InfoTech Spectrum: Iraqi Journal of Data Science (IJDS)
ISSN: 3007-5467, Vol 1(No.1), 42-54 (2024)
DOI: https://doi.org/10.51173/ijds.v1i1.10
https://ijds.mtu.edu.iq

EVALUATING THE IMPACT OF FEATURE EXTRACTION TECHNIQUES ON ARABIC REVIEWS CLASSIFICATION

Hawraa Fadhil Khalil [1], Mohammed Fadhil Ibrahim [2*], Hafsa Ataallah Hussein [3]

[1,2,3] *Middle Technical University (MTU)- Iraq, Technical College of Management, Iraq*

[*] *Corresponding author E-mail: mfi@mtu.edu.iq (Mohammed Fadhil Ibrahim)*

RESEARCH ARTICLE

ARTICLE INFORMATION

SUBMISSION HISTORY:

Received: 30 March 2024
Revised: 21 May 2024
Accepted: 19 June 2024
Published: 30 June 2024

KEYWORDS:

Texts Classification;
Stemming;
Feature Extraction;
Deep Learning;

ABSTRACT

With the advent of AI text-based tools and applications, the need to introduce and investigate word-processing tools has also been raised. NLP tools and techniques have developed rapidly for some languages, such as English. However, other languages, such as Arabic, still need to introduce more methods and techniques to provide more explanations. In this study, we present a sample to classify customer reviews which are written in Arabic. The data set (HARD) is used to be certified as a dataset for work. This study adopted four classifications in machine learning and deep learning (CNN, RNN, NB, LR). In addition, the texts were cleaned using data cleaning techniques, and the stemming technique was used, and three types of them were implemented (Khoja Stemmer, Snowball Stemmer, Thashaphyne Stemmer). Moreover, two methods of feature extraction were used (TF-IDF, N-gram). The results of the model provided several explanations. The best performance resulted from the use of (CNN+ Snowball Stemmer +N-gram) with accuracy (%93.5). The results of the model stated that some workbooks are sensitive to the use of different tools, and some accuracy performance can also be affected if there are different methods for extracting the features used. Either feature extraction has an impact on accuracy performance. The model also proved that colloquial Arabic could cause some limitations because different dialects can give different meanings across different regions or countries. The results of the study open the door to exploring other tools and methods to enrich natural Arabic language processing and contribute to the development of new applications that support Arabic content.

1. INTRODUCTION

Social media is become a valuable tool for learning about a variety of subjects. Creating and implementing automatic algorithms that can retrieve data and knowledge from Arabic social network posts is challenging. The International Data Corporation estimates that the quantity of digital data generated by global servers surpassed 33 zettabytes in 2018 and is projected to reach 175 zettabytes by 2025 [1]. In the Middle East countries, the number of people using the Internet has increased from 2.1% in 2005 to 24.4% in 2017 [2]. Everyone knows today how important reviews, opinions, and comments on a variety of topics are, given the rise in social media users giving their thoughts and criticism about particular services or goods [3], whether they are user-written evaluations of a specific service or product or remarks on social media [4].

Given that Arabic is the official language of 22 nations worldwide[5], Additionally, it ranks as the fourth most popular language online [6]. The majority of people find Arabic to be quite complex due to morphology and diacritical marks, among other factors . We must process these Arabic-language assessments using natural language processing (NLP), a branch of computer

DOI: https://doi.org/10.51173/ijds.v1i1.10

science that aims to improve human-machine communication [7] The application of natural language processing (NLP) can streamline and automate a great deal of business processes, particularly those involving large volumes of unstructured textual communication, such as social media chats, e-mails, and surveys. Businesses can assess their data more successfully with NLP to enable informed decision-making. This is a crucial area of study in artificial intelligence, and its applications to popular machine translation, speech recognition, public opinion research, text classification, and other areas have greatly facilitated our lives and studies [8]. Text classification is a crucial component of NLP. The categorization problem has been extensively explored in data mining, machine learning, and information retrieval, with ramifications for numerous fields.

Nonetheless, there are still many text categorization domains where this broad strategy has not yet been established, particularly for texts written in Arabic [9]. Among the world's most intricate and ancient languages is Arabic. Arabic presents difficulties for language recognition and neuro-linguistic programming since it is a fully morphological language with numerous dialects. In contrast to evaluations written in English, there are more sophisticated tools available for assessing reviews written in Arabic, despite the significance of doing so and its role in shaping business plans and strategies [10]. But Arabic is one of the trickiest languages. Therefore, in order to enhance worldwide scientific outputs, new tools and methodologies for classifying reviews in Arabic must be used. Consequently, it is crucial to provide a first-rate method for processing Arabic texts and analyzing review material, particularly in light of the growing popularity of social media and online applications. The primary aim of the research is to develop a model for the categorization of Arabic text reviews through the application of machine learning and deep learning techniques, along with various stem and feature extraction tools and an evaluation of the model's performance.

2. LITERATURE REVIEW

This part aims to discuss some important previous studies in the field of processing and classification of texts written in Arabic, which is the basic start to start this work and to understand the research topic, its requirements and objectives. We will discuss in this part some of the research related to the processing and classification of Arabic texts by the methods used in [11]. The NB algorithm was applied in this research to analyze opinions by exploring text categories and classifying them to the appropriate layer. This research dealt with the impact of using two feature extraction methods, namely (TF-IDF with classifiers, on the accuracy of classifying Arabic articles. Accuracy was used the results showed that using TF with TF-IDF improved accuracy. Ibrahim et al. (2021) evaluated the classification of theses and titles of theses in Arabic using standard classifiers (CNB, NB, MNB, GNB) [12]. They reported their best performance (84%) using the CNB classifier. [13] Zafer and others in 2023 published a study in which they used a preliminary dataset consisting of (2122) sentences and (15331) words compiled from 206 publicly available online publications to conduct emotion classification using advanced machine learning technology based on artificial neural networks, artificial neural networks were used to classify people's opinion publications, the results of comprehensive simulations indicated good accuracy. Kerimi and others proposed in 2021 a technique to help improve performance in text classification tasks using RNN, CNN algorithms, and AEDA involves only the random insertion of punctuation marks in the original text [14]. They demonstrate this by using AEDA-optimized data for training, where the models showed superior performance compared to using AEDA-optimized data in general.

A method for categorizing Arabic tweets using several deep-learning algorithms was published by researchers in 2020 [15]; the authors used the Twitter API to gather 160,870 Arabic tweets. Eight categories were created from the data set. With 10% going into testing, the authors (90%) trained and validated DL models using a dataset. The DL models' performances were highly similar to one another. In order to reduce the workload for humans, the authors proposed a model for the classification of medical texts that makes use of two novel deep-learning structures [16].

The first strategy is the hybrid deep learning model of long-term memory (QC-LSTM), while the second is the deep learning model of biGRU. Both strategies have been developed and put into practice with success. Two sets of medical textual data were used to validate the suggested methodology, and a thorough analysis was performed. The suggested deep quality control technology produced the greatest results in terms of rating accuracy. A monograph was published in [17], which dealt with the classification of a number of articles (political weblog posts in particular). Supervised machine learning was used with two feature extraction techniques (TF-IDF)in the classification process. To investigate the matter, SVM was used. Following the test, the outcomes demonstrated that TF-IDF performed better. From the studies mentioned above, we can draw some conclusions that have an impact on the classification results, regardless of whether the studies used deep learning or normal learning. First, tribal processing of texts—regardless of length—is one of the crucial steps in producing the best text that sentiment analysis techniques can comprehend, analyze, and classify. The more methods used in the processing process—such as data cleaning and feature extraction—the higher the accuracy extracted. Secondly, texts written in colloquial Arabic require more text processing techniques than texts written in traditional classical Arabic, where the more colloquial dialect, the lower the accuracy. In this research, multiple stemming techniques and feature extraction algorithms were used

3. METHODOLOGY

The majority of NLP studies generally follow a set of standard procedures that outline the entire investigation process. Fig 1 outlines our process.

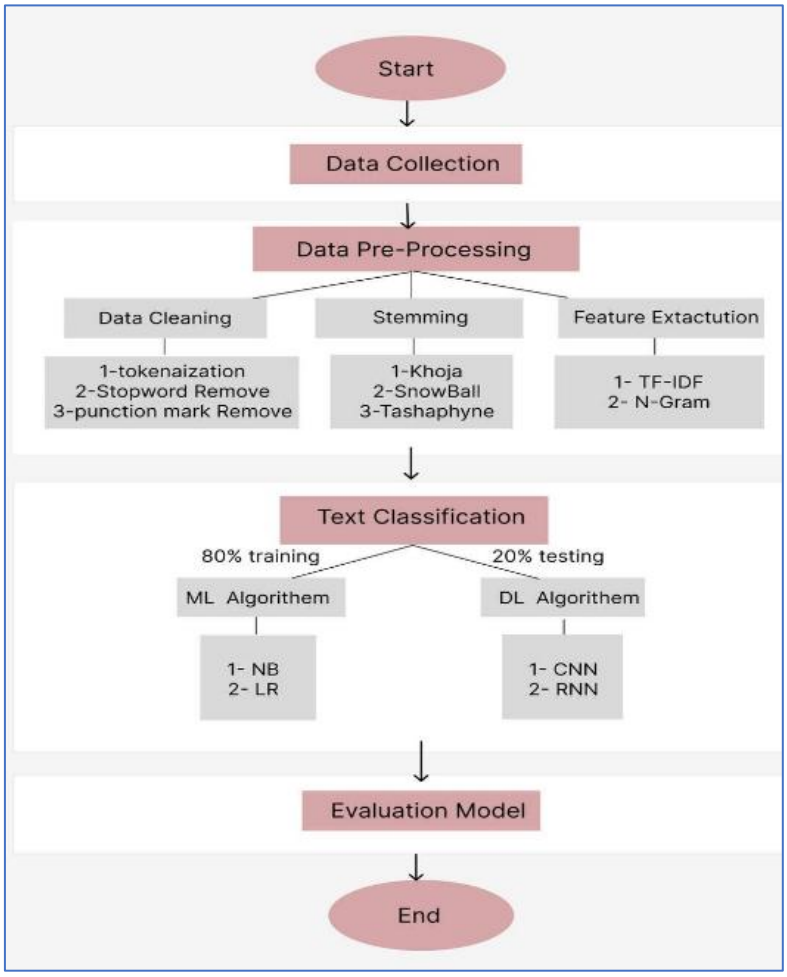

Figure 1: Research Framework

3-1 Data Collection:

Arabic-language hotel reviews can be found in the dataset utilized for this work, the Hotel Arabic Reviews Dataset (HARD) [18]. A number of studies and research projects involving Arabic language and NLP have made use of the HARD dataset [19], [20], [21]. Between June and July of 2016, this dataset was gathered from the Booking.com website. About 93,700 evaluations of positive and negative classes are included in the first balanced HARD dataset (Figure 2). In terms of Arabic, Modern Standard Arabic (MSA), the formal language, and Dialectal Arabic (DA), which is colloquial Arabic, customer reviews were inconsistent. Every region of the Arabic countries has a different definition for DA, and each country's official Arabic language has multiple variants according to its location [22].

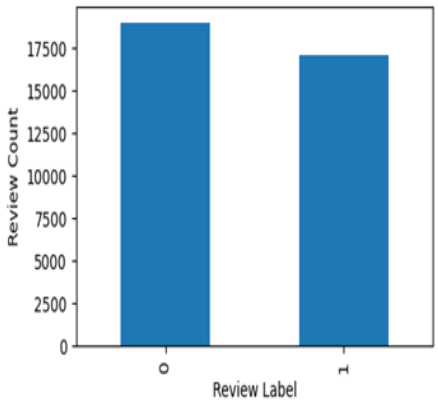

Figure 2: Dataset Description

Through the representation of the dataset, it was demonstrated that there is a significant degree of variation in the text length of the customer reviews (Figure 3). Thus, while some people record a few words, others register lengthy comments. Short texts generally suffer from NLP's drawbacks because there is less information contained in them. As a result, the data set underwent an early stage of removal of the lengthy text (more than 800 characters) and the succinct remarks (less than 100 characters). Following this stage, the dataset grows to 36098, and reviews are categorized as Positive (17106) and Negative (18992).

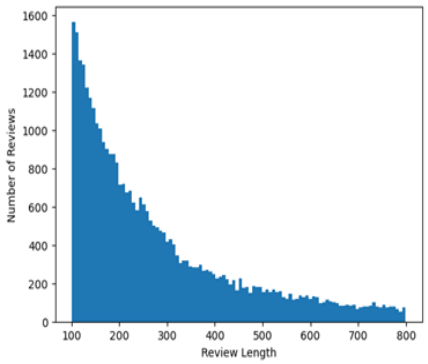

Figure 3: Customer Reviews Distribution Based on Text Length.

3-2 Data Pre-processing:

Setting up data for a classification task is one of the main steps in any NLP-related work. This process includes several steps to prepare the data for further analysis. Implementing similar methods on any text has some limitations since the quality of the processed information mainly determines its output efficiency. Arabic is one of the most complex scripts and is very sensitive to operations such as encoding and cleaning [23]. This complexity results from dealing with the diversity of dialects, their highly derived nature, and the uncertainty caused by diacritical marks.

In order for enterprise applications to deliver robust, accurate, and reliable results, data preprocessing is a necessary step in nearly all data analytics, data science, and artificial intelligence development processes [24]. Real-world data is messy, created, processed and stored by different people, organizations and software programs. As a result, records for the same entity may have different names, contain human input errors, contain duplicate data, or be missing entire fields.

In many cases, people are able to identify and solve these problems in the data they use to do their work. However, deep learning algorithms or machine learning training require automatic data preprocessing [25]. Feature engineering methods allow raw information to be reorganized into models suitable for specific methods, including processing data, modification, reduction, feature selection, and scaling. This can significantly reduce the time and computing power required to train new artificial intelligence or machine learning algorithms or compare results to them. Preprocessing techniques used for Arabic text in this study include the following:

3.2.1 Tokenization:

One of the initial phases in any NLP processing pipeline is encoding [26]. It breaks raw text into tiny word or sentence fragments called tokens. Generally, "spaces" are used to encode words, and characters such as "periods, exclamation points, and line breaks" are used to encode sentences [27]. The choice of the correct encoding method depends on the specific neurolinguistics programming (NLP) task. This method breaks down text content into its basic units, which are words.

3.2.2 Removing Stop Words:

Prepositions, character names, hyphenated nouns, and interrogative tools are examples of words that are categorized as stop words since they have no semantic relationship to the context in which they are used [28]. Stop words are words that show up frequently in the majority of documents within a particular group. It doesn't appear that these broad phrases matter when deciding which posts satisfy consumers' needs. Stated differently, these terms are unable to distinguish between favourable and unfavourable assessments because they are present in both [29]. As dimensionality reduction is crucial for the majority of machine learning problems, removing these phrases does not affect the classifier's performance because they do not affect the classification challenge itself (Table 1).

Table 1: Arabic stop words examples

Arabic Word	Meaning in English	Pronunciation of English	(Classification)
في	(In)	(Fey)	**(Prepositions)**
على	(on)	(Alaa)	
أنا	(I)	(Ana)	**(Pronouns)**
نحن	(we)	(Nahno)	
تحت	(Below)	(Taht)	**(Adverbs)**
فوق	(Above)	(Fawk)	
لماذا	(What)	(Lemaza)	**(Question)**
متى	(When)	(Mata)	
اذا	(If)	(Eza)	**(Conjunctions)**
ثم	(Then)	(Soma)	

3.2.3 Stemming:

The stem word plays a vital role in modern indexing and search engines [30]. Applications for text extraction, NLP systems, and information retrieval all depend on indexing and searching. The main goal of stemming is to combine related words and substitute [31] them in a single sentence.

For instance, the term "Histori" would take the place of both "Historical" and "History". Despite the lack of a coherent definition in the English language, the term "Histori" can nonetheless be used to classify a certain text appropriately. While stemming has its uses in classification issues, it might not always work as planned in other NLP applications. When working with scripts as complicated as Arabic, the stemming technique effectively returns words that are identical to a single root [32]. The primary objective of the root-based root is to determine the fundamental form of any given word through morphological analysis. This example demonstrates the intricacy and diversity of Arabic letters. Such words can be difficult to decode and analyze, and because Arabic does not adhere to a set orthographic style based on letters, businesses may have a high error rate. Rather, a word with a similar meaning can be clearly different from the original one [33]. Therefore, by using stemming, the past example "يلعب") and all its similar words can be returned to a united root such as ("لعب"), which means ("play"). In this study, two distinct stemmers are used, and their performance is assessed as a result.

Khoja's method is one of the most used morphological derivation algorithms [34]. The Khoja Stemmer removes a word's longest suffix and longest prefix. It then compares the remaining words to nouns (nouns) and linguistic patterns in order to determine the word's root. The algorithm evaluates the remaining piece of the word with its verbal and nominal patterns to identify the root after removing the largest prefix and suffix from the word Snowball Stemmer [35]. The Porter Stemmer algorithm has various shortcomings, which this derivation algorithm, also called the porter2 derivation algorithm, addresses. Tashaphyne Stemmer: It is a syllable and mild Arabic derivation. Its main goal is to promote light derivation by removing prefixes and suffixes and offering every possible division. It uses a modified finite state automaton to produce all of the partitions. Because, in contrast to Khoja, ESRI, Asim, and Frasa stemmers, it allows both extraction and root extraction concurrently. Tashaphyne enables utilizing a list of custom prefixes and suffixes in addition to its usual prefixes and suffixes. This allows it to handle extra features and create bespoke derivatives without requiring changes to the code .

3.2.4 Feature extraction

Feature extraction is part of the dimensionality reduction process, in which an initial raw data set is divided and reduced into more manageable groups, making the process more straightforward [40][41]. These huge data sets' numerous variables are their most crucial feature. Processing these variables takes a lot of computing resources. Consequently, By selecting and merging variables into features, feature extraction effectively minimizes the amount of data, assisting in the extraction of the best features from such vast data sets. These features are easy to work with and provide an accurate and imaginative description of the actual data set. We shall use multiple feature extraction methods in this aspect [39].

a. TF-IDF:

During the pre-processing phase, text features such as keyword information are extracted from the text using TF-IDF, stop word removal, and word segmentation [40]. A statistical technique for determining a word's meaning in a text is called TF-IDF. The significance of a word rises in direct proportion to how frequently it appears in the document, but it falls in proportion to how frequently it appears in the corpus. Some discernible words can be identified more accurately by TF-IDF. In some works, these words are used more frequently than in others. We apply the following formula to determine the TF-IDF value of each word in the lexicon of test sample K, allowing us to determine the frequencies of all the words:

$$TF - IDF = TF * IDF \qquad \dots (1)$$

Where TF represents the number of occurrences for any given term in a specific text divided by the total number of terms in a given text, IDF represents the logarithmic value of the number of texts in the dataset divided by the number of texts in the dataset.

The unique ratings for every word in the text are extracted using this formula. Higher-valued words are more selective and best differentiated between text groups. By keeping the terms that are most helpful in categorization, these words reduce the quantity of text and the number of attributes that must be computed. TF-IDF arranges words in distinct texts based on their significance, sorting them from largest to lowest. Greater-meaning words have more discriminative power for each text because words with greater weights in one text have less weight in other texts. To precisely extract the most essential text features and eliminate unnecessary features, we employ TF-IDF [41].

b. N-grams

"N-grams" are contiguous collections of "n" items—typically words or characters—drawn from a corpus of texts. These models have been crucial to both early and contemporary Natural Language Processing because they offer a simple means of capturing the statistical characteristics of the language[42]. N-gram models are especially well-suited for tasks requiring short-range contextual information because they take advantage of local dependencies within a text. When attempting to estimate a word's likelihood based on its previous context, n-gram models are frequently used in language modelling tasks. Every word in a unigram model is handled separately, but bigram models take into account word pairings. Higher-order models and trigrams can capture longer dependencies. N-gram models are helpful when more complicated models are computationally expensive, even though they provide a foundation for language modelling despite their simplicity [43].

3.3 Texts classification

One of the key components of NLP is the classification of texts. The best text classifier cannot be defined, as is well known. For instance, there is broad agreement on a standard approach for creating models, neural networks in particular, and other recognized methodologies in fields like computer vision [44]. Besides, this common approach continues to be lacking in many aspects of text classification. Since Arabic information on the Internet is growing at an ongoing rate, one of the key themes in large-scale Arabic text mining is the classification of Arabic texts. It is one of the most significant study areas, where excellent data is taken from texts, and the themes to which those texts belong are categorized, particularly when these texts [45]. The vast number of Arabic texts available on the Internet leads to scholarly problems that have recently been addressed. Researchers are trying to make use of this data by classifying the texts using methods such as data mining. The objective of this study was to apply the beneficial effects of algorithms that have been effective across various Arabic language domains. In this study, we will be covering certain algorithms, namely CNN, RNN, NB and LR.

a. Convolution Neural Network (CNN)

Although early 2D CNNs were applied extensively in computer vision, however, text classification tasks are a relatively new application for them, and they have shown superior performance than sequence-based methods [46]. Using a convolutional layer and a subsampling layer (also known as the maximum pooling layer), the CNN creates a feature map through a series of convolutions and pooling [47]. A sliding convolution window with changing kernels is used by the convolution layer of the 1D CNN to perform a 1D cross-correlation operation across the text being input from left to right. It uses a max-over-time pooling layer, which lowers the amount of features required for text encoding by using a 1D global maximum pooling layer.

b. Recurrent Neural Network (RNN)

RNNs are widely used in scenarios involving sequential data. This is because the model uses layers, which offer short-term memory. It can predict the subsequent data more accurately thanks

to this memory [48]. The period of the past data's retention is determined by its associated weight is a dynamic process [49]. Consequently, speech marking, sign sequence analysis, emotion analysis, and other applications need RNN. By far, the most significant advantage of an RNN is its capacity to communicate previous knowledge to the latest output, i.e., to connect the current production of the series with the last one. Text analysis is improved by the bidirectional repetitive neural network's ability to correlate context semantics in word processing accurately.

c- *Naive* Bayes (NB)

The text is sorted using a straightforward algorithm that considers the likelihood that certain occurrences will occur [50]. The Bayes Theorem, on which this technique is based, aids in determining the conditional probability of the events that have occurred based on the likelihood that each event will occur independently. We presume that the aim is to determine whether the supplied sentence is a good or negative comment in order to understand better how to use them in the text classification[51]. The NB model needs a training dataset, which is a collection of sorted words with their categories, much like all other machine learning models do. The probability for each category is computed using its sentences using a Bayesian equation (1,2). Based on the probability value, the algorithm determines if the text falls into the positive or negative category.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \quad \dots (2)$$

$$P(c|x) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c) \quad \dots (3)$$

Where:

P(C|X) is the probability of C given that X is True.

P(X) and P(C) are the independent probabilities of X and C.

d- Logistic regression (LR)

This method functions similarly to an NB classifier because it further predicts the likelihood that Y is related to the input variable X [52]. A single predictor binary logistic regression statistics model is given by formula (1). P is the likelihood that, given the value of "x," the dependent variable "X," which is the independent variable, will take the value 1.

$$P(X) = \frac{e^{(\beta_\circ + \beta_1 X_1 + \cdots + \beta_p X_p)}}{1 + e^{(\beta_\circ + \beta_1 X_1 + \cdots + \beta_p X_p)}} \quad \dots (4)$$

Where:

P: Probability that X =1 given x, X: dependent variable, X_p, X_1: independent variables, β: Model parameters.

It fits the parameter perfectly. "β°" and "β1" are calculated using the maximum probability method. With this approach, the probability function is maximized. After the two parameters have been assessed, the logistic function may be used to predict the probability of the target variable, p(Xi), for a given input, Xi. Since LR is a simple model, training is quick. It is capable of handling a large number of features. Despite the word regression being in its name, its range is always between 0 and 1. Hence, we can only use it for classification tasks [53]. It performs poorly on multi-category classification problems and is limited to binary classification problems.

4. EXPERIMENTS AND RESULTS DISCUSSION

The model's performance was assessed using four classifiers, as previously indicated: CNN, RNN, NB, and LR. The results of using the classifiers on three distinct stemming techniques—Khoja, Snowball, and Tashaphyne—have been used to assess the classification performance. Because every stemming technique has its protocols for stemming the Arabic language, there will be variations amongst stemmers with regard to the rooting of every word. That means that different classifiers should produce different results. The suggested model is assessed using accuracy. Generally speaking, accuracy depends on the confusion matrix, a commonly used visual aid for demonstrating how well classification algorithms work. It compares the correctly and incorrectly classified values to the actual outcomes in the test data. The accuracy assessment takes into account four variables, as shown by the following formula:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad \ldots (5)$$

Where:

TP: are the instances that the model accurately anticipated.
TN are the instances where the model anticipated an inaccurate outcome.
FP are the scenarios that the model predicts but turn out to be false.
FN: The model predicted that these cases would be false, but they are real.

Python programming tools were used to carry out all of the study processes. The cleaned data has been divided into two sets: the training set, which comprised 80% of the total, and the testing set, which contained the rest of the data. The settings are identical for all the categorization and stemming tools and techniques.

Table 2: Classification Results

Classifiers	Stemming Method	Feature extraction		
		NO FE	TF-IDF	N-Gram
CNN	No Stemming	0.923	0.913	0.924
	Khoja	0.927	0.910	0.932
	Snowball	0.929	0.914	0.935
	Tashaphyne	0.926	0.910	0.929
RNN	No Stemming	0.895	0.910	0.915
	Khoja	0.902	0.907	0.910
	Snowball	0.829	0.913	0.915
	Tashaphyne	0.907	0.905	0.909
NB	No Stemming	0.821	0.892	0.922
	Khoja	0.882	0.893	0.910
	Snowball	0.894	0.891	0.910
	Tashaphyne	0.890	0.819	0.912
LR	No Stemming	0.891	0.922	0.930
	Khoja	0.892	0.910	0.910
	Snowball	0.914	0.922	0.914
	Tashaphyne	0.901	0.911	0.910

The first experiment is conducted on the dataset using the CNN technique. As stated earlier, the classification process is performed on different processed data. Table 2 illustrates the results of the four classifiers (CNN, RNN, NB, And LR along with different stemming techniques (Khoja, Snowball, Tashaphyne) also three FE methods (TF-IDF, N-Gram). The classification performance is described for each method so that Table 2 states the results according to the text processing methods. All three stemming techniques are evaluated according to the FE method used, in addition to the (No FE), which refers to the classification without using any feature extraction.

Python programming tools were used to implement all of the study instruments and procedures. The cleaned dataset was split into two sets: the training set, which comprised 80% of

the total, and the testing set, which contained the remaining data. The configurations of all the categorization and stemming tools and techniques are identical. Four distinct classifiers were employed in addition to the three shown in Table 2. The process began with encoding the reviews that were part of the dataset and using the primary tools for text cleaning and encoding.

In deep learning algorithms, for the (CNN) classifier, the best performance was obtained when using with. (N-Gram, Snowball) it is worth mentioning that the use of(N-Gram) can overcome other feature extraction methods when using rounding algorithms, which means that whenever rounding algorithms are used, (N-Gram) can perform well. Despite this, (N-Gram) can give the best performance to our model, but we can see that the model has some stability when dealing with testing in case feature extraction algorithms are not used. As for the RNN algorithm, it is the least performing compared to other algorithms, as there is a decrease in performance when using (Snowball+ no FE) with an accuracy of (0.829). The highest performance of the (RNN) classifier comes from. (Snowball+ N-Gram) and here, we note that the highest and lowest performance comes from using the same stem. Still, the difference was related to feature extraction algorithms, meaning that (RNN) has a high sensitivity to feature extraction algorithms, which has a significant impact on the results according to the results reached.

As for the machine learning algorithms, the results were, for the (LR) classifier, the highest accuracy was recorded (0.89), and the best accuracy achieved when using the rooting algorithms was at (Snowball Stemmer), which reached (0.91), while for the (Khoja, Tashaphyne)algorithms, the performance was similar by an accuracy of (0.90 and 0.89), respectively, as for the (NB) algorithm, it got an accuracy of (0.82), after which the performance increased to accurately (0.89) at the (tashaphyne, snowball) algorithms.

5. CONCLUSION

Based on the results gained from the study's experiments, it is noteworthy to observe that the performance of Arabic text classification still needs more enhancement, and this is due to the Arabic text complexity. The performance of using different stemming techniques has a tiny impact on the performance results, and this is also related to the nature of the dataset, where there is a big part of the comments were written in a colloquial style that has clear rules. Some methods are sensitive to using stemmers; others are not. FE, on the other hand, sometimes gives lower accuracy, which means that the text may lose some essential features when manipulating it using FE. The process of stemming can also affect the text quality and cause some features to be lost, and this explains some fluctuations in the results. Despite all the mentioned limitations, our model presented significant results by implementing widespread techniques of FE and stemming. The results of this study open the doors toward investigating more methods and techniques that deal with Arabic text classification. The implementation of word embedding methods might overcome the dialectal limitation by building some corpora for some Arabic dialects, which can serve in presenting more advanced models.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest regarding the publication of this paper

REFERENCES

[1] M. M. Almanea, "Automatic Methods and Neural Networks in Arabic Texts Diacritization: A Comprehensive Survey," IEEE Access, vol. 9, no. Dl, pp. 145012–145032, 2021, doi: 10.1109/ACCESS.2021.3122977.

[2] F. Habibi and M. A. Zabardast, "Digitalization, education and economic growth: A comparative analysis of Middle East and OECD countries," Technol Soc, vol. 63, 2020, doi: 10.1016/j.techsoc.2020.101370.

[3] M. B. Ressan and R. F. Hassan, "Naïve-Bayes family for sentiment analysis during COVID-19 pandemic and classification tweets," Indonesian Journal of Electrical Engineering and Computer Science, vol. 28, no. 1, 2022, doi: 10.11591/ijeecs.v28.i1.pp375-383.

[4] R. A. Bagate and R. Suguna, "Sarcasm detection of tweets without #sarcasm: Data science approach," Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 2, 2021, doi: 10.11591/ijeecs.v23.i2.pp993-1001.

[5] M. O. Hegazi, Y. Al-Dossari, A. Al-Yahy, A. Al-Sumari, and A. Hilal, "Preprocessing Arabic text on social media," Heliyon, vol. 7, no. 2, 2021.

[6] R. Obiedat, D. Al-Darras, E. Alzaghoul, and O. Harfoushi, "Arabic aspect-based sentiment analysis: A systematic literature review," IEEE Access, vol. 9, pp. 152628–152645, 2021.

[7] H. Elfaik and E. H. Nfaoui, "Deep Bidirectional LSTM Network Learning-Based Sentiment Analysis for Arabic Text," Journal of Intelligent Systems, vol. 30, no. 1, pp. 395–412, 2021, doi: 10.1515/jisys-2020-0021.

[8] M. F. Ibrahim and A. Al-Taei, "Based Document Classification for Arabic Theses and Dissertations," in Advances in Data and Information Sciences: Proceedings of ICDIS 2021, Springer, 2022, pp. 189–203.

[9] I. Guellil, H. Saâdane, F. Azouaou, B. Gueni, and D. Nouvel, "Arabic natural language processing: An overview," Journal of King Saud University-Computer and Information Sciences, vol. 33, no. 5, pp. 497–507, 2021.

[10] M. O. Hegazi, Y. Al-Dossari, A. Al-Yahy, A. Al-Sumari, and A. Hilal, "Preprocessing Arabic text on social media," Heliyon, vol. 7, no. 2, p. e06191, 2021, doi: 10.1016/j.heliyon.2021.e06191.

[11] D. H. Abd, A. T. Sadiq, and A. R. Abbas, "Classifying political arabic articles using support vector machine with different feature extraction," in International Conference on Applied Computing to Support Industry: Innovation and Technology, Springer, 2019, pp. 79–94.

[12] M. F. Ibrahim, M. A. Alhakeem, and N. A. Fadhil, "Evaluation of Naïve Bayes Classification in Arabic Short Text Classification," Al-Mustansiriyah Journal of Science, vol. 32, no. 4, pp. 42–50, 2021, doi: 10.23851/mjs.v32i4.994.

[13] D. H. Abd, W. Khan, B. Khan, N. Alharbe, D. Al-Jumeily, and A. Hussain, "Categorization of Arabic posts using Artificial Neural Network and hash features," J King Saud Univ Sci, vol. 35, no. 6, p. 102733, 2023, doi: 10.1016/j.jksus.2023.102733.

[14] A. Karimi, L. Rossi, and A. Prati, "AEDA: An Easier Data Augmentation Technique for Text Classification," Findings of the Association for Computational Linguistics, Findings of ACL: EMNLP 2021, pp. 2748–2754, 2021, doi: 10.18653/v1/2021.findings-emnlp.234.

[15] A. M. Bdeir and F. Ibrahim, "A framework for arabic tweets multi-label classification using word embedding and neural networks algorithms," in Proceedings of the 2020 2nd International Conference on Big Data Engineering, 2020, pp. 105–112.

[16] S. K. Prabhakar, "Models with Multihead Attention," vol. 2021, 2021.

[17] D. H. Abd, A. T. Sadiq, and A. R. Abbas, "Political articles categorization based on different naïve bayes models," in International Conference on Applied Computing to Support Industry: Innovation and Technology, Springer, 2019, pp. 286–301.

[18] A. Elnagar, Y. S. Khalifa, and A. Einea, "Hotel Arabic-reviews dataset construction for sentiment analysis applications," Intelligent natural language processing: Trends and applications, pp. 35–52, 2018.

[19] H. El Rifai, L. Al Qadi, and A. Elnagar, "Arabic text classification: the need for multi-labeling systems," Neural Comput Appl, vol. 34, no. 2, 2022, doi: 10.1007/s00521-021-06390-z.

[20] Y. S. and E. A. Elnagar Ashraf and Khalifa, "Hotel Arabic-Reviews Dataset Construction for Sentiment Analysis Applications," in Intelligent Natural Language Processing: Trends and Applications, A. E. and T. F. Shaalan Khaled and Hassanien, Ed., Cham: Springer International Publishing, 2018, pp. 35–52. doi: 10.1007/978-3-319-67056-0_3.

[21] Hawraa Fadhil Khelil, Mohammed Fadhil Ibrahim, Hafsa Ataallah Hussein, and Raed Kamil Naser, "Evaluation of Different Stemming Techniques on Arabic Customer Reviews," Journal of Techniques, vol. 6, no. 1, pp. 103–111, Feb. 2024, doi: 10.51173/jt.v6i1.2313.

[22] S. Alyami, A. Alhothali, and A. Jamal, "Systematic literature review of Arabic aspect-based sentiment analysis," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 9, pp. 6524–6551, 2022.

[23] N. Boudad, R. Faizi, R. Oulad Haj 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, doi: https://doi.org/10.1016/j.asej.2017.04.007.

[24] H. J. Aleqabie, M. S. Sfoq, R. A. Albeer, and E. H. Abd, "A Review Of Text Mining Techniques: Trends, and Applications In Various Domains," Iraqi Journal for Computer Science and Mathematics, vol. 5, no. 1. 2024. doi: 10.52866/ijcsm.2024.05.01.009.

[25] A. Oussous, A. A. Lahcen, and S. Belfkih, "Impact of Text Pre-processing and Ensemble Learning on Arabic Sentiment Analysis," Proceedings of the 2nd International Conference on Networking, Information Systems \& Security, 2019.

[26] B. Jurish and K.-M. Würzner, "Word and Sentence Tokenization with Hidden Markov Models," Journal for Language Technology and Computational Linguistics, vol. 28, no. 2, pp. 61–83, 2013, doi: 10.21248/jlcl.28.2013.176.

[27] Z. A. Abutiheen, A. H. Aliwy, and K. B. S. Aljanabi, "Arabic text classification using master-slaves technique," J Phys Conf Ser, vol. 1032, no. 1, 2018, doi: 10.1088/1742-6596/1032/1/012052.

[28] A. Alajmi, E. M. Saad, and R. R. Darwish, "Toward an ARABIC stop-words list generation," Int J Comput Appl, vol. 46, no. 8, pp. 8–13, 2012.

[29] I. A. El-Khair, "Effects of Stop Words Elimination for Arabic Information Retrieval: A Comparative Study," pp. 1–15, 2017.

[30] T. Kanan, O. Sadaqa, A. Almhirat, and E. Kanan, "Arabic light stemming: A comparative study between p-stemmer, khoja stemmer, and light10 stemmer," in 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), IEEE, 2019, pp. 511–515.

[31] M. Alhawarat, H. Abdeljaber, and A. Hilal, "Effect of Stemming on Text Similarity for Arabic Language at Sentence Level," PeerJ Comput Sci, vol. 7, May 2021, doi: 10.7717/peerj-cs.530.

[32] S. Bahassine, A. Madani, and M. Kissi, "Arabic text classification using new stemmer for feature selection and decision trees," Journal of Engineering Science and Technology, vol. 12, no. 6, pp. 1475–1487, 2017.

[33] H. A. Almuzaini and A. M. Azmi, "Impact of Stemming and Word Embedding on Deep Learning-Based Arabic Text Categorization," IEEE Access, vol. 8, pp. 127913–127928, 2020, doi: 10.1109/ACCESS.2020.3009217.

[34] F. E. Zamani, K. Umam, W. D. I. Azis, and W. S. Abdillah, "Analysis and implementation of computer-based system development of stemming algorithm for finding Arabic root word," J Phys Conf Ser, vol. 1402, no. 6, 2019, doi: 10.1088/1742-6596/1402/6/066030.

[35] A. M. Alayba, V. Palade, M. England, and R. Iqbal, "Improving Sentiment Analysis in Arabic Using Word Representation," 2nd IEEE International Workshop on Arabic and Derived Script Analysis and Recognition, ASAR 2018, pp. 13–18, 2018, doi: 10.1109/ASAR.2018.8480191.

[36] X. Li, Z. Li, H. Qiu, G. Hou, and P. Fan, "An overview of hyperspectral image feature extraction, classification methods and the methods based on small samples," Applied Spectroscopy Reviews, vol. 58, no. 6. 2023. doi: 10.1080/05704928.2021.1999252.

[37] D. P. Tian, "A review on image feature extraction and representation techniques," International Journal of Multimedia and Ubiquitous Engineering, vol. 8, no. 4. 2013.

[38] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in Proceedings of 2014 Science and Information Conference, SAI 2014, 2014. doi: 10.1109/SAI.2014.6918213.

[39] M. Avinash and E. Sivasankar, "A study of feature extraction techniques for sentiment analysis," in Advances in Intelligent Systems and Computing, 2019. doi: 10.1007/978-981-13-1501-5_41.

[40] X. Chen, Y. Xue, H. Zhao, X. Lu, X. Hu, and Z. Ma, "A novel feature extraction methodology for sentiment analysis of product reviews," Neural Comput Appl, vol. 31, pp. 6625–6642, 2019.

[41] R. Ahuja, A. Chug, S. Kohli, S. Gupta, and P. Ahuja, "The impact of features extraction on the sentiment analysis," Procedia Comput Sci, vol. 152, pp. 341–348, 2019.

[42] J. Mutinda, W. Mwangi, and G. Okeyo, "Lexicon-pointed hybrid N-gram Features Extraction Model (LeNFEM) for sentence level sentiment analysis," Engineering Reports, vol. 3, no. 8, p. e12374, 2021.

[43] J. Mutinda, W. Mwangi, and G. Okeyo, "Lexicon-pointed hybrid N-gram Features Extraction Model (LeNFEM) for sentence level sentiment analysis," Engineering Reports, vol. 3, no. 8, 2021, doi: 10.1002/eng2.12374.

[44] T. Kanan and E. A. Fox, "Automated arabic text classification with P-S temmer, machine learning, and a tailored news article taxonomy," J Assoc Inf Sci Technol, vol. 67, no. 11, pp. 2667–2683, 2016.

[45] W. Alabbas, H. M. Al-Khateeb, and A. Mansour, "Arabic text classification methods: Systematic literature review of primary studies," Colloquium in Information Science and Technology, CIST, vol. 0, no. x, pp. 361–367, 2016, doi: 10.1109/CIST.2016.7805072.

[46] S. Bodapati, H. Bandarupally, R. N. Shaw, and A. Ghosh, "Comparison and analysis of RNN-LSTMs and CNNs for social reviews classification," Advances in Applications of Data-Driven Computing, pp. 49–59, 2021.

[47] D. Alsaleh and S. Larabi-Marie-Sainte, "Arabic Text Classification Using Convolutional Neural Network and Genetic Algorithms," IEEE Access, vol. 9, pp. 91670–91685, 2021, doi: 10.1109/ACCESS.2021.3091376.

[48] M. Ahmed, P. Chakraborty, and T. Choudhury, "Bangla document categorization using deep RNN model with attention mechanism," in Cyber Intelligence and Information Retrieval: Proceedings of CIIR 2021, Springer, 2022, pp. 137–147.

[49] J. Du, C.-M. Vong, and C. L. P. Chen, "Novel efficient RNN and LSTM-like architectures: Recurrent and gated broad learning systems and their applications for text classification," IEEE Trans Cybern, vol. 51, no. 3, pp. 1586–1597, 2020.

[50] C. Zong, R. Xia, and J. Zhang, "Text Classification," in Text Data Mining, Springer, 2021, pp. 93–124.

[51] J. Ababneh, "Application of Naïve Bayes, Decision Tree, and K-Nearest Neighbors for Automated Text Classification," Mod Appl Sci, vol. 13, no. 11, p. 31, 2019, doi: 10.5539/mas.v13n11p31.

[52] H. El Rifai, L. Al Qadi, and A. Elnagar, Arabic Multi-label Text Classification of News Articles, vol. 1339, no. March. Springer International Publishing, 2021. doi: 10.1007/978-3-030-69717-4_41.

[53] A. Yousaf et al., "Emotion recognition by textual tweets classification using voting classifier (LR-SGD)," IEEE Access, vol. 9, pp. 6286–6295, 2020.