Journal of Techniques, ISSN: 2708-8383, Vol. 6, No. 2, 2024, Pages 1-8
DOI: https://doi.org/10.51173/jt.v6i2.2313

*JOURNAL OF TECHNIQUES*

Journal homepage: *http://journal.mtu.edu.iq*

*RESEARCH ARTICLE - ENGINEERING (MISCELLANEOUS)*

# Evaluation of Different Stemming Techniques on Arabic Customer Reviews

**Hawraa Fadhil Khelil[1], Mohammed Fadhil Ibrahim[2*], Hafsa Ataallah Hussein[1], Raed Kamil Naser[2]**

[1]Technical College of Management - Baghdad, Middle Technical University, Baghdad, Iraq

[2]School of Computer Science, Universiti Sains Malaysia (USM), Penang, Malaysia

[*] Corresponding author E-mail: mohammad63@student.usm.my

**Article Info.**

*Article history:*

Received
30 November 2023

Accepted
19 January 2024

Publishing
30 June 2024

**Abstract**

Customer opinions and reviews play a vital role in marketing expansion. Big companies worldwide assign a lot of their efforts to analyzing customers' feedback to keep track of their needs. Natural Language Processing (NLP) is widely used to analyze such review texts. Arabic customer analysis and classification also began to gain researchers' attention due to the wide range of Arabic language speakers. Working with the Arabic language is a very challenging task because of the orthographic nature of Arabic. Also, customers often write their reviews dialectically, diverting from standard Arabic. This study presents a method to classify Arabic customer reviews using four classifiers (K-nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic Regression (RL), and Naïve Bayes (NB)). The classification phase uses three stemming techniques (Snowball, Khoja, and Tashaphyne). The HARD dataset is adopted to perform the experiments. The results stated that the stemming methods could enhance classification performance despite the complexity of Arabic scripts and dialects, where the best accuracy of the results was 91% when using SVM and LR with Snowball Stemmer. This work sheds light on utilizing and investigating more machine learning (ML) techniques and evaluating the results.



Publisher: Middle Technical University

**Keywords**: NLP; KNN; NB; LR; Snowball Stemmer; Khoja Stemmer; Tashaphyne Stemmer.

## 1. Introduction

Social media has emerged as a valuable resource for learning about various topics. It is challenging to develop and put into use automated algorithms that can extract knowledge and information from the Arabic language on social media networks. The amount of global digital data produced by global servers surpassed 33 zettabytes in 2018, and 175 zettabytes are predicted to be generated by 2025, according to the International Data Corporation [1]. Globally, there are approximately 4 billion internet users and applications; in the Middle East, the number of users has grown from 147 million to 164 million during the previous few years [2]. With the increasing number of social media users sharing their opinions or leaving reviews or feedback about services or products [3], it is no secret to anyone today the role played by reviews, opinions or feedback on various things, whether they are comments on social media or user-written reviews about a particular service or product [4, 5]. Since Arabic is an official language in 22 countries around the world [6-8], it is also the 4th most used language on the Internet. For most people, Arabic is considered super complex for several reasons, such as morphology and diacritics[9, 10]. To understand these reviews written in Arabic, we need to process them using NLP techniques, an area of computer science that aims to facilitate communication between machines and humans [11, 12].

Many corporate operations can be made simpler and more automated by utilizing (NLP), particularly those that involve a lot of unstructured text, such as emails, surveys, social media chats, etc. With NLP, businesses may more effectively evaluate their data to support wise decision-making. It is an essential topic in artificial intelligence, and it has brought great convenience to our lives and studies, including popular machine translation, speech recognition, public opinion analysis, text classification, etc. [5]. One essential part of NLP is the classification of texts. With implications for many different fields, the classification problem has been thoroughly studied in data mining, machine learning, and information retrieval. However, this generic approach has yet to be developed in many areas of text classification, especially within Arabic-language texts. Arabic is one of the world's oldest and most complex languages. Because it is a complete morphological language with numerous dialects, Arabic poses challenges for language identification and NLP. For instance, let's consider the word (عقد), if it come individually, then there are different connotations in meaning like (عَقَدَ – عَقَّدَ - عُقَّدْ عُقْد – عِقد - عَقْد -). (Necklace –Decade –Contract – Held – Complicated –Knots). So, if the Arabic word is written out of diacritics, then it will be ambiguous and have several potential meanings, so the context is very important in such cases. Despite the importance of analyzing customer reviews and the role of this process in determining the plans and strategies of commercial organizations, there is a shortage of advanced tools to analyze Arabic reviews compared to English ones [1]. However, Arabic is among the most complex languages. Therefore, it is necessary to utilize more tools and methods to classify reviews in Arabic to support the international scientific output [13]. Therefore, providing an excellent way to process Arabic texts and analyze the content of reviews is essential, particularly considering the expanding usage of social media and web apps. The study's main objective is to provide a model for classifying reviews of Arabic texts using machine learning techniques and some stemming tools and evaluate the model's performance.

| **Nomenclature & Symbols** | | | |
|---|---|---|---|
| NLP | Natural Language Processing | LR | Logistic Regression |
| SVM | Support Vector Machine | KNN | K-nearest Neighbor |
| NB | Naïve Bayes | | |

The structure of this research is organized as follows: Section 2 is dedicated to describing the most related work, followed by the methodology of the study, where a demonstration of the whole process of the word is displayed. Lastly, a number of conclusions and future trends are listed.

## 2. Related Work

A proposal to classify Arabic tweets using group learning was presented by [14]. Using the Twitter search API, a dataset of 500 tweets was gathered and evenly separated into six groups: public, sports, politics, technology, culture, and sports. The classification was carried out utilizing NB, J48, and Sequential Minimal Optimization (SMO). In [15], three methods have been applied to group learning: stacking each workbook, bagging, and reinforcement. Ten-fold cross-checking revealed that the combined techniques increased each classifier's accuracy independently. The best accuracy of the bagging method was reported by a noticeable percentage of SMOs, who achieved the highest accuracy using the reinforcement and stacking algorithms, respectively.

By examining the categories of the text and categorizing it into the appropriate class (reformist, conservative, or revolutionary) along with the term frequency-inverse document frequency (TF-IDF) for feature extraction (FE), Naive Bayes (NB) classification was presented in [16] to analyze opinions. The results showed that using TF with TF-IDF improved the accuracy to a good ratio. In [17], the researchers conducted artificial neural network-based emotion classification using a dataset of 2,122 sentences from 206 Arabic documents. They also employed this method to categorize people's opinion posts into three groups: conservative, revolutionary, and reform. The results of comprehensive simulations indicated excellent accuracy in the classification of categories. It has been demonstrated that adding discriminating characteristics to a disclosure form for political orientation articles in Arabic greatly improves the performance of the suggested approach. In [18], three different classifiers were tested: Support Vector Machine (SVM), Gaussian naive Bayes (GNB), and Random Forest (RF). Each classifier adjusted the hyper parameters of the models, which comprised a dataset of about 35,600 Arabic tweets that were manually annotated for experiments. In terms of statistics, RF and SVM performed similarly with TF-IDF and stemming, but the performance was lower with word embeddings. In another study presented in [19], four common algorithms were investigated to assess the effects of feature selection techniques. Standard Arabic text datasets known as Saudi Press Agency (SPA) were compared to SVM, NB, K-Nearest Neighbors (KNN), and Decision Tree (DT). The results showed that the three feature selection methods often improve classification accuracy by removing irrelevant features.

In [11] and [20], a classification of Arabic theses and dissertation titles was evaluated using standard classifiers. The findings showed that the sensitivity and complexity of the Arabic language make it extremely challenging to classify Arabic short texts. Three techniques were employed to categorize Arabic tweets: CNN, DL, and RNN [21]. The authors used the Twitter API to gather 160,870 Arabic tweets. Eight areas comprise the data set: football, basketball, traffic accidents, crimes, vocalists, beauty and fashion, technology, and the economy. Excluding traffic accidents with 8,600 tweets, the number of tweets for the remaining categories was more balanced and ranged from 20 thousand to 24 thousand. The performance of the DL models was very close. From all of the above, we can come to some conclusions that affect the classification results, whether it is normal learning or deep learning, that the processing of texts, whether long or short, is one of the important processes to get the best text that emotion analysis techniques can understand, analyze and classify, where the more the processing process is developed and based on the use of more methods in terms of data cleaning or feature extraction, the higher the extracted accuracy, the texts written in the colloquial dialect require the use of text processing techniques more than those written in formal Arabic, where the more colloquial dialect, the lower the accuracy. This study presents another experiment using different methods to process texts written in Arabic but also in the context of dialect. Additionally, we measure performance by analyzing the role of different stemming methods and evaluating the results.

## 3. Methodology

In general, most studies dealing with NLP have common steps depicting the whole research process. Fig. 1 describes our methodology.

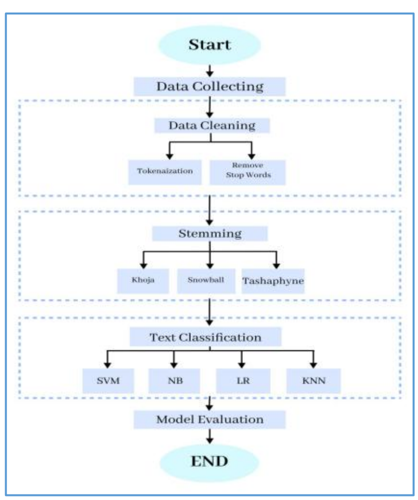


Fig. 1. Research Framework

*3.1. Data collection*

The dataset of this study is known as the Hotel Arabic Reviews Dataset (HARD), which has Arabic-language hotel reviews [22]. The HARD dataset has been utilized in many studies and research involved with NLP and Arabic language such as ([22–27]). This dataset was collected from the Booking.com site during the period (June- July 2016). The initial balanced HARD dataset has 93,700 reviews of positive and negative classes. The Arabic language used in customer reviews in the dataset was a combination of Modern Standard Arabic (MSA), which is the formal form of the language, and Dialectal Arabic (DA), which is colloquial Arabic. Every region of the Arabic countries has a distinct form for a given word in official Arabic, which varies depending on the country [28].

By representing the dataset, it was shown that the length of the review written by the customer has a high level of variation regarding the text length. Thus, some users leave long comments while others write just a few words. NLP, in general, has shortcomings when the text is short; less text means less information. Therefore, an early step was performed on the data set: removing the concise comments (less than 100 characters) and the very long text (more than 800 characters). After completing this step, the dataset consists of 36,098 reviews divided into two categories: Negative (18992) and Positive (17106), as described in Figs. 2 and 3.

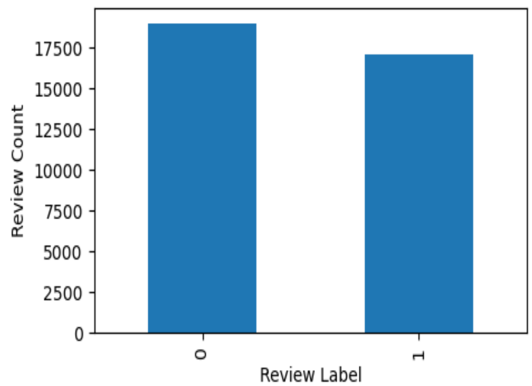


Fig. 2. Dataset Description

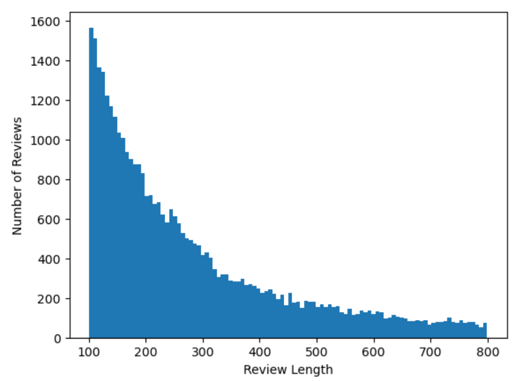


Fig. 3. Customer Reviews Distribution Based on Text Length

*3.2. Data pre-processing*

One of the main processes of any NLP-related task is preparing the data for a machine learning task. This process involves some procedures that aim to prepare the data for further analysis. For any text, there are some limitations to performing such functions because the performance quality mainly relies on the quality of processed data. Therefore, the pre-processing steps are very critical. As mentioned before, Arabic is considered one of the most complex scripts and has a high sensitivity against cleaning and encoding tasks [30]. This complexity is also increased when dealing with the variety of dialects, their highly derivative nature, and the ambiguity caused by diacritics. For enterprise applications, nearly all forms of data analysis, data science, and artificial intelligence development need data pre-processing to produce robust, accurate, and dependable findings. Real-world data is chaotic and is produced, handled, and retained by a variety of people, organizations, and software programs.

Consequently, the dataset can have different names for the same entity, contain human entry errors, duplicate data, or be missing entire fields. In many cases, humans can recognize and fix these issues in the data they utilize for their work. Nevertheless, automatic data pre-processing is required for deep learning algorithms or machine learning training [31]. Restructuring raw data into a model appropriate for specific algorithms is made possible by feature engineering techniques that include feature processing, transformation, reduction, selection, and scaling [32]. This can drastically cut down on the amount of processing power and time needed to train and evaluate a new AI or machine learning system. In this research, the pre-processing methods for Arabic texts such as the following:

3.2.1. Tokenization

Tokenization is one of the first steps in any NLP processing path [33]. It divides raw text into tiny fragments of words or sentences called tokens. In general, the “space” is used to carry out word encoding, and characters such as “dots, exclamation mark and newline character are used to encode the sentence". The choice of the correct coding method depends on the specific neuro-linguistic programming (NLP) task at hand [31]. This approach dissects textual content into its basic units, namely, words.

3.2.2. Removing Stop Words

Words classified as stop words are not semantically significant to the context in which they are used, such as prepositions, sign names, hyphenated nouns, and interrogative tools [34]. Terms that appear frequently in most documents within a given group are known as stopwords. These frequent terms may not seem significant in determining which posts fit the user's requirements. In other words, these words cannot contribute to the differentiation between positive and negative reviews since they exist in both [35]. Therefore, eliminating these phrases won't impact the classifier's performance because they do not affect the classification task, in addition to dimensionality reduction, which is important in most ML tasks (see Table 1).

Table 1. Sample Stop words in the Arabic language

| English Meaning | Arabic Word | | Category |
|---|---|---|---|
| In | Fey | في | Prepositions |
| on | Alaa | على | |
| to | Ela | الى | |
| I | Ana | أنا | Pronouns |
| we | Nahno | نحن | |
| Below | Tahit | تحت | Adverbs |
| Above | Fawk | فوق | |
| Now | Alaaan | الان | |
| Since | Monzo | منذ | |
| What | Lemaza | لماذا | Question |
| When | Mata | متى | |
| If | Eza | اذا | Conjunctions |
| Then | Soma | ثم | |
| except | Adaa | عدا | |

3.2.3. Stemming

The stem word is a crucial component that modern indexing and search engines support [36]. Information retrieval systems, NLP systems, and text extraction applications include indexing and searching. The main idea of stemming is to unite similar words and replace them with a single one. For example, both words (Historical, History) would be replaced by the word (Histori). The critical point is that the possibility of giving a similar meaning appears whenever such a form of the word appears. Even though the word in the past example (Histori) is meaningless in the English Language, it can still be used to put a particular text in its desired category, hence with classification problems, stemming can do a good job, but for other NLP applications, it may be not giving the desired results. In Arabic, the stemming process returns similar words to a single root, which is an effective technique when dealing with such complex scripts [37]. The main goal of root-based stemming is to extract the primary form of any given word by performing a morphological analysis of the word. An example of the "يلع" which means (Play) in English, can have multiple forms like {"لعبة"→"Game", "ملعب"→"Stadium", "لاعب"→"Player", "لاعبات"→"Players (Females)", "لاعبين"→"Players (Males)", "لاعبان"→" Two Players (Males)", "لاعبتين"→"Two Players (Females)"}, and so on. This example shows the diversity and complexity of Arabic scripts. Encoding and analyzing such words can be hectic, and companies with a high error rate since Arabic does not follow a fixed letter-based orthographic style. Instead, a word with a related meaning can noticeably differ from the original one [38, 39]. Therefore, by using stemming, the past example "يلع") and all its similar words can be returned to a united root such as ("لعب"), which means ("play"). This study involves using two different stemmers and evaluating their performance accordingly. Khoja Stemmer: One of the most used morphological derivation algorithms is Khoja's algorithm [30, 34]. The Khoja Stemmer eliminates the longest suffix and longest prefix of a word. To find the word's root, it then compares the remaining words to nouns (nouns) and linguistic patterns. After eliminating the biggest prefix and suffix from the word, the algorithm compares the remaining portion of the word with its verbal and nominal patterns to determine the root [40-42]. Snowball Stemmer: This derivation algorithm, sometimes referred to as the Porter2 derivation algorithm, is an improvement on the Porter Stemmer algorithm because several of its flaws have been addressed. Tashaphyne Stemmer: It is a light Arabic derivative and syllable [43]. It primarily encourages light derivation, eliminating suffixes and prefixes and providing all conceivable divisions. It creates all partitions by utilizing a modified finite state automaton because it offers both extraction and root extraction simultaneously. In contrast to Khoja, ESRI, Asim, and Frasa stemmers. In addition to having prefixes and suffixes by default, Tashaphyne also supports using a list of custom prefixes and suffixes, enabling it to handle additional features and generate custom derivatives without requiring code modifications [44-46].

*3.3. Classification of texts*

One of the essential uses of NLP is text classification [47]. It is well recognized that defining the ideal text classifier is impossible to attain since the process of classification depends on different criteria, like the type of texts and the purpose of classification. There is widespread consensus on a standard approach for creating models, neural networks, and other accepted techniques in fields like computer vision [48]. Otherwise, text classification still lacks this general method in many areas. The classification of Arabic texts is one of the important topics in large-scale Arabic text mining due to the steady increase in Arabic content on the Internet. It is one of the research topics of great importance, where high-quality information is extracted from the texts and the topics to which those texts belong to be classified, especially when these texts are large and cannot be classified manually [49]. The current research problems are related to the huge number of Arabic texts found on the Internet, which is constantly increasing, and researchers are working to solve and benefit from this data by applying some data mining techniques to classify them. This research aimed to utilize the benefits of algorithms proven in other areas of the Arabic language [50]. Some of the algorithms we will present in this research are SVM, NB, LR, and KNN.

3.3.1. Support Vector Machine (SVM)

In order to create the best model possible for additional data, supervised learning models called SVM classifiers search for the optimum hyperplane to split two different data classes -in our instance, reviews. Vapnik produced the SVM approach [51], which has shown very high accuracy in text categorization and pattern recognition [52, 53]. Better performance with fewer samples and faster speed are its two key advantages. Because of this, the approach is practical for classifying text issues.

3.3.2. Naive Bayes (NB)

It is a simple algorithm that classifies text based on the probability of occurrence of events [49]. This algorithm is based on Bayes Theorem, which helps to find the conditional probabilities of the events that have occurred based on the probabilities of each event occurring separately. To further understand how to use it in text classification, we assume that the task is to determine whether the given sentence is a negative or positive comment [54]. Like all machine learning models, the NB model also requires a training dataset containing a set of labeled sentences with their categories. Using a Bayesian equation (1, 2), the probability for each category is calculated with its sentences. The algorithm determines whether the sentence belongs to the positive or negative category based on the probability value.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \tag{1}$$

$$P(c|x) = P(x_1|c) \times P(x_2|c) \times \ldots \times P(x_n|c) \times P(c) \tag{2}$$

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.

3.3.3. Logistic regression (LR)

Since this technique also predicts the likelihood that Y is related to the input variable X, it works similarly to an NB classifier [47]. Formula (1) is a statistical model for binary logistic regression with a single predictor. *P* is the probability that the dependent variable 'X' will take the value 1, given the value of 'x', where X is the independent variable.

$$P(X) = \frac{e^{(\beta_\circ+\beta_1 X_1+\cdots+\beta_p X_p)}}{1+e^{(\beta_\circ+\beta_1 X_1+\cdots+\beta_p X_p)}} \tag{3}$$

Where:

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

It is appropriate for the parameter. Using the maximum probability technique, "$\beta_\circ$" and "$\beta_1$". The probability function is maximized in this method. The logistic function can be used to forecast the probability of the target variable p($Xi$) for a given input ($Xi$) with ease once the two parameters have been evaluated. LR is a simple model, so training takes a short amount of time. It can cope with a considerable number of features. Although it has the word regression in its name, we can only use it for classification problems due to its range, which is always between 0 and 1 [55]. It can only be used for binary classification issues and responds poorly to multi-category ones.

3.3.4. K-Nearest Neighbor (KNN)

According to the KNN method, the sample and most of the k nearest neighbors in a feature space belong to the same category. The algorithm uses several primary components, including the selection of k-value and distance measurement. First, a K value selection experiment is conducted, and a straightforward cross-validation method is used to determine the ideal K value from the text datasets. By altering the training sets' and test sets' sample selection, Second, the distance between neighbours in space is measured using distance measures. Euclidean distance is the most popular distance metric for determining the exact distance between two points in a multidimensional space. Euclidean distance is computed using the KNN algorithm to determine the distance of the data point based on formula (4) [56-58].

$$\mathrm{d(p,q)} = \sqrt{\sum_{\mathrm{i=1}}^{\mathrm{n}}(\mathrm{q_i} - \mathrm{p_i})^2} \tag{4}$$

## 4. Experiments and Results Discussion

As mentioned, four classifiers were used to evaluate the model performance (SVM, KNN, NB, and LR). The classification performance has been evaluated according to the outcomes of applying the classifiers on three different stemming techniques (Khoja, Snowball, and Tashaphyne). Since each stemming method has its procedures for stemming Arabic words, there will be a difference between stemmers in terms of rooting each word. Based on that, there should be a variety in the outcomes among the classifiers. Accuracy is adopted to evaluate the proposed model.

In general, accuracy relies on the confusion matrix, which compares the properly and erroneously categorized values to the actual results in the test data and is a widely used visual aid for illustrating how well classification algorithms perform. There are four variables included in the accuracy measurement as the following formula:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \tag{5}$$

Where:

$\boldsymbol{TP}$: are the cases that were correctly predicted by the model.
$\boldsymbol{TN}$ are the cases that were expected to be incorrect by the model.

$\boldsymbol{FP}$ are the cases predicted by the model but are incorrect.
$\boldsymbol{FN}$: These cases were predicted to be incorrect by the model but are true.

All the study tools and procedures were implemented using Python programming tools. The dataset was divided into two sets: a training set, which formed (80%) of the whole cleaned dataset, and the rest assigned to the testing phase. All the classification and stemming tools and methods were applied with the same settings.

As stated in Table 2, four different classifiers were used in addition to the three classifiers already mentioned. The procedure started by encoding the reviews included in the dataset and applying the main text cleaning and encoding tools.

Table 1. Classification Performance

| **Classifier** | **No Stemming** | **Stemming** | | |
|---|---|---|---|---|
| | | **Khoja** | **Snowball** | **Tashaphyne** |
| SVM | 0.88 | 0.90 | 0.91 | 0.90 |
| NB | 0.82 | 0.88 | 0.89 | 0.89 |
| LR | 0.89 | 0.89 | 0.91 | 0.90 |
| KNN | 0.78 | 0.85 | 0.85 | 0.81 |

From the results illustrated in Table 2, it is seen that SVM classifiers outperformed other classifiers with 88% accuracy without stemming, and the best performance was achieved with Snowball stemming, reaching 91% accuracy. It also presented a steady performance even when applying different stemming methods, with other stemming tools showing slight differences from the best one. The lowest performance goes to the KNN classifier, which started from the performance of (78%). It also presented a fluctuating performance along with the stemming methods, with Snowball and Khoja at about (85%) and going down with Tashaphyne stemmer at only (81%). For the NB classifier, the performance started at a low rate of 82% but noticeably rose to 89% with the Tashaphyne and Snowball stemmers. The performance of LR seems to be close to that of SVM, showing stability across different stemming tools. Additionally, LR continued to deliver a similar performance rate even when applying stemming tools (Fig. 4).

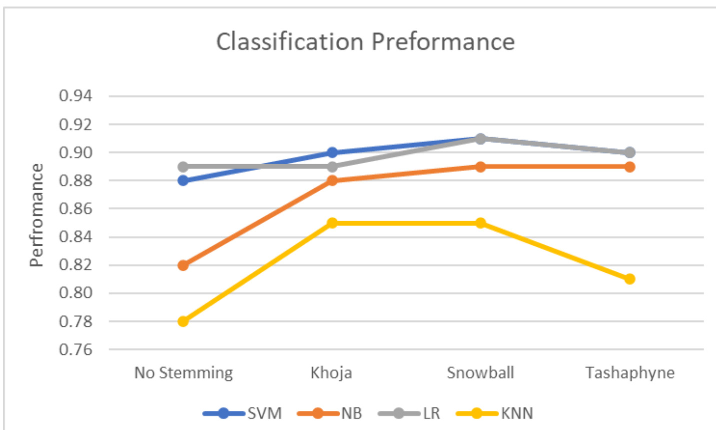


Fig. 4. Classification Performance along with different stemming tools

From the results obtained, we can see that stemming methods enhance the results for most classifiers in our model, except for the KNN classifier, which seems not to be affected by stemming tools. Since the idea of stemming relies on the attempt to reduce the number of similar words into one root word, this helped enhance the performance. Snowball presented the best stemmer among all three stemmers, which is depicted by the performance rate, which increased with most classifiers. Despite the positive effect of the stemming tools, the enhancement rate is still not satisfactory, and this is due to several factors. Firstly, the complexity of the Arabic language in terms of structure, which mainly relies on its orthographic nature, makes it highly sensitive to processing tools, leading to different terms with different meanings. Secondly, the processed dataset is a customer review with a free writing style with no rules and restrictions. This makes the reviews unpredictable, and sometimes people express their opinions using symbols or emojis, which are ambiguous to machines when processed. Lastly, the dataset is full of dialect terms, and in Arab countries, each country has its own dialect (with even many dialects in the same region). Dialectal terms significantly affect the results because some terms can mean different things depending on the dialect context.

## 5. Conclusion

From the results obtained from the presented model, it can be said that, in general, stemming tools contribute positively to enhancing the performance rate for text classification. SVM and Snowball were identified as the best methods according to our model, as they showed the highest performance rates compared to other methods. However, the positive impact is still below expectations, as the effect is not sufficiently significant. Based on the study's procedure and the nature of the dataset, companies face limitations and difficulties in classifying Arabic texts, especially when dealing with Arabic dialects where a word's meaning might differ from its exact meaning. The proposed method offered a commendable attempt at classifying Arabic reviews by comparing different tools and methods. This study paves the way for adopting more advanced methods to effectively deal with Arabic texts and explore tools and methods to overcome dialect limitations, such as WordNet and Word Embedding.

**Acknowledgment**

The authors would like to acknowledge the cooperation and assistance they received from the Technical College of Management - Baghdad, Middle Technical University, Baghdad, Iraq, to complete this research.

**References**

[1] H. Elzayady, K. M. Badran, and G. I. Salama, "Arabic Opinion Mining Using Combined CNN - LSTM Models," International Journal of Intelligent Systems and Applications, vol. 12, no. 4, pp. 25–36, Aug. 2020, doi: 10.5815/ijisa.2020.04.03.
[2] H. H. Do, P. W. C. Prasad, A. Maag, and A. Alsadoon, "Deep Learning for Aspect-Based Sentiment Analysis: A Comparative Review," Expert Systems with Applications, vol. 118. 2019. doi: 10.1016/j.eswa.2018.10.003.
[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, doi: 10.1016/j.heliyon.2021.e06191.
[6] 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. 2021. doi: 10.1016/j.jksuci.2019.02.006.
[7] R. Obiedat, D. Al-Darras, E. Alzaghoul, and O. Harfoushi, "Arabic Aspect-Based Sentiment Analysis: A Systematic Literature Review," IEEE Access, vol. 9. 2021. doi: 10.1109/ACCESS.2021.3127140.
[8] 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. 2018. doi: 10.1016/j.asej.2017.04.007.
[9] H. Elzayady, M. S. Mohamed, K. M. Badran, and G. I. Salama, "Detecting Arabic textual threats in social media using artificial intelligence: An overview," Indonesian Journal of Electrical Engineering and Computer Science, vol. 25, no. 3, 2022, doi: 10.11591/ijeecs.v25.i3.pp1712-1722.
[10] S. Larabi Marie-Sainte, N. Alalyani, S. Alotaibi, S. Ghouzali, and I. Abunadi, "Arabic Natural Language Processing and Machine Learning-Based Systems," IEEE Access, vol. 7, pp. 7011–7020, 2019, doi: 10.1109/ACCESS.2018.2890076.
[11] M. F. Ibrahim and A. Al-Taei, Title-Based Document Classification for Arabic Theses and Dissertations, vol. 318. 2022. doi: 10.1007/978-981-16-5689-7_17.
[12] Jannat Tariq, Mahmood F. Mosleh, Maha Abdulameer, Huthaifa A. Obeidat, and Omar A. Obeidat, "Hybrid Lossless Compression Techniques for English Text," Journal of Techniques, vol. 5, no. 1, pp. 52–57, Apr. 2023, doi: 10.51173/jt.v5i1.1059.
[13] P. Hajek, L. Hikkerova, and J. M. Sahut, "Fake review detection in e-Commerce platforms using aspect-based sentiment analysis," J Bus Res, vol. 167, 2023, doi: 10.1016/j.jbusres.2023.114143.
[14] H. M. Abdelaal, A. N. Elmahdy, A. A. Halawa, and H. A. Youness, "Improve the automatic classification accuracy for Arabic tweets using ensemble methods," Journal of Electrical Systems and Information Technology, vol. 5, no. 3, pp. 363–370, 2018, https://doi.org/10.1016/j.jesit.2018.03.001.
[15] 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, https://doi.org/10.1007/978-3-030-38752-5_7.
[16] 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.
[17] J. K. Alwan, A. J. Hussain, D. H. Abd, A. T. Sadiq, M. Khalaf, and P. Liatsis, "Political Arabic Articles Orientation Using Rough Set Theory with Sentiment Lexicon," IEEE Access, vol. 9, pp. 24475–24484, 2021, doi: 10.1109/ACCESS.2021.3054919.
[18] R. Elhassan and M. Ali, "The Impact of Feature Selection Methods for Classifying Arabic Texts," in 2nd International Conference on Computer Applications and Information Security, ICCAIS 2019, 2019. doi: 10.1109/CAIS.2019.8769526.
[19] G. F. Issa*, M. Abu-Arqoub*, and W. M. Hadi, "The Impact of Feature Selection Methods for Classifying Arabic Textual Data," International Journal of Recent Technology and Engineering (IJRTE), vol. 8, no. 4, pp. 1333–1338, 2019, doi: 10.35940/ijrte.d7163.118419.
[20] 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, 2021, doi: 10.23851/mjs.v32i4.994.
[21] 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, https://doi.org/10.1145/3404512.3404526.
[22] 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, https://doi.org/10.1007/978-3-319-67056-0_3.
[23] 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.
[24] R. M. K. Saeed, S. Rady, and T. F. Gharib, "An ensemble approach for spam detection in Arabic opinion texts," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 1, 2022, doi: 10.1016/j.jksuci.2019.10.002.
[25] H. Chouikhi, H. Chniter, and F. Jarray, "Arabic Sentiment Analysis Using BERT Model," in Communications in Computer and Information Science, 2021. doi: 10.1007/978-3-030-88113-9_50.
[26] A. Elnagar, R. Al-Debsi, and O. Einea, "Arabic text classification using deep learning models," Inf Process Manag, vol. 57, no. 1, 2020, doi: 10.1016/j.ipm.2019.102121.
[27] M. Abdul-Mageed, A. R. Elmadany, and E. M. B. Nagoudi, "ARBERT & MARBERT: Deep bidirectional transformers for Arabic," in ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference, 2021. doi: 10.18653/v1/2021.acl-long.551.

[28] 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.
[29] M. E. M. Abo, R. G. Raj, and A. Qazi, "A Review on Arabic Sentiment Analysis: State-of-the-Art, Taxonomy and Open Research Challenges," IEEE Access, vol. 7, pp. 162008–162024, 2019, doi: 10.1109/ACCESS.2019.2951530.
[30] O. Oueslati, E. Cambria, M. Ben HajHmida, and H. Ounelli, "A review of sentiment analysis research in Arabic language," Future Generation Computer Systems, vol. 112, pp. 408–430, 2020, https://doi.org/10.1007/978-3-319-67056-0_3.
[31] M. Alhanjouri, "Pre Processing Techniques for Arabic Documents Clustering," International Journal of Engineering and Management Research, no. 2, pp. 70–79, 2017.
[32] 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.
[33] 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.
[34] I. A. El-Khair, "Effects of Stop Words Elimination for Arabic Information Retrieval: A Comparative Study," pp. 1–15, 2017, https://doi.org/10.48550/arXiv.1702.01925.
[35] 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.
[36] 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, https://doi.org/10.1109/SNAMS.2019.8931842.
[37] K. L. Tan, C. P. Lee, K. M. Lim, and K. S. M. Anbananthen, "Sentiment Analysis With Ensemble Hybrid Deep Learning Model," IEEE Access, vol. 10, no. July, pp. 103694–103704, 2022, doi: 10.1109/ACCESS.2022.3210182.
[38] M. El-Masri, N. Altrabsheh, and H. Mansour, "Successes and challenges of Arabic sentiment analysis research: a literature review," Social Network Analysis and Mining, vol. 7, no. 1. 2017. doi: 10.1007/s13278-017-0474-x.
[39] A. M. Alayba, V. Palade, M. England, and R. Iqbal, "Improving Sentiment Analysis in Arabic Using Word Representation," in 2nd IEEE International Workshop on Arabic and Derived Script Analysis and Recognition, ASAR 2018, 2018. doi: 10.1109/ASAR.2018.8480191.
[40] M. N. Al-Kabi, S. A. Kazakzeh, B. M. Abu Ata, S. A. Al-Rababah, and I. M. Alsmadi, "A novel root based Arabic stemmer," Journal of King Saud University - Computer and Information Sciences, vol. 27, no. 2, 2015, doi: 10.1016/j.jksuci.2014.04.001.
[41] 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 6th International Conference on Social Networks Analysis, Management and Security, SNAMS 2019, 2019. doi: 10.1109/SNAMS.2019.8931842.
[42] 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," in Journal of Physics: Conference Series, IOP Publishing Ltd, Dec. 2019. doi: 10.1088/1742-6596/1402/6/066030.
[43] T. Zerrouki, "Tashaphyne, Arabic light stemmer," Tashaphyne/0.2, 2010.
[44] A. Oussous, A. A. Lahcen, and S. Belfkih, "Impact of text pre-processing and ensemble learning on Arabic sentiment analysis," in ACM International Conference Proceeding Series, 2019. doi: 10.1145/3320326.3320399.
[45] Y. A. Alhaj, J. Xiang, D. Zhao, M. A. A. Al-Qaness, M. Abd Elaziz, and A. Dahou, "A Study of the Effects of Stemming Strategies on Arabic Document Classification," IEEE Access, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2903331.
[46] M. O. Alhawarat, H. Abdeljaber, and A. Hilal, "Effect of stemming on text similarity for Arabic language at sentence level," PeerJ Comput Sci, vol. 7, p. e530, May 2021, doi: 10.7717/peerj-cs.530.
[47] 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, doi: 10.14257/ijgdc.2018.11.9.09.
[48] 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, https://doi.org/10.1002/asi.23609.
[49] C. Zong, R. Xia, and J. Zhang, "Text Classification," in Text Data Mining, Springer, 2021, pp. 93–124, https://doi.org/10.1007/978-981-16-0100-2_5.
[50] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," Multimed Tools Appl, vol. 78, pp. 3797–3816, 2019, https://doi.org/10.1007/s11042-018-6083-5.
[51] V. N. Vapnik, "An overview of statistical learning theory," IEEE Transactions on Neural Networks, vol. 10, no. 5. 1999. doi: 10.1109/72.788640.
[52] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1998. doi: 10.1007/s13928716.
[53] Ahmed Burhan Mohammed, "Decision Tree, Naïve Bayes and Support Vector Machine Applying on Social Media Usage in NYC / Comparative Analysis," Tikrit Journal of Pure Science, vol. 22, no. 9, pp. 94–99, 2023, doi: 10.25130/tjps.v22i9.881.
[54] 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.
[55] A. Yousaf et al., "Emotion recognition by textual tweets classification using voting classifier (LR-SGD)," IEEE Access, vol. 9, pp. 6286–6295, 2020, https://doi.org/10.1109/ACCESS.2020.3047831.
[56] I. Prayoga and M. Dwifebri, "Sentiment Analysis on Indonesian Movie Review Using KNN Method With the Implementation of Chi-Square Feature Selection," Jurnal Media Informatika Budidarma, vol. 7, no. 1, 2023, http://dx.doi.org/10.30865/mib.v7i1.5522.
[57] F. M. J. M. Shamrat et al., "Sentiment analysis on twitter tweets about COVID-19 vaccines using NLP and supervised KNN classification algorithm," Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 1, 2021, doi: 10.11591/ijeecs.v23.i1.pp463-470.
[58] Z. Chen, L. J. Zhou, X. Da Li, J. N. Zhang, and W. J. Huo, "The Lao text classification method based on KNN," in Procedia Computer Science, 2020. doi: 10.1016/j.procs.2020.02.053.