2024 آب العدد 14
No.14 Aug 2024
المجلة العراقية للبحوث الإنسانية والإجتماعية والعلمية
Iraqi Journal of Humanitarian, Social and Scientific Research
Print ISSN 2710-0952 Electronic ISSN 2790-1254

Arabic Text Classification Using Machine Learning

Ahmed Najm Abood-Majed AL-Freh
Department of Computer Science, Qom University, Iran
Corresponding author:ahmedstar1225@gmail.com

Abstract

The exponential increase in the availability of electronic text across companies, universities, and the Internet necessitates the development of automated classification tools for efficient content exploration and analysis. These tools leverage artificial intelligence, particularly machine learning (ML) and natural language processing (NLP), to enable systems to learn from data, identify patterns, and make decisions with minimal human intervention. Text classification (TC) is a crucial NLP technique used to categorize text into predefined classes, facilitating information management and retrieval. Despite substantial research in text classification for languages like English, French, and Chinese, there has been limited focus on Arabic text classification due to the language's inherent complexity. Arabic, spoken by over 420 million people worldwide, poses unique challenges in automatic processing due to its rich morphology, syntax, and semantics. This paper presents an innovative method for Arabic text classification using a combination of an Arabic stemming algorithm, Term Frequency-Inverse Document Frequency (TF-IDF) for feature weighting, and Convolutional Neural Networks (CNNs) for classification. The proposed approach demonstrates significant improvement in classification accuracy, highlighting the potential of machine learning techniques in handling the complexities of Arabic text.

Keywords ***: Arabic Text, Machine Learning, Classification, TF-IDF, Techniques.***

تصنيف النص العربي باستخدام التعلم الآلي

احمد نجم عبود ماجد الفريح
قسم علوم حاسبات ، جامعة قم ،ايران

خلاصة

إن الزيادة الهائلة في توفر النص الإلكتروني عبر الشركات والجامعات والإنترنت تستلزم تطوير أدوات تصنيف آلية لاستكشاف المحتوى وتحليله بكفاءة. وتستفيد هذه الأدوات من الذكاء الاصطناعي، وخاصة التعلم الآلي **(ML)**ومعالجة اللغة الطبيعية**(NLP)** ، لتمكين الأنظمة من التعلم من البيانات، وتحديد الأنماط، واتخاذ القرارات بأقل قدر من التدخل البشري. يعد تصنيف النص **(TC)** أحد تقنيات البرمجة اللغوية العصبية المهمة المستخدمة لتصنيف النص إلى فئات محددة مسبقًا، مما يسهل إدارة المعلومات واسترجاعها. على الرغم من الأبحاث الكبيرة في تصنيف النصوص للغات مثل الإنجليزية والفرنسية والصينية، كان هناك تركيز محدود على تصنيف النص العربي بسبب التعقيد المتأصل في اللغة. تشكل اللغة العربية، التي يتحدث بها أكثر من 420 مليون شخص في جميع أنحاء العالم، تحديات فريدة في المعالجة التلقائية بسبب مورفولوجيتها الغنية وتركيبها ودلالاتها. تقدم هذه الورقة طريقة مبتكرة لتصنيف النص العربي باستخدام مزيج من الخوارزمية

الجذعية العربية، وتكرار المصطلح - تردد المستند العكسي **(TF-IDF)** لوزن الميزات، والشبكات العصبية التلافيفية **(CNNs)** للتصنيف. يوضح النهج المقترح تحسنًا كبيرًا في دقة التصنيف، ويسلط الضوء على إمكانات تقنيات التعلم الآلي في التعامل مع تعقيدات النص العربي.
الكلمات الدالة :النص العربي، التعلم الآلي، التصنيف، ***TF-IDF***، *التقنيات*.

1.Introduction

The availability of electronic text in companies and universities, as well as on the Internet, has increased exponentially in recent years. This makes it necessary to develop automatic classification tools that allow fast content exploration and analysis. The most important of these tools is that they are based on the concepts of artificial intelligence methods, machine learning and natural language processing. Machine learning (ML) is a data analysis method that automates the creation of analytical models. It is a branch of artificial intelligence that allows systems to learn from data, identify patterns and make decisions with minimal human intervention.

Natural Language Processing (NLP) is a subfield of artificial intelligence that enables computers to understand and analyze human language. It is applied now in several areas such as document indexing, automatic summarization of documents, plagiarism detection and document classification.

Text classification (TC) is the process of grouping texts into categories regarding their content. Text classification is an important learning issue that is at the core of much information management and retrieval tasks. Until now a lot of works focused on text classification such as the classification of English, French and Chinese texts etc., but, for Arabic texts, there are few studies relating to the classification of Arabic texts. This is explained by the complexity of the Arabic texts. Arabic is one of the commonly spoken languages with more than 420 million speakers around the world. The Arabic alphabet is a collection of 28 letters:

أ ب ت ث ج ح خ د ذ ر ز س ش ص ض ظ ط ع غ ف ق ك ل م ن ه و ي

In the Arabic linguistics, there is also hamza (ء) considered as a letter, the vowels are (ا و ي), it also differs by diacritics that represent a small vowel like (fatha, kasra, damma, sukun, shadda, and tanween). The orthography system of the Arabic language is based on the diacritical effect [1].

2.Classification and Representation of Text

Text classification is a natural language processing method that allows us to classify texts into predefined classes. It is a very popular technique that has a variety of use cases. Text classification consists in assigning a Text to one or more classes to index the Text in a predefined set of categories, originally designed to assist in the documentary classification of books or articles in technical or scientific domains. The objective of this process is to be able to automatically perform the categories of

a set of new texts. Text classification consists in learning, from examples characterizing thematic classes, a set of discriminating descriptors to classify a given Text into the class (or classes) corresponding to its content [2].

The factorization of texts is one of the most important tasks in document classification. In fact, it is necessary to use an efficient representation technique that allows texts to be represented in a machine-readable form. The most popular representation is the vector model in which each text is represented by a vector of n weighted terms. There are many approaches to do this. Bag of Words. The idea is to transform texts into vectors, each component of which represents a word. Words have the advantage of having an explicit meaning. However, there are several problems. Firstly, it is necessary to define what "a word" is in order to be able to process it automatically. It can be considered as a sequence of characters belonging to a dictionary, or, more practically, as a sequence of non-delimiting characters framed by delimiting characters (punctuation characters); it is then necessary to manage the acronyms, as well as the compound words; this requires linguistic preprocessing [3].

Text Classification is the process of categorizing text into one or more different classes to organize, structure, and filter into any parameter. For example, text classification is used in legal documents, medical studies and files, or as simple as product reviews. Data is more important than ever; companies are spending fortunes trying to extract as many insights as possible. With text/document data being much more abundant than other data types, new methods of utilizing them are imperative. Since data is inherently unstructured and extremely plentiful, organizing data to understand it in digestible ways can drastically improve its value. Using Text Classification with Machine Learning can automatically structure relevant text in a faster and more cost-effective way.

3.Arabic Language and Classification

Arabic Language is one of the most spread languages. It is the 5th most used language in the word and the 5th most used language in the internet. Moreover, the Arabic language is used by more than 422 million people in 2017: 290 million as first language and 132 million as second language. The Arabic language has a special way for writing. In the contrary of English, French and all the western languages in general, the Arabic language is writing from the right to the left, and there is no lowercase and uppercase in Arabic letters.

The shapes of some letters can change according to their position in the word, for example the letter "ع "can be writing in four different ways depending on his place in the word: in the beginning of the word "box, *علبة:* عـ ", in the middle of the word "game, *لعبة:* ـعـ "and in the end of the word "Square, *مربع:* ـع | radio, *مذياع:* ع ". It is

one of the most challenging languages in the world with its rich morphology, its complex syntax, and its difficult semantics. This makes its analysis and automatic processing very hard and complex

3.1Arabic Text Classification Process

The text classification system aims at predicting the class (category) of a newly introduced text. To be performed, the text classification relays on three main phases, namely Training, Test and Prediction.

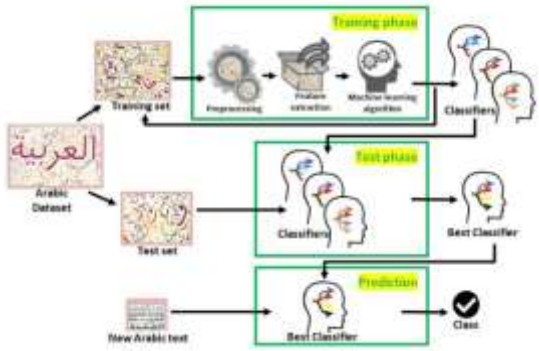

Figure 1 Flowchart of the Arabic text classification process

Feature extraction is also referred to as feature weighting, Indexing, or document representation. We also note that the classification process can be enhanced by more additional step, namely feature selection, which aims at selecting a subset of the features available for describing the texts to reduce the dimensions of the feature matrix; if this step is ignored, all features will be considered to build the feature matrix.

4.Related Work

Text classification is the process of gathering documents into classes and categories based on their contents. This process is becoming more important due to the huge textual information available online. The main problem in text classification is how to improve the classification accuracy. Many algorithms have been proposed and implemented to solve this problem in general.

However, few studies have been carried out for categorizing and classifying Arabic text. Technically, the process of text classification follows two steps; the first step consists on selecting some special features from all the features available from the text by applying features selection, features reduction and features weighting techniques. And the second step applies classification algorithms on those chosen features. In this paper, present an innovative method for Arabic text classification,

use an Arabic stemming algorithm to extract, select and reduce the features that need. After that, use the Term Frequency-Inverse Document Frequency technique as feature weighting technique. And finally, for the classification step, use one of the deep learning algorithms that is very powerful in other field such as the image processing and pattern recognition, but still rarely used in text mining, this algorithm is the Convolutional Neural Networks. With this combination and some hyperparameter tuning in the Convolutional Neural Networks algorithm we can achieve excellent results on multiple benchmarks [4].

Text categorization refers to the process of grouping text or documents into classes or categories according to their content. Text categorization process consists of three phases which are: preprocessing, feature extraction and classification. In comparison to the English language, just few studies have been done to categorize and classify the Arabic language. For a variety of applications, such as text classification and clustering, Arabic text representation is a difficult task because Arabic language is noted for its richness, diversity, and complicated morphology. This paper presents a comprehensive analysis and a comparison for researchers in the last five years based on the dataset, year, algorithms and the accuracy they got. Deep Learning (DL) and Machine Learning (ML) models were used to enhance text classification for Arabic language. Remarks for future work were concluded [5].
The process of classifying texts into categories by subject, author or title is called Arabic text classification. The core of this systemic analysis was reporting regarding various algorithms and datasets. The databases used in the presented work are constructed using websites of Arabic news, while other studies used datasets created by other researchers such as open-source Arabic corpus.
It criticized the classification for corpus and the approaches create the model, either they included deep learning or machine learning technique. The types of deep learning used were also listed such as RNN, MLP, CNN, GRU, LSTM, FFNN and others. In addition, the attention was upon publication year and the datasets of which the articles were written. Furthermore, it reviewed the performance metrics used to compare the built models. In addition, in this systematic analysis we have found for several reasons we cannot generalize one kind of deep learning as the efficient one in Arabic text classification because in each study the neural networks used were distinct. There was other missed information in situations where the researchers used the same kind of NN. But after deep analysis, we noticed that LSTM is more appropriate than other because for text classification tasks, such networks are an attractive solution since word order in text can be essential. The researchers did not demonstrate what parameters they used in these networks in depth and how the

parameters are tuned. Typically, by adjusting parameters and rerunning the tests to produce significant effects, the machine learning algorithms are tuned. This made it impossible to compare or make sharp choices on which neural networks were the strongest. The majority of the work, if the data size is huge, showed a better output measurement of the deep learning technique over machine learning. But traditional Machine Learning algorithms are superior to limited data sizes. So, we used deep learning because the dataset SANAD size is large enough and specifically we used akhbarona of this dataset which it contains 7 categories [Medical, Sports, Finance, Religion, Culture, Politics, and Tech]. Our direction will be toward the deep learning algorithms because in machine learning the testing accuracy will reach a certain limit and cannot increase while the deep learning algorithms increases more in testing accuracy whenever the dataset is large.

Nowadays, the volume of data offered on the Internet is growing every moment, and the necessity to analyze these data and convert to useful information increased. There are several types of research exploring techniques to deal with Text Classification (TC) in many languages; however, In Arabic, the researchers are limited. TC is the process of categorizing text document into classes or categories according to the text contents. This research will focus on classifying Arabic Text using a Convolution neural network (CNN), which considered one of deep learning (DL) methods, as it achieved an excellent result in different Natural language processing (NLP) project types, also introduced a novel algorithm to group similar Arabic words based on extra Arabic letters and word embeddings distances, named this algorithm as GStem [6].

Make our experiments in two types of models. CNN-Norm which all word vectors learned from scratch using SG word2vec then learn document classes using the CNN without using GStem. CNNGStem which all word vectors learned from scratch using SG word2vec then run GStem algorithm to group similar words based on their extra letters finally learn document classes using the CNN. Log the result of CNN-GStem model with some variations of GStem parameters as 0.3 and 0.4 for AWED and 200, 500 and 1000 for TCCS. Select model hyper-parameter by a grid search on the tested dataset. For word2vec learn it using SG architecture, select 5 for window size,50 for vector dimensionality and train it with 200 iterations. For CNN filter windows, we made it 2,3, 5 with 10 feature maps for each filter, 0.5 for dropout rate,50 for mini-batch and 20 learning epochs. We split the dataset as 80% for learning and 20% for testing and evaluation. The implementation of CNN is done by Kera's with TensorFlow backend.

Introduced a new technique (GStem) to group similar words that share the same root based on the Arabic extra letters as a preprocessing layer for the CNN. In order to test our model, collected an Arabic news dataset and used it in our experiments.

Trained our WE from scratch depending on the collected dataset training samples and didn't depend on any pre-trained data for word2vec or for GStem. Our experiments showed that when using GStem as a preprocessing step it increases the accuracy of the CNN model and this because the number of distinct words has been reduced. Our future plan is to adjust GStem to add special weights for most frequent letters and adjust it to deal with other extra letters such as prepositions attached to the word. Also planned to modify it to deal with another language.

Text classification or categorization is the process of automatically tagging a textual document with most relevant labels or categories. When the number of labels is restricted to one, the task becomes single-label text categorization. However, the multi-label version is challenging. For Arabic language, both tasks (especially the latter one) become more challenging in the absence of large and free Arabic rich and rational datasets. Therefore, we introduce new rich and unbiased datasets for both the single-label (SANAD) as well as the multi-label (NADiA) Arabic text categorization tasks. Both corpora are made freely available to the research community on Arabic computational linguistics. Further, we present an extensive comparison of several deep learning (DL) models for Arabic text categorization in order to evaluate the effectiveness of such models on SANAD and NADiA. A unique characteristic of our proposed work, when compared to existing ones, is that it does not require a pre-processing phase and fully based on deep learning models. Besides, we studied the impact of utilizing word2vec embedding models to improve the performance of the classification tasks. Our experimental results showed solid performance of all models on SANAD corpus with a minimum accuracy of 91.18%, achieved by convolutional-GRU, and top performance of 96.94%, achieved by attention-GRU. As for NADiA, attention-GRU achieved the highest overall accuracy of 88.68% for a maximum subset of 10 categories on "Masrawy" dataset [7]. In this work, we presented two new large corpora for Arabic text categorization tasks as a contribution to the research community on Arabic computational linguistics. Namely, SANAD and NADiA. While SANAD is designed for single-label Arabic text, NADiA is intended for multi-label categorization tasks. Both datasets are collected from annotated Arabic news articles. SANAD consists of 3 datasets; two (Arabiya and Akhbarona) are imbalanced while Khaleej dataset is a balanced one.

Arabic text representation is a challenging assignment for several applications such as text categorization and clustering since the Arabic language is known for its variety, richness and complex morphology. Until recently, the Bag-of-Words remains the most common method for Arabic text representation. However, it suffers from several shortcomings such as semantics deficiency and high dimensionality of feature space. Moreover, most existing methods ignore the explicit knowledge contained in

semantic vocabularies such as Arabic WordNet. To overcome these shortcomings, proposed a deep Autoencoder based representation for Arabic text categorization. It consisted of three stages: (1) Extracting from Arabic WordNet the most relevant concepts based on feature selection processes (2) Features learning via an unsupervised algorithm for text representation (3) Categorizing text using deep Autoencoder. Our method allowed for the consideration of document semantics by combining both implicit and explicit semantics and reducing feature space dimensionality. To evaluate our method, conducted several experiments on the standard Arabic dataset, OSAC. The obtained results showed the effectiveness of the proposed method compared to state-of-the-art ones [8].

Experiments was carried out and has shown that using the Autoencoder as text representation model combined with Chi-Square and classifier outperformed state-of-the-art techniques and achieved the best results by 94% and 93% for precision and F-measure, respectively. The principal advantages are: (1) Integrating explicit semantics in order to improve the quality of text vectors; (2) Modeling semantic structure within texts; (3) Reducing representation dimensionality and (4) Exploring deep learning networks for Arabic text categorization.

The process of tagging a given text or document with suitable labels is known as text categorization or classification. The aim of this work is to automatically tag a news article based on its vocabulary features. To accomplish this objective, 2 large datasets have been constructed from various Arabic news portals. The first dataset contains of 90k single-labeled articles from 4 domains (Business, Middle East, Technology and Sports). The second dataset has over 290 k multi-tagged articles. To examine the single-label dataset, we employed an array of ten shallow learning classifiers. Furthermore, we added an ensemble model that adopts the majority-voting technique of all studied classifiers. The performance of the classifiers on the first dataset ranged between 87.7% (AdaBoost) and 97.9% (SVM). Analyzing some of the misclassified articles confirmed the need for a multi-label opposed to single-label categorization for better classification results. For the second dataset, we tested both shallow learning and deep learning multi-labeling approaches. A custom accuracy metric, designed for the multi-labeling task, has been developed for performance evaluation along with hamming loss metric. Firstly, we used classifiers that were compatible with multi-labeling tasks such as Logistic Regression and XGBoost, by wrapping each in a OneVsRest classifier. XGBoost gave the higher accuracy, scoring 84.7%, while Logistic Regression scored 81.3%. Secondly, ten neural networks were constructed (CNN, CLSTM, LSTM, BILSTM, GRU, CGRU, BIGRU, HANGRU, CRF-BILSTM and HANLSTM). CGRU proved to be the best

multi-labeling classifier scoring an accuracy of 94.85%, higher than the rest of the classifies [9].

The ultimate aim of Machine Learning (ML) is to make machine acts like a human. In particular, ML algorithms are widely used to classify texts. Text classification is the process of classifying texts into a predefined set of categories based on the texts' content. It contributes to improving information retrieval on the Web. In this paper, we focus on the "Arabic" text classification since there is a large community in the world that uses this language. The Arabic text classification process consists of three main steps: preprocessing, feature extraction and ML algorithm. This paper presents a comparative empirical study to see which combination (feature extraction -ML algorithm) acts well when dealing with Arabic documents. So, we implemented one hundred sixty classifiers by combining 5 feature extraction techniques and 32 machine learning algorithms. Then, we made these classifiers open access for the benefit of the AI and NLP communities. Experiments werecarried out using a huge open dataset. The comparison study reveals that TFIDF-Perceptron is the best performing combination of a classifier [10].

5.System Propose

5.1Datasets

Arabic News Articles Dataset, SANAD: Single-Label Arabic News Articles Dataset, two datasets were used to evaluate the proposed method these datasets are: datasets context and datasets content [11].

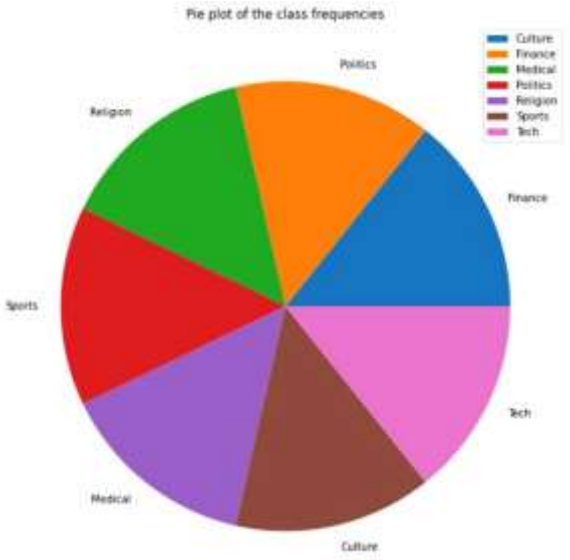

Figure 2 Pie plot of the class frequencies

5.2 Steps of system

1- Panda's data frame: It is a spreadsheet or a SQL table, where data is organized in rows and columns.

2- Remove stop words: Stop words are common words that are often removed from text data during natural language processing (NLP) tasks because they are considered to carry little meaning and don't contribute much to the overall understanding of the text. Removing stop words can help reduce the dimensionality of the data and improve the efficiency of NLP algorithms. Examples of stop words in English include words like "the", "is", "and", "a", "an", "in", "of", etc.

3- Remove punctuations, emoji and word token: To remove punctuations from Arabic text in Python, you can use the string module and regular expressions. To remove emojis from Arabic text in Python, you can use regular expressions to identify and remove them. Emojis are represented by Unicode characters, and you can match them using their Unicode ranges. If you want to remove word tokenization from a sentence, it means you want to convert the sentence back into a single string without breaking it down into individual words.

4- Label encoder: A Label Encoder is a preprocessing technique used to convert categorical data into numerical format, specifically integers. Categorical data represents variables that have discrete values, and they do not have any inherent numerical meaning. Many machine learning algorithms and statistical models require data to be in numerical form, and label encoding is one way to achieve this transformation for categorical variables.

5- Split data: Split data to x and y, x = attempt_million_cleaned and y = encodedLabel.When working with data for supervised learning tasks, we typically split the data into two parts: the features (often denoted as "X") and the corresponding labels or target variable (often denoted as "y"). The features (X) represent the independent variables, which are the inputs to the model, and the labels (y) represent the dependent variable or the target we want the model to predict.

- Test = 30%
- Training = 70%

6- TF-IDF Vectorizer: TF-IDF is a form of data retrieval that ranks the relevance of words in a text. It is based on the premise that words that appear more frequently in a document are more important to the material. The combination of Term Frequency and Inverse Document Frequency is TF-IDF. Below is the formula for TF-IDF computation, with equation 1 indicating the development of the term TF-IDF.

*TF-IDF=[Term Frequency (TF) * Inverse Document Frequency (IDF)]* (1)

IDF(t) =[log(N/(DF + 1))] (2)

*TF-IDF(t, D) = [TF(t, D) * log(N/(DF + 1))]* (3)

7- Support Vector Machine (SVM): SVM stands for Support Vector Machine, and it is a powerful and widely used supervised machine learning algorithm for classification and regression tasks. SVM is particularly effective for solving binary classification problems, where the goal is to classify data into one of two classes.
8- Pipeline: In machine learning, a Pipeline is a series of data processing steps that are chained together in a specific order to automate the workflow.

5.3 Evaluation

1- Accuracy: If a measurement is accurate, accuracy equation is depicted in Equation 4.

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

where TP determine True Positive, FP determine False Positive, TN = True Negative and FN determine False Negative.

2- Precision: Measures the proportion of predicted positive cases that were accurate. An accurate measurement has a value that agrees with other measures of the same thing.

$$\text{Precision} =\frac{TP}{TP+FP} \quad (5)$$

3- Recall: Represents the fraction of actual positive cases accurately predicted by the model. Recall measures the model's proficiency in correctly identifying positive samples. It is computed as the ratio of correctly identified positive samples to the total number of positive samples. A higher recall value signifies an increased ability to detect positive samples:

$$\text{Recall}=\frac{TP}{TP+FN} \quad (6)$$

6.Result and discussion

Table 1 presents the performance metrics of three machine learning models—Linear Support Vector Classifier (SVC), Random Forest (RF), and Logistic Regression

(LR)—across three key evaluation metrics: accuracy, precision, and recall. The results highlight the effectiveness of each model in a classification task.

Accuracy:
Both Linear SVC and Logistic Regression achieved an impressive accuracy of 0.98, indicating that 98% of the predictions made by these models were correct. The Random Forest model, while slightly lower, also demonstrated strong performance with an accuracy of 0.96. This marginal difference suggests that the Random Forest model may have a slightly higher tendency for misclassification compared to the other two models.

Precision:
The precision for both Linear SVC and Logistic Regression is 0.98, showing that these models are highly effective at correctly identifying positive instances out of the total predicted positive instances. Random Forest, with a precision of 0.96, is again slightly less precise, suggesting a few more false positives compared to the other models.

Recall:
Linear SVC and Logistic Regression both achieved a recall of 0.98, indicating their high ability to identify nearly all relevant instances (true positives) in the dataset. The Random Forest model recorded a recall of 0.96, which, while still high, suggests a slightly lower capability to capture all true positives compared to Linear SVC and Logistic Regression.

Table 1: Result of system

	Linear SVC	**Random Forest (RF)**	**Logistic Regression (LR)**
Accuracy	0.98	0.96	0.98
Precision	0.98	0.96	0.98
Recall	0.98	0.96	0.98

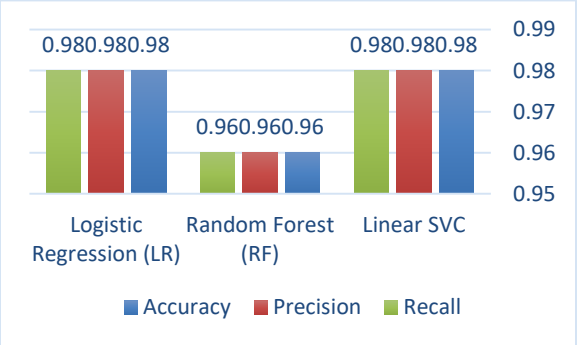

Figure 3: Result of system

Table 2: Classification of Arabic text

Arabic text	**Classification**
text_clf.predict=([' العراق مهد الحضارات '])	'Culture'
text_clf.predict ([' تلعب الرياضة دوراً مهماً في الحفاظ على المستوى الذِهني للأفراد بل وتحسينه خلال فترة التقدّم في العمر، بالإضافة إلى رفع مستوى التركيز '])	'Culture'
text_clf.predict ([' عمل البنج الموضعي على تخدير الجزء الذي يتم وضعه عليه، حيث يقوم بمنع الأعصاب من إيصال إحساس الألم من المنطقة المصابة إلى الدماغ، ويتم استخدامه في العديد من الحالات الجراحية البسيطة، مثل جراحة الأسنان التقليدية '])	'Medical'

7.Conclusion and future work

This study has demonstrated the effectiveness of using a combination of Arabic stemming, TF-IDF for feature weighting, and Convolutional Neural Networks (CNNs) for the classification of Arabic text. The proposed method achieved high accuracy, precision, and recall across various benchmarks, underscoring the potential of deep learning algorithms in addressing the complexities associated with Arabic language processing. The results indicate that deep learning approaches, particularly CNNs, can significantly enhance the accuracy of Arabic text classification tasks.

Future work will focus on several key areas to further improve the classification performance and robustness of the proposed system. Firstly, enhancing the GStem algorithm to assign special weights to frequently occurring letters and to handle additional Arabic linguistic features such as prepositions attached to words will be explored. Secondly, extending the algorithm to support other languages, potentially

making the system multilingual, is a promising direction. Additionally, experimenting with other deep learning architectures, such as Recurrent Neural Networks (RNNs) and Transformer models, may yield further improvements. Finally, the development and integration of larger and more diverse Arabic text datasets will be essential for refining and validating the model's performance across different domains and text types. These advancements will contribute to the broader application of automated text classification in various Arabic language contexts, enhancing the efficiency and accuracy of information retrieval and content analysis.

References

[1] Duwairi, R.M.: Arabic text categorization. Int. Arab J. Inf. Technol. 4(2), 125–132 (2007)
[2] Sahin, Ö.: Text Classification (2021). https://doi.org/10.1007/978-1-4842-6421-8_3
[3] Jalam, R.: Apprentissage automatique et catégorisation de textes multilingues, pp. 9–10.
[4] Boukil, Samir, et al. "Arabic text classification using deep learning technics." International Journal of Grid and Distributed Computing 11.9 (2018): 103-114.
[5] Abdulghani, Farah A., and Nada AZ Abdullah. "A survey on Arabic text classification using deep and machine learning algorithms." Iraqi Journal of Science (2022): 409-419.
[6] M. Galal, M. M. Madbouly, and A. El-Zoghby, “Classifying Arabic text using deep learning,” J. Theor. Appl. Inf. Technol., vol. 97, no. 23, pp. 3412–3422, 2019.
[7] Elnagar, R. Al-Debsi, and O. Einea, “Arabic text classification using deep learning models,” Inf. Process. Manag., vol. 57, no. 1, p. 102121, 2020, doi: 10.1016/j.ipm.2019.102121.
[8] F. El-Alami, A. El Mahdaouy, S. O. El Alaoui, & N. En-Nahnahi, “A deep autoencoder-based representation for Arabic text categorization.,” vol. 3, no. 3, pp. 381–398, 2020.
[9] El Rifai, Hozayfa, Leen Al Qadi, and Ashraf Elnagar. "Arabic text classification: the need for multi-labeling systems." Neural Computing and Applications 34.2 (2022): 1135-1159.
[10] Bouchiha, Djelloul, Abdelghani Bouziane, and Noureddine Doumi. "Machine Learning for Arabic Text Classification: A Comparative Study." Malaysian Journal of Science and Advanced Technology (2022): 163-173.
[11] Einea, Omar, Ashraf Elnagar, and Ridhwan Al Debsi. "Sanad: Single-label arabic news articles dataset for automatic text categorization." Data in brief 25 (2019): 104076.