850
1	Finance	850
2	Medical	850
3	Politics	850
4	Religion	850
5	Sports	850
6	Tech	850

4.2 Experimental Settings

To evaluate the performances of the proposed model, we carried out an experimental comparison among nine Machine Learning models and used the same semantic features. The list of used ML models is as follows:

- Logistic Regression (LR): It is a statistical model frequently used for forecasting binary outcomes. However, it is not suitable when the correct model should be nonlinear in the parameters [41].
- k-Nearest Neighbors (kNN): It is a simple and efficient model widely used in practice. As a local method, kNN is known to be very efficient for handling large data sets and low dimensions [42].
- Multinomial Naïve Bayes: Multinomial NB classifier is a generative model that is considered an NB classifier variant used for multinomially distributed data, as found in text classification applications [43].
- Decision trees: They are sequential models, which logically combine a sequence of simple tests. Practically, each test compares a numerical attribute against a threshold value or a nominal attribute against a set of possible values [44].
- Linear Discriminant Analysis: It aims to find the projection hyperplane that minimizes the interclass variance and maximizes the distance between the projected means of the classes [45].
- Support Vector Machine (SVM): It is a supervised ML algorithm that can be used for classification as well as regression. It has been applied successfully to many computer-related applications, including text classification [46].
- Gaussian Naïve Bayes (GNB): Although GNBs are efficient, they suffer from the weak assumption of conditional independence between the attributes [47].
- Neural Network (NN): It has recently become popular, especially for the classification task. NN layers are independent of one another, such that a specific layer can have an arbitrary number of nodes [48].
- Random Forest (RF): RF techniques are a combination of tree predictors, such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest [49].

All of the previously mentioned models are implemented using sklearn, a Python library. Furthermore, we use a 10-fold cross-validation procedure: we split the dataset into 10 parts and each time we use 9 parts for training and 1 part for the test. We repeat this process 10 times until all parts were used

for both training and testing. For each experiment, we record the evaluation metrics and the global metric will be the mean of all the recorded values. This procedure is very important to avoid over-fitting, bias, and any systematic errors. In the same time, Arabic and English text analysing had been performed with Python's NLTK library.

4.3 Results

We apply ML models to the dataset, using each time a different type of feature. Table 5 and Fig. 1 summarize the results using stylistic features. The results show that the Random forest outperforms other models with F1-score = 0.73. However, Gaussian Naïve Bayes provides the worst performance with 0.38.

Table 5: Performance comparison of the different models in terms of precision, recall, and F1-score using stylistic features. The best values are highlighted in bold

ML models	Precision (avg.)	Recall (avg.)	F1-score (avg.)
Logistic regression	0.73	0.72	0.72
Decision tree	0.55	0.53	0.53
kNN	0.66	0.63	0.64
Linear discriminant analysis	0.73	0.7	0.71
Multinomial Naïve Bayes	0.69	0.67	0.67
Gaussian Naïve Bayes	0.44	0.35	0.38
SVM	0.71	0.72	0.71
Random forest	**0.75**	**0.73**	**0.73**
Neural network	0.74	0.71	0.72

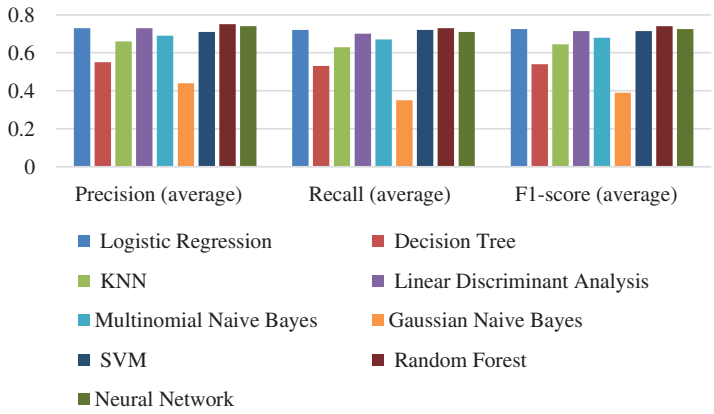

Figure 1: Performance comparison of the different ML models using stylistic features in terms of precision, recall, and F1-score

Table 6 and Fig. 2 show that Logistic regression and Neural Network outscored other models in terms of Precision, Recall, and F1-score when using semantic features. These models provided similar values for the three evaluation metrics. Besides, SVM provided a good result (F1-score = 0.85) compared with other models, such as Linear Discriminant Analysis and Random Forest. On the other

hand, the results show that the performance of Gaussian Naïve Bayes is very far from other models and provides the worst result (F1-score = 0.36).

Table 6: Performance comparison of the different models in terms of precision, recall, and F1-score using semantic features

ML models	Precision (avg.)	Recall (avg.)	F1-score (avg.)
Logistic regression	**0.86**	**0.86**	**0.86**
Decision tree	0.62	0.62	0.62
kNN	0.79	0.78	0.78
Linear discriminant analysis	0.85	0.81	0.82
Multinomial Naïve Bayes	0.79	0.78	0.78
Gaussian Naïve Bayes	0.52	0.36	0.36
SVM	0.85	0.85	0.85
Random forest	0.81	0.81	0.81
Neural network	**0.86**	**0.86**	**0.86**

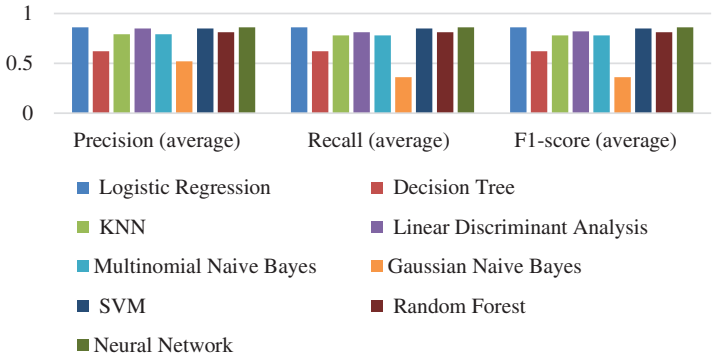

Figure 2: Performance comparison of the different ML models using semantic features in terms of precision, recall, and F1-score

The final test in our experimental study uses a hybrid type of features, a combination of stylistic and semantic features. Table 7 and Fig. 3 show that Neural Network performs the best, with F1-score = 0.88. Besides, Logistic Regression, Linear Discriminant Analysis, and SVM provide good results.

Table 7: Performance comparison of the different models in terms of precision, recall, and F1-score using hybrid features

ML models	Precision (avg.)	Recall (avg.)	F1-score (avg.)
Logistic regression	0.86	0.87	0.86
Decision tree	0.72	0.73	0.72
kNN	0.82	0.83	0.82

(Continued)

Table 7: Continued

ML models	Precision (avg.)	Recall (avg.)	F1-score (avg.)
Linear discriminant analysis	0.87	0.85	0.85
Multinomial Naïve Bayes	0.81	0.82	0.81
Gaussian Naïve Bayes	0.61	0.55	0.57
SVM	0.87	0.88	0.87
Random forest	0.83	0.82	0.82
Neural network	**0.88**	**0.89**	**0.88**

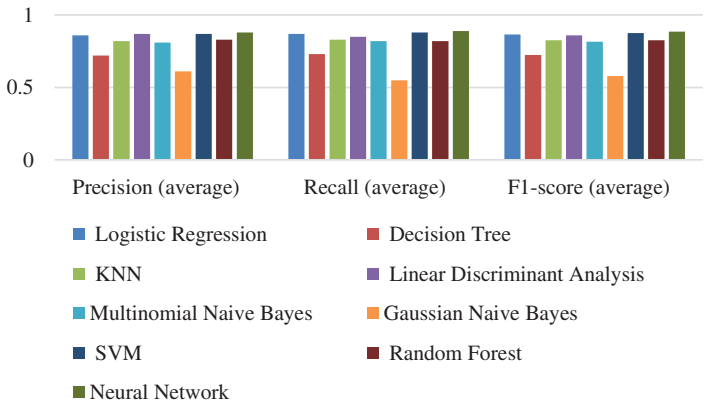

Figure 3: Performance comparison of the different ML models using hybrid features in terms of precision, recall, and F1-score

If we compare the results supplied by three types of features, we can easily observe that hybrid features outperform other types of features as shown in Table 8. Besides, stylistic features provide the worst performance. Also, the neural network model provides good results for the three types of features and especially for the hybrid model as it outperforms all the other models.

Table 8: Performance comparison of the different features in terms of F1-score

Features' type	ML model	F1-score (avg.)
Stylistic	Random forest	0.73
Semantic	Neural network	0.86
Hybrid	Neural network	**0.88**

5 Discussion

The results presented in the previous section can lead to many important conclusions. First, we can observe that hybrid features are the best choice to guarantee the best performance for the ML for Arabic text classification applications. Secondly, the semantic features are always better than stylistic ones, even though it is much simpler to extract stylistic features as compared to semantic

features. Furthermore, the results show that the neural network can be a good choice for Arabic text classification, especially when using hybrid features.

Also, the results reveal that a simple topic-based model can perform well for an Arabic text classification task. In fact, we do not need to carry out an in-depth linguistic specification to determine which relevant semantic features should be extracted to ensure good performance for our model. We have shown in our work that an existing topic can be a relevant feature for the classification task.

Besides, we have demonstrated that the automatic translation to the English language can be a good solution for the lack of Arabic linguistic resources. Actually, the automatic translation software has been significantly improved in the past few years and can offer an accurate translation from Arabic to English. Once the Arabic text is translated into the English language, it can exploit the wide specter of semantic and linguistic external resources available for the English language.

Nevertheless, the model's performance can be further improved by integrating more semantic and linguistic features that can be automatically extracted from the text [50,51].

6 Conclusion and Future Work

In this paper, we proposed a novel approach for Arabic text classification. This idea consists of using an existing topic's database in order to recognize the list of topics in the textbook. Since such a database does not exist for the Arabic language, we translated the dataset to English in order to use an available English database. Moreover, we used three types of features: stylistic, semantic, and hybrid in order to investigate the most adequate type that ensures good performance.

Furthermore, we conducted an experimental study on a well-known dataset in order to compare ML models while employing different types of features. Actually, we integrated the same features (stylistic, topics, or hybrid) into a set of well-known machine learning algorithms. The results show that neural network outscores other models using hybrid features.

In future, we intend to extend our model to cover other languages and integrate more semantic features as the topic's database does not contain all possible topics. Also, the used topics are generic, which can affect the performance of the learning model. An automatic model that can split a generic topic into a set of specific ones can definitively improve the classification model.

Acknowledgement: The researchers would like to thank the Deanship of Scientific Research, Qassim University for funding the publication of this project.

Funding Statement: The authors received no specific funding for this study.

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

References

[1] A. Fernández, S. del Río, A. Bawakid and F. Herrera, "Fuzzy rule based classification systems for big data with MapReduce: Granularity analysis," *Advances in Data Analysis and Classification*, vol. 11, pp. 711–730, 2017.

[2] C. -G. Artene, M. N. Tibeică, D. D. Vecliuc and F. Leon, "Convolutional neural networks for web documents classification," in *Proc. of Asian Conf. on Intelligent Information and Database Systems (ACIIDS)*, Phuket, Thailand, pp. 289–302, 2021.

[3] F. Fkih and M. N. Omri, "Hybridization of an index based on concept lattice with a terminology extraction model for semantic information retrieval guided by WordNet," in *Proc. of the 16th Int. Conf. on Hybrid Intelligent Systems (HIS 2016)*, Marrakech, Morocco, pp. 144–152, 2017.
[4] S. Jusoh, "A study on NLP applications and ambiguity problems," *Journal of Theoretical and Applied Information Technology*, vol. 96, no. 6, pp. 1486–1499, 2018.
[5] A. Y. Muaad, H. J. Davanagere, D. S. Guru, J. V. B. Benifa, C. Chola *et al.,* "Arabic document classification: Performance investigation of preprocessing and representation techniques," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–16, 2022.
[6] M. F. Ibrahim and A. Al-Taei, "Title-based document classification for arabic theses and dissertations," in *Proc. of Int. Conf. on Data and Information Sciences (ICDIS)*, Agra, India, pp. 189–203, 2021.
[7] A. Y. Muaad, G. H. Kumar, J. Hanumanthappa, J. V. B. Benifa, M. N. Mourya *et al.,* "An effective approach for arabic document classification using machine learning," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 267–271, 2022.
[8] A. Setyanto, A. Laksito, F. Alarfaj, M. Alreshoodi, Kusrini *et al.,* "Arabic language opinion mining based on long short-term memory (LSTM)," *Applied Sciences*, vol. 12, no. 9, pp. 1–18, 2022.
[9] N. Boudad, R. Faizi, R. O. H. Thami and R. Chiheb, "Sentiment analysis in Arabic: A review of the literature," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 2479–2490, 2018.
[10] M. Boussakssou, H. Ezzikouri and M. Erritali, "Chatbot in Arabic language using seq to seq model," *Multimedia Tools and Applications*, vol. 81, no. 2, pp. 2859–2871, 2022.
[11] S. Boukil, M. Biniz, F. El Adnani, L. Cherrat and A. E. El Moutaouakkil, "Arabic text classification using deep learning technics," *International Journal of Grid and Distributed Computing*, vol. 11, no. 9, pp. 103–114, 2018.
[12] M. Galal, M. M. Madbouly and A. El-Zoghby, "Classifying Arabic text using deep learning," *Journal of Theoretical and Applied Information Technology*, vol. 97, no. 23, pp. 3412–3422, 2019.
[13] A. Elnagar, R. Al-Debsi and O. Einea, "Arabic text classification using deep learning models," *Information Processing & Management*, vol. 57, no. 1, pp. 102–121, 2020.
[14] K. Sundus, F. Al-Haj and B. Hammo, "A deep learning approach for arabic text classification," in *Proc. of the 2019 2nd Int. Conf. on New Trends in Computing Sciences (ICTCS)*, Amman, Jordan, pp. 1–7, 2019.
[15] M. El Kourdi, A. Bensaid and T. -E. Rachidi, "Automatic Arabic document categorization based on the Naïve Bayes algorithm," in *Proc. of the Workshop on Computational Approaches to Arabic Script-Based Languages*, Geneva, Switzerland, pp. 51–58, 2004.
[16] H. Sawaf, J. Zaplo and H. Ney, "Statistical classification methods for Arabic news articles," in *Proc. of the Natural Language Processing in ACL2001*, Toulouse, France, 2001.
[17] A. M. El-Halees, "A comparative study on Arabic text classification," *Egyptian Computer Science Journal*, vol. 30, no. 2, pp. 1–11, 2008.
[18] M. A. R. Abdeen, S. AlBouq, A. Elmahalawy and S. Shehata, "A closer look at Arabic text classification," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 11, pp. 677–688, 2019.
[19] J. Ababneh, O. Almomani, W. Hadi, N. K. T. El-Omari and A. Al-Ibrahim, "Vector space models to classify Arabic text," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 7, no. 4, pp. 219–223, 2014.
[20] R. Mamoun and M. Ahmed, "Arabic text stemming: Comparative analysis," in *Proc. of 2016 Conf. of Basic Sciences and Engineering Studies (SGCAC)*, Khartoum, Sudan, pp. 88–93, 2016.
[21] M. Abdeen, A. Elsehemy, T. Nazmy and M. C. E. Yagoub, "Classifying the Arabic web—a pilot study," in *Proc. of 2011 24th Canadian Conf. on Electrical and Computer Engineering (CCECE)*, Niagara Falls, ON, Canada, pp. 000865–000868, 2011.
[22] S. Alsaleem, "Automated Arabic text categorization using SVM and NB," *International Arab Journal of e-Technology*, vol. 2, no. 2, pp. 124–128, 2011.
[23] R. Duwairi, M. N. Al-Refai and N. Khasawneh, "Feature reduction techniques for Arabic text categorization," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 11, pp. 2347–2352, 2009.

[24] A. M. Mesleh, "Feature sub-set selection metrics for Arabic text classification," *Pattern Recognition Letters*, vol. 32, no. 14, pp. 1922–1929, 2011.
[25] R. Elhassan and M. Ahmed, "Arabic text classification review," *International Journal of Computer Science and Software Engineering*, vol. 4, no. 1, pp. 1–5, 2015.
[26] F. -Z. El-Alami and S. O. El Alaoui, "An efficient method based on deep learning approach for Arabic text categorization," in *Proc. of Int. Arab Conf. on Information Technology (ACIT'2016)*, Beni-Mellal, Morocco, pp. 1–7, 2016.
[27] A. Altaher, "Hybrid approach for sentiment analysis of Arabic tweets based on deep learning model and features weighting," *International Journal of Advanced and Applied Sciences*, vol. 4, no. 8, pp. 43–49, 2017.
[28] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint, arXiv: 1301.3781, 2013.
[29] M. Alsuhaibani, D. Bollegala, T. Maehara and K. -I. Kawarabayashi, "Jointly learning word embeddings using a corpus and a knowledge base," *PloS One*, vol. 13, no. 3, pp. e0193094, 2018.
[30] M. Alhawarat and A. O. Aseeri, "A superior arabic text categorization deep model (SATCDM)," *IEEE Access*, vol. 8, pp. 24653–24661, 2020.
[31] W. H. G. Gwad, I. M. I. Ismael and Y. Gultepe, "Twitter sentiment analysis classification in the Arabic language using long short-term memory neural networks," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 9, no. 3, pp. 2249–8958, 2020.
[32] F. -Z. El-Alami, A. El Mahdaouy, S. O. El Alaoui and N. En-Nahnahi, "A deep autoencoder-based representation for Arabic text categorization," *Journal of Information and Communication Technology*, vol. 19, no. 3, pp. 381–398, 2020.
[33] S. Bahassine, A. Madani, M. Al-Sarem and M. Kissi, "Feature selection using an improved Chi-square for Arabic text classification," *Journal of King Saud University–Computer and Information Sciences*, vol. 32, no. 2, pp. 225–231, 2020.
[34] H. Chantar, M. Mafarja, H. Alsawalqah, A. A. Heidari, I. Aljarah *et al.,* "Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification," *Neural Computing & Applications*, vol. 32, pp. 12201–12220, 2020.
[35] S. Marie-Sainte and N. Alalyani, "Firefly algorithm based feature selection for Arabic text classification," *Journal of King Saud University–Computer and Information Sciences*, vol. 32, no. 3, pp. 225–231, 2020.
[36] I. S. I. Abuhaiba and H. M. Dawoud, "Combining different approaches to improve Arabic text documents classification," *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 9, no. 4, pp. 39–52, 2017.
[37] Y. A. Alhaj, M. A. A. Al-qaness, A. Dahou, M. Abd Elaziz, D. Zhao *et al.,* "Effects of light stemming on feature extraction and selection for Arabic documents classification," in *Recent Advances in NLP: The Case of Arabic Language. Studies in Computational Intelligence*, Cham, Switzerland: Springer, pp. 59–79, 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-34614-0_4
[38] Y. A. Alhaj, W. U. Wickramaarachchi, A. Hussain, M. A. A. Al-Qaness and H. M. Abdelaal, "Efficient feature representation based on the effect of words frequency for arabic documents classification," in *Proc. of the 2nd Int. Conf. on Telecommunications and Communication Engineering (ICTCE 2018)*, Beijing, China, pp. 397–401, 2018.
[39] L. K. Senel, I. Utlu, V. Yucesoy, A. Koc and T. Cukur, "Semantic structure and interpretability of word embeddings," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 26, no. 10, pp. 320–328, 2018.
[40] O. Einea, A. Elnagar and R. A. Debsi, "Sanad: Single-label Arabic news articles dataset for automatic text categorization," *Data in Brief*, vol. 25, pp. 1–5, 2019.
[41] A. DeMaris and S. H. Selman, "*Logistic Regression*," New York, NY: Springer, pp. 115–136, 2013. [Online]. Available: https://doi.org/10.1007/978-1-4614-7792-1_7
[42] O. Kramer, "*K-Nearest Neighbors*," Berlin, Heidelberg: Springer , pp. 13–23, 2013. [Online]. Available: https://doi.org/10.1007/978-3-642-38652-7_2
[43] S. Xu, Y. Li and Z. Wang, "Bayesian multinomial Naïve Bayes classifier to text classification," in *Proc. of 12th Int. Conf. on Future Information Technology (FutureTech 2017)*, Seoul, Korea, pp. 347–352, 2017.

[44] S. B. Kotsiantis, "Decision trees: A recent overview," *Artificial Intelligence Review*, vol. 39, pp. 261–283, 2013.
[45] P. Xanthopoulos, P. M. Pardalos and T. B. Trafalis, "Linear discriminant analysis," in *Robust Data Mining. SpringerBriefs in Optimization*, New York, NY, USA: Springer, pp. 27–33, 2013. [Online]. Available: https://doi.org/10.1007/978-1-4419-9878-1_4
[46] R. Gholami and N. Fakhari, "Chapter 27—Support Vector Machine: Principles, parameters, and applications," in *Handbook of Neural Computation*, Cambridge, MA, USA: Academic Press, pp. 515–535, 2017.
[47] A. H. Jahromi and M. Taheri, "A Non-parametric mixture of Gaussian Naïve Bayes classifiers based on local independent features," in *Proc. of 2017 Artificial Intelligence and Signal Processing Conf. (AISP)*, Shiraz, Iran, pp. 209–212, 2017.
[48] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed *et al.,* "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, pp. 1–41, 2018.
[49] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[50] F. Fkih and M. N. Omri, "Information retrieval from unstructured web text document based on automatic learning of the threshold," *International Journal of Information Retrieval Research (IJIRR)*, vol. 2, no. 4, pp. 12–30, 2012.
[51] S. Ouni, F. Fkih and M. N. Omri, "Toward a new approach to author profiling based on the extraction of statistical features," *Social Network Analysis and Mining*, vol. 11, no. 1, pp. 1–16, 2021.