IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 1, 2019, accepted February 18, 2019, date of publication March 11, 2019, date of current version March 26, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2903331

A Study of the Effects of Stemming Strategies on Arabic Document Classification

YOUSIF A. ALHAJ[1], JIANWEN XIANG[1], DONGDONG ZHAO[1], MOHAMMED A. A. AL-QANESS[2], MOHAMED ABD ELAZIZ[3], AND ABDELGHANI DAHOU[1]

[1]Hubei Key Laboratory of Transportation of Internet of Things, School of Computer Science and Technology, Wuhan University of Technology, 430070, Wuhan, China
[2]School of Computer Science, Wuhan University, Wuhan 430072, China
[3]Department of Mathematics, Faculty of Science, Zagazig University, Zagazig 44519, Egypt

Corresponding author: Jianwen Xiang (jwxiang@whut.edu.cn)

This work was supported in part by the Defense Industrial Technology Development Program under Grant JCKY2018110C165, and in part by the Hubei Provincial Natural Science Foundation of China under Grant 2017CFA012.

ABSTRACT Stemming is one of the most effective techniques, which has been adopted in many applications, such as machine learning, machine translation, document classification (DC), information retrieval, and natural language processing. The stemming technique is meant to be applied during the classification of documents to reduce the high dimensionality of the feature space, which, in turn, raises the functioning of the classification system, particularly with extreme modulated language, for instance, Arabic language. This paper aims to study the impact of stemming techniques, namely Information Science Research Institute (ISRI), Tashaphyne, and ARLStem on Arabic DC. The classification algorithms, namely Naïve Bayesian (NB), support vector machine (SVM), and K-nearest neighbors (KNN), are used in this paper. In addition, the chi-square feature selection is used to select the most relevant features. Experiments are conducted on CNN Arabic corpus, which is collected from Arabic websites to assess the performance of the classification system. In order to evaluate the classifiers, the K-fold cross-validation method and Micro-F1 are used. Findings of this paper indicate that the ARLStem outperforms the ISRI and Tashaphyne stemmers. The outcomes clearly showed the effectiveness of the SVM over the KNN and NB classifiers, which achieved 94.64% Micro-F1 value when using the ARLStem stemmer.

INDEX TERMS Arabic text classification, text preprocessing, stemming techniques, feature extraction, feature selection.

I. INTRODUCTION

Recently, with the increment of the data in diverse resources, Document Classification (DC) becomes essential to extract the knowledge and significant information in the research field of Natural Language Processing (NLP) [1], Information Retrieval (IR) [2], and data mining [3]. One of the critical tasks of text mining that is dependable on recognizing, understanding, discerning, and organizing various types of textual data collections is known as DC. It also defined as the process of classifying documents into different categories based on their linguistic features [4]. Based on this concepts, DC plays a vital role in several applications such as website classification [5], spam filtering [6], automatic indexing [7], and email filtering [8].

The associate editor coordinating the review of this manuscript and approving it for publication was Hao Ji.

In general, the DC framework consists of three main steps [9] namely documents preprocessing, document modeling (indexing) and classification. Document preprocessing is the task that changes the documents into an appropriate presentation for the classification system. Several preprocessing techniques can be applied such as normalization, tokenization, stop word elimination, and stemming technique. The second step of DC is the document modeling which is known as the method that extracts features from the text and converts it into the algebraic vector. The Final step is that the classification known as the method that constructs the classification model and model evaluation.

Document preprocessing is one of many challenges in Arabic DC and has the largest effect on the performance of the DC model. Due to the richness of the Arabic language which contains complex morphology compared to other languages [4], [10], [11]. There are many techniques

which have been adopted as preprocessing to improve the performance of the DC model. The stemming technique is an important method of preprocessing that is used to make the task less dependent, reduce the feature space and enhance the performance of classification [12]. Stemming techniques are adopted in several applications for example, text summarization [13], documents classification [12], sentiment analysis [14], data compression [15].

Many researchers investigated various Arabic stemming techniques to enhance the performance of Arabic DC. This paper presents the results of an experiment by comparing three stemmers which are Information Science Research Institute stemmer (ISRI) [16], Tashaphyne [17] and Arabic light stemmer (ARLStem) [18] on Arabic DC. The study is the first of its kind that deals with the ARLStem stemmer on Arabic DC, especially with public dataset such as CNN dataset collected by Saad and Ashour [19] according to the background search. Excessive numbers of features lead to an increase in the computational process and less classification accuracy [20]. Therefore, the chi-square feature selection technique [21] employed to select the essential features. Whereas, the Naïve Bayesian (NB) [22], K-Nearest Neighbors (KNN) [23], and Support Vector Machine (SVM) [21] are applied to build the model of classification. The results of this paper can help the researchers to select the optimal stemmer, classifier, and the size of features for Arabic DC.

This paper has been organized as follows: a review of the literature which has been published in Arabic DC has been given in section 2. Introduction to supervised learning algorithms in section 3. Research methodology in section 4. Results of experiments have been presented in section 5. Finally, conclusions and future works are shown in the last section.

II. RELATED WORK

Many studies have been presented to solve the problem of DC in Latin-based languages and other languages. Nevertheless, limited researches have been conducted on the classification of Arabic documents.

Yousif *et al.* [24] used Naïve Bayes (NB) classifier on Arabic DC to evaluate the effect of a Light stemmer, Khoja stemmer, and Root extractor. They conclude that the best performance can be obtained by root extractor.

Ayedh *et al.* [9] studied the influence of preprocessing techniques such as normalization, stemming and stop words exclusion on Arabic DC. SVM, NB and KNN classifiers are applied to build the classification model. Chi-square, feature selection method, considered to select the critical features. They determined that SVM classifier outperformed others classifiers when the stemming and normalization were combined.

Duwairi *et al.* [25] assessed the influence of stemming methods on Arabic DC. The stemming methods namely-light stemming and word clusters. They affirm that the light stemming method improved the accuracy more than the other methods.

Kanan and Fox [26] presented a new Arabic light stemmer namely-P-stemmer, a modified version of Larkey's light stemmers. They validated that their stemmer increases the accuracy of DC when applying NB, SVM and random forest (RF) classifiers. Their experiments indicated that the SVM classifier performed better than other classifiers.

Elhassan and Ahmed [27] evaluated two stemming techniques namely Khoja stemmer and Light stemmer on Arabic DC. Several machine classification algorithms such as Sequential Minimal Optimization (SMO) [28], NB, J48, and KNN are employed to build the model of classification. Their results showed that the Light stemmer outperforms the Khoja stemmer.

Rouhia *et al.* [29] studied the impact of normalization and the stemming techniques such as Tashaphyne stemmer, and ISRI stemmer on Arabic DC. Bag of Words (Bow) used to extract features from the documents. Their results showed that the best outcome found by normalization and the stemmers have less accuracy. Finally, this paper deal with new stemming technique and using a public dataset with selected essential features from the dataset using a feature selection method.

Abainia *et al.* [18] built a novel Arabic Light Stemmer (ARLStem) based on stripping prefixes, suffixes, and infixes from the words. They determine that their work is the first work that deals with remove infixes and their results outperform many present light stemmers.

Bahassine *et al.* [30] presented a novel stemming algorithm that decreases the attributes to their root for Arabic document classification. Decision tree (DT) classifier is used to build the model. Chi-square is used as a features selection method. Their results were compared with Khoja stemmer. The outcomes indicated that their proposed stemmer outperform Khoja stemmer.

Nehar *et al.* [31] proposed a new model for word root extraction on Arabic text classification without relying on any dictionary. At the preprocessing tasks such as non-Arabic letters, symbols, digits, and stop words were removed. LibSVM used to build the model of classification. The outcomes indicate that the proposed method of the root extraction increases the performance of the classification algorithm.

Oraby *et al.* [32] studied the impact of stemming techniques on Arabic sentiment analysis. Their results show that the Tashaphyne stemmer resulted in 93.2% of accuracy, and 92.6% with ISRI stemmer, and 92.2% with Khoja stemmer.

Bsoul and Mohd [33] investigated the effect of ISRI stemmer on the improvement of document clustering. Their results showed good enhancements with the stemmed approach as compared with the non-stemmed approach.

Bahassine *et al.* [34] developed a feature selection technique to improve the Arabic DC and compared it with three traditional features selection metrics namely Information Gain (IG), Mutual Information (MI) and Chi-square. The stemming technique used as a preprocessing task. DT and SVM classifiers are used to build the model of classification using CNN dataset [19]. The best experimental outcomes

observed when applying the improved chi-square feature selection method with SVM classifier.

III. SUPERVISED LEARNING ALGORITHMS

Machine learning (ML) methods can be categorized into supervised and unsupervised learning [35]. The supervised learning builds the model based on learning from a labeled corpus for next prediction classification [3]. The output function could be discrete (for example, ML classifiers) or continuous (such as linear regression), there is a target for this data (labeled data), such as classification, but in unsupervised learning, there is no target for this data (unlabeled data), such as clustering [33].

The main supervised classifiers exploited in this study are discussed in the following paragraphs.

SVM is an algorithm that can be adopted for regression or classification methods. It builds N-dimensional hyperplane that splits the data into two groups in the classification method. We can use the SVM as efficiently perform a non-linear classification using the kernel trick; it is considered as a signature algorithm to solve the problem of high dimensionality in DC. The weaknesses of SVM include high memory requirements, poor interpretability, and complexity [36], [37].

K-NN is an algorithm used for classification and regression; K-NN suffers from several disadvantages such as poor accuracy if K has been not appropriately selected, sensitivity to irrelevant parameters and the need for a proper similarity measure such as the Cosine measure [21].

NB classifier is a technique based on the Bayesian theorem, according to equation 1, and is particularly suited when the dimensionality of the inputs is significant, regardless of its simplicity. NB is a classifier that depends on the probabilistic condition:

$$p(c|x) = \frac{p(x|c)p(c)}{p(x)} \quad (1)$$

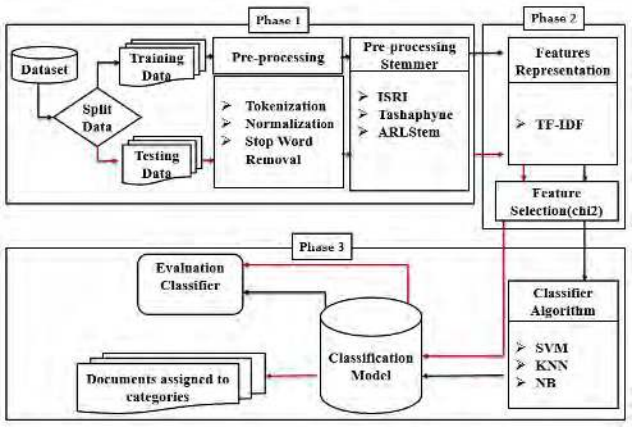

FIGURE 1. Arabic DC framework.

IV. METHODOLOGY

The three main phases in Arabic DC are the preprocessing phase, feature representation (extraction) phase, and document classification phase as illustrated in Fig.1. Preprocessing phase comprises the tasks that convert the text into an appropriate format. Feature representation (extraction) includes the process that extracts the feature from the document and changes it into a numerical vector. Feature selection method selects the necessary features. Classification model and its evaluation represent the last DC phase.

A. DOCUMENT PREPROCESSING

Document preprocessing is the method that converts the documents into the suitable form of the classification. Preprocessing technique can be used to reduce the computational process and feature space which enhances the performance and classification accuracy. In this phase all non-Arabic words, stop words, digits, numbers, and punctuation were removed [38], the preprocessing includes the following steps.

1) TOKENIZATION

Tokenization is the method that separated the text into tokens. These tokens might be separate words, sentences or paragraphs. Words are often separated from each other by breaks such as white space, commas, periods, semicolons, and quotes. The output of tokens becomes the input for the next preprocessing step.

2) NORMALIZATION

Normalization is the process that transforms the letter in text into a canonical form, for instance transform Hamza "ء" in (آ, أ, إ) into ا, Taa Marboutah "ة" into "ه", Yaa Mamdoda "ى" into "ي", remove the Diacritics and elongation such as in "جَمِيْلـــــــــــةٌ" into "جميله". Therefore in this paper, we used the normalization rules that indicated by [39].

3) STOP WORDS ELIMINATION

Stop words known as the frequent words that bring no meaning or indications about the content (i.e., prepositions, pronouns, conjunctions) such as "for" لاجل, "so, لذا" or "with, مع", these words do not help in distinguishing between different classes [4]. Stop words removal include the elimination of insignificant words. The list of stop words was prepared and removed from all documents.[1]

4) STEMMING TECHNIQUES

Stemming techniques are considered as an essential preprocessing stage before tackling any task of DC or information retrieval in NLP. After removing the unimportant information, the different words that came from the same word were mapped to their root or stem [26]. Word extracting method reducing the number of words that extract from the documents and decrease the morphological variance of words [39]. The high dimensionality of features is a significant challenge in DC. The stemming technique is one method of preprocessing techniques that reduce the high dimensionality of vector space by reducing the word to its root or stem.

[1] https://github.com/yalhag1/Arabic-stop-word-list

With regards to the Arabic language, the most appropriate stemming techniques are the root-based technique [40] and the stem based technique [41].

- Root-based stemmer applies morphological procedures to retrieve the root. Many root-based stemming methods have been built for Arabic DC such as in [42] which initially peels off layers of suffixes and prefixes after that checks a list of formats and roots to decide if the remainder may be a known root with a known format adopted. If so, then it returns the root. Else, it returns the actual word, not modified. The Root-based stemmer has limitations, for example, the root dictionary demands an update to make sure that new detected terms are stemmed appropriately and if the root comprises feeble character (i.e., alif, waw or yah), the form of this character could be changed during derivation. In order to address this, the stemmer should be checked to ascertain if the weak character is in the right pattern. If it is not, the stemmer yields the right patterns of this weak character, which then offers the right pattern of the root. Again, it takes over a weak message with (ا و ي) which yields a wrong root. For instance, the word (''Munathammat'' منظمات) is stemmed to (''Thama'' ظما) instead of (''Nath'ama'' نظم). For example, ISRI stemmer [16] is a rule-based stemmer that stems the word according to specific rules to find its root and is similar to Khoja stemmer [45] but without using root dictionary [16]. Furthermore, the word that cannot be rooted, the ISRI bring a normalized form after it normalizes the word (for illustration, removal of the end patterns and specific determinants) instead of allowing for the word unchanged [32]. The improvements of the ISRI stemmer has been showed in document clustering [33].
- The light stemming technique is the process of removing the very often prefixes and suffixes based on a predefined list of suffixes and prefixes. The light stemming technique is not meant to retrieve the root of a selected Arabic word; hence, this technique is not meant for dealing with infixes or recognize patterns. Many light stemmers have been recommended for the Arabic language [18]. With regards to the light stemmer, its technique gets rid of affixes, predefined in the list, without finding it out if the leftover is a stem. In some cases, truncates it from the letter and yields a wrong stem (e.g. ''Bustan'' بستان offers ''Busta'' بستا). There is no standard algorithm for Arabic light stemming; all attempts in this aspect were a set of guidelines to remove a small amount of suffix and prefixes. Again, there is an indefinite list of these removable affixes [43]. For instance, Tashaphyne [44], and ARLStem [18] are two stemmers that employ light stemming technique. To be more precise, Tashaphyne stemmer works by first normalizing words in preparation for the ''search and index'' tasks that require stemming. Secondly, segmentation and stemming of the input are performed using a default Arabic affix lookup list which allows for various levels of stemming and rooting [32]. Whereas, ARLStem is a new light stemmer based on updated some new rules for stripping prefixes, suffixes, and infixes smartly to obtain the stem [18]. The ARLStem stemmer begins by characters normalizing, suffixes and prefixes elimination. After that dealing with plural, feminine, and verb to extract the stem of the word.

B. DOCUMENT REPRESENTATIO

Document modeling includes feature extraction (representation) and feature selection. Term Frequency-Inverse Document Frequency (TF-IDF) is used to extract features from the document and represent it as a numerical vector. Feature selection technique is the method that removes unnecessary features and selects the essential features. We investigated chi-square feature selection approach , which is defined as the method of testing the hypothesis of discrete data known from statistics; this method assesses the relationship among two variables and finds out whether these variables are interrelated or correlated [46]. χ^2 value for each a category μ and term t can be defined by using equation (2).

$$\chi^2(t_k, \mu_i) = \frac{|\mathrm{Tr}|.[\mathrm{p}(\mathrm{t}_k, \mu_i) * \mathrm{p}(\mathrm{t}_k^-, \mu_i^-) - \mathrm{p}(\mathrm{t}_k, \mu_i^-) * \mathrm{p}(\mathrm{t}_k^-, \mu_i)]^2}{\mathrm{p}(\mathrm{t}_k) * \mathrm{p}(\mathrm{t}_k^-) * \mathrm{p}(\mu_i) * \mathrm{p}(\mu_i^-)} \tag{2}$$

Moreover, χ^2 it is estimated using the following equation:

$$\chi^2(\mathrm{t}, \mu) = \frac{\mathrm{N} * (\alpha\omega - \mathrm{U}\beta)^2}{(\alpha + \mathrm{U}) * (\beta + \omega) * (\alpha + \beta) * (\mathrm{U} + \omega)} \tag{3}$$

where α is the frequency of t and μ occurrences, β is the frequency of t occurrences without μ, U is the frequency of μ without t, ω is the frequency of non-occurrence of both μ and t, and N is the quantity of document.

C. DOCUMENT CLASSIFICATION

The final phase of the DC framework of classification is known as the process of classifying documents based on their content, where the training and testing data are separated from a dataset. In this step, the classification algorithms build the model from training and evaluated by testing the data. We assess the effect of each stemmer using the most popular automatic learning and statistical classification techniques such as NB [22], KNN [23], and SVM [21].

V. EXPERIMENT RESULTS

We have implemented our application using python 3.6.0 programming. Also, we used tools for machine learning and data analysis known as scikit-learn[2] to study the effect of stemming techniques on Arabic DC alongside with NB, KNN, and SVM classifiers.

[2]https://scikit-learn.org/stable/index.html

TABLE 1. The statistic of CNN dataset.

Category Name	Number of Documents
Business	836
Entertainments	474
Middle East News	1462
Science & Technology	526
Sports	762
World News	1010
Total	**5070**

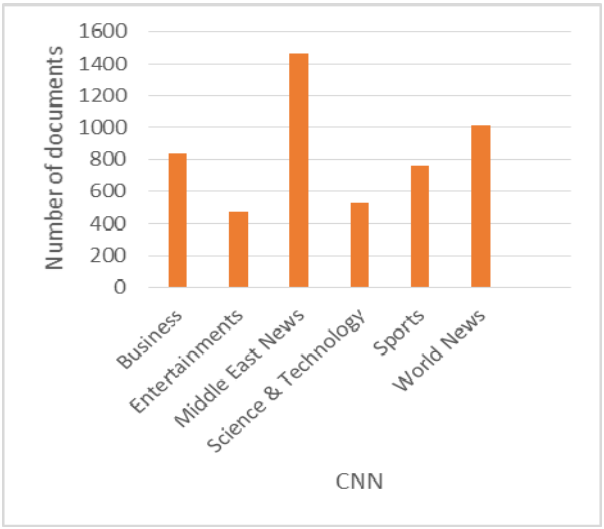

FIGURE 2. Distribution of documents in each category.

A. DATASET DESCRIPTION

In most Arabic DC studies, authors have collected their dataset individually. In this paper, we evaluated the effect of well-known stemmers on publicly available Arabic dataset which is CNN Arabic corpus obtained by Saad and Ashour [19]. The statistics and the data class distribution are described in Table 1, and Fig.2.

B. PERFORMANCE ANALYSIS

In this paper, the documents in each class in the dataset were firstly prepared by converting them to UTF-8 encoding. All non-Arabic characters, numbers, and symbols were removed from the dataset. After that, the preprocessing tasks such as normalization, stop word removal were applied. For the normalization, we used four rules as used indicated by [39]. For stop words elimination, we used a file that contains 1057 stop words to remove them. Then, the stemming technique is applied to extract the root or stem from each word. Concerning the feature extraction process, TF-IDF is used to represent documents as a numerical vector. Chi-square is used to select the essential feature sets that can contain 1000, 2000, 3000, 4000, or 5000 features. The effect of stemmer was observed within a wide range of feature size. NB, KNN, and SVM classifiers were used to observe the classification performance of the Arabic DC system. The parameters setting of ML algorithms which were adopted in our experiments are configured as for NB classifier the alpha $= 0.1$, for KNN Classifier the number of k neighbors $= 6$, for SVM classifier the kernel $=$ linear and the C $= 1$.

Ten-fold Cross-validation is used in all classification experiments, which split the data into ten folds that are mutually exclusive subsets, where each fold includes 507 documents. The testing set is one of the subsets, where the reminder subsets are used as training sets.

This study used Micro-F1 score which is known as F1 measure that calculated as in equation (4):

$$\text{Micro} - \text{F1} = \frac{2 * \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

where the recall and precision are defined in the following equations:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

where TP represents all the documents which are indicated correctly to the specified class. TN represents all the documents which are correctly indicated not to belong to the class. FP represents all documents which are incorrectly indistinct to the class. FN represents all documents which are incorrectly not defined to the class.

C. RESULTS AND DISCUSSION

Firstly, we run three different sets of experiments to study the effect of ISRI, Tashaphyne, and ARLStem stemmers on Arabic DC. Three different tests were performed using four different sets representations of the same dataset. The original dataset using normalization, stop word removal in this experiment represented as the baseline. In the second we used experiment ISRI stemmer. Tashaphyne stemmer used in the third experiment. In the last experiment, we used ARLStem stemmer.

The results of the experiments for the baseline and stemmers on three classification algorithms are illustrated in Table 2 and Fig.3. The positively impacted results have marked as bold. The outcomes of classification based on ten-fold Cross-validation method and Micro-F1 used to test classifiers.

According to the proposed method ISRI, Tashaphyne, and ARLStem stemmers have different impacts on the classification performance of Arabic DC as compared with the baseline dataset as shown in Table 2 and Fig.3.

Moreover, the conclusion can be made based on the following observations: First, applying ISRI stemmer rendered significant improvement in classification performance with NB and SVM classifiers as the feature size decreased. However, it has a negative impact when feature size increased. ISRI stemmer has a significant improvement with KNN in all features without relying on feature size.

TABLE 2. Micro-F1 scores for evaluating stemmers on CNN dataset.

CA[1]	Fs[2]	Baseline	ISRI	Tashaphyne	ARLStem
NB	1000	0.8823	**0.9101**	**0.9099**	**0.9140**
	2000	0.9130	**0.9264**	**0.9254**	**0.9274**
	3000	0.9243	**0.9272**	**0.9308**	**0.9320**
	4000	0.9322	0.9286	**0.9331**	**0.9359**
	5000	0.9347	0.9306	0.9331	**0.9371**
KNN	1000	0.8422	**0.8949**	**0.8818**	**0.8834**
	2000	0.8438	**0.8878**	**0.8813**	**0.8842**
	3000	0.8059	**0.8797**	**0.8777**	**0.8682**
	4000	0.8037	**0.8602**	**0.8785**	**0.8603**
	5000	0.8306	**0.8900**	**0.8972**	**0.8749**
SVM	1000	0.9099	**0.9363**	**0.9353**	**0.9337**
	2000	0.9345	**0.9402**	**0.9424**	**0.9428**
	3000	0.9389	**0.9408**	**0.9454**	**0.9448**
	4000	0.9406	0.9398	**0.9460**	**0.9460**
	5000	0.9434	0.9395	**0.9450**	**0.9464**

[1] Classifier algorithm, [2] Features size

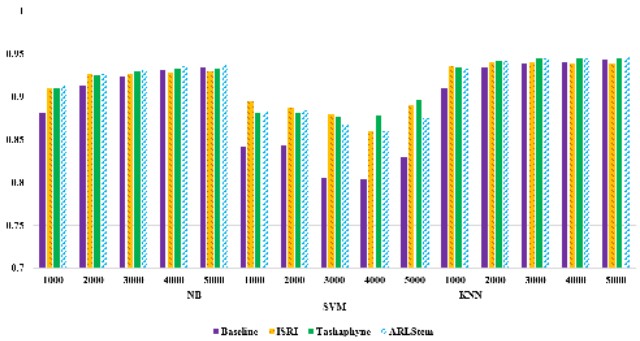

FIGURE 3. Micro-F1 Scores for evaluating stemmers on CNN dataset.

Secondly, Tashaphyne stemmer rendered a significant improvement in classification performance with KNN and SVM classifiers without depending on feature size. However, it has a negative impact on NB classifier when increasing the size of features.

Finally, applying the ARLStem stemmer has a dominant impact and provides a significant improvement in the classification performance for NB, KNN and SVM classifiers without relying on feature size.

The outcomes show that the stemmer techniques assisted in improving the performances and provided a significant improvement in the classification performance when applying stemming techniques. It has been observed from the experiment conducted that ARLStem stemmer outperformed other stemmers when it recorded the highest results when using SVM classifier. Therefore, SVM classifier is outstanding as compared to other classifiers because this technique can handle high dimensional data.

Findings of this study show that the stemmers techniques help to enhance the performance of DC and provide a significant improvement in Arabic DC by considering the feature size, and classifier algorithms. The outcomes could help the researcher to determine the appropriate method of stemming technique, classifier algorithm, and determine the feature size without applying these methods directly.

Furthermore, it is worth to mention that, during experiment analysis, we found that normalization rules combined with specific classifiers and features size can improve the classification performance such as the usage of Diacritics and Yaa Mamdoda "ى" with SVM classifier and 5000 as feature size. On the contrast, using normalization rules such as "أ" and "ة", SVM classifier, and feature size of 5000 features may have a negative impact as shown in Table 3. Some normalization rules may change the format of the word which can affect the classification accuracy. Researchers may have to pay attention when dealing with the normalization process depending on different dataset, classifier algorithms and feature selection method.

TABLE 3. Micro-F1 scores for effect normalization rules on CNN dataset.

CA	Fs	Baseline	SP[1]+ Nor[2](Diacritics+"ى")	SP+ Nor("أ" + "ة")
NB	1000	0.8823	0.8813	0.8825
	2000	0.913	0.9130	0.9124
	3000	0.9243	0.9237	0.9241
	4000	0.9322	0.9326	0.9310
	5000	0.9347	0.9353	0.9345
KNN	1000	0.8422	0.8463	0.8428
	2000	0.8438	0.8310	0.8408
	3000	0.8059	0.8104	0.8059
	4000	0.8037	0.8286	0.8010
	5000	0.8306	0.8432	0.8280
SVM	1000	0.9099	0.9099	0.9104
	2000	0.9345	0.9333	0.9345
	3000	0.9389	0.9398	0.9392
	4000	0.9406	0.9412	0.9412
	5000	0.9434	0.9438	0.9426

[1] Stop word removal, [2] Normalization

VI. CONCLUSIONS

In this paper, we investigated the effect of Information Science Research Institute (ISRI), Tashaphyne, and ARLStem stemmers on the performance of Arabic Document Classification. Term Frequency-Inverse Document Frequency (TF-IDF) used to extract the feature from text on documents and represented it as numerical in the vector space model. Chi-square feature selection applied to select the high ranked features from the dataset that contain several thousands of features. Three machine learning algorithms have been examined to solve the problem of Document Classification.

Findings of this study show that the stemming techniques reduce the computational process of classification algorithm

by reducing the feature space and provide a slight improvement in the performance of Arabic Document Classification. Besides, the experimental results show that ARLStem stemmer offers a good performance regarding other classification results, and dimension reduction rate in comparison with the other two Arabic stemmers. Moreover, the results show that the SVM classifier outperforms different classifiers. The SVM classifier reached 94.64% Micro-F1 by applying the ARLStem stemmer. These results will be helpful for examiners in Arabic Document Classification system in selecting the type of stemmer, classification algorithm and relevant features from the dataset. Future research could focus on developing an Arabic stemmer which aims to overcome the disadvantages of state-of-the-art stemming techniques.

REFERENCES

[1] S. Sun, C. Luo, and J. Chen, "A review of natural language processing techniques for opinion mining systems," *Inf. Fusion*, vol. 36, pp. 10–25, Jul. 2017.
[2] B. Xu, H. Lin, Y. Lin, L. Yang, and K. Xu, "Improving pseudo-relevance feedback with neural network-based word representations," *IEEE Access*, vol. 6, pp. 62152–62165, 2018.
[3] A. M. Mesleh and G. Kanaan, "Support vector machine text classification system: Using ant colony optimization based feature subset selection," in *Proc. Int. Conf. Comput. Eng. Syst.*, Nov. 2008, pp. 143–148.
[4] J. Ababneh, O. Almomani, W. Hadi, N. K. T. El-Omari, and A. Al-Ibrahim, "Vector space models to classify Arabic text," *Int. J. Comput. Trends Technol.*, vol. 7, no. 4, pp. 219–223, 2014.
[5] A. Ahmadi, M. Fotouhi, and M. Khaleghi, "Intelligent classification of Web pages using contextual and visual features," *Appl. Soft Comput. J.*, vol. 11, no. 2, pp. 1638–1647, 2011.
[6] M. Al-Kabi, H. Wahsheh, I. Alsmadi, E. Al-Shawakfa, A. Wahbeh, and A. Al-Hmoud, "Content-based analysis to detect Arabic Web spam," *J. Inf. Sci.*, vol. 38, no. 3, pp. 284–296, 2012.
[7] G. Percannella, D. Sorrentino, and M. Vento, "Automatic indexing of news videos through text classification techniques," in *Proc. Int. Conf. Pattern Recognit. Image Anal.* Berlin, Germany: Springer, Aug. 2005, pp. 512–521.
[8] G. Mujtaba, L. Shuib, R. G. Raj, N. Majeed, and M. A. Al-Garadi, "Email classification research trends: Review and open issues," *IEEE Access*, vol. 5, pp. 9044–9064, 2017.
[9] A. Ayedh, G. Tan, K. Alwesabi, and H. Rajeh, "The effect of preprocessing on Arabic document categorization," *Algorithms*, vol. 9, no. 2, p. 27, 2016.
[10] S. H. Mustafa, "Word stemming for Arabic information retrieval: The case for simple light stemming," *Abhath Al-Yarmouk, Sci. Eng. Ser.*, vol. 21, no. 1, p. 2012, 2012.
[11] H. Froud, A. Lachkar, and S. Ouatik, "A comparative study of root-based and stem-based approaches for measuring the similarity between Arabic words for Arabic text mining applications," *Adv. Comput., Int. J.*, vol. 3, no. 6, pp. 55–67, 2012.
[12] A. Ayedh and G. Tan, "Building and benchmarking novel Arabic stemmer for document classification," *J. Comput. Theor. Nanosci.*, vol. 13, no. 3, pp. 1527–1535, 2016.
[13] C. Slamet, A. R. Atmadja, D. S. Maylawati, R. S. Lestari, W. Darmalaksana, and M. A. Ramdhani, "Automated text summarization for indonesian article using vector space model," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 288, no. 1, 2018, Art. no. 012037.
[14] A. Dahou, S. Xiong, J. Zhou, M. H. Haddoud, and P. Duan, "Word embeddings and convolutional neural network for Arabic sentiment classification," in *Proc. 26th Coling Int. Conf. Comput. Linguistics, Tech. Papers*, 2016, pp. 2418–2427.
[15] A. Sinaga, Adiwijaya, and H. Nugroho, "Development of word-based text compression algorithm for Indonesian language document," in *Proc. 3rd Int. Conf. Inf. Commun. Technol. (ICoICT)*, May 2015, pp. 450–454.
[16] K. Taghva, R. Elkhoury, and J. Coombs, "Arabic stemming without a root dictionary," in *Proc. Int. Conf. Inf. Technol., Coding Comput. (ITCC)*, vol. 1, 2005, pp. 152–157.
[17] *Tashaphyne Stemmer*. Accessed: Apr. 28, 2018. [Online]. Available: https://pypi.python.org/pypi/Tashaphyne/
[18] K. Abainia, S. Ouamour, and H. Sayoud, "A novel robust Arabic light stemmer," *J. Exp. Theor. Artif. Intell.*, vol. 29, no. 3, pp. 557–573, 2017.
[19] M. Saad and W. Ashour, "OSAC: Open source Arabic corpora," in *Proc. 6th Int. Conf. Elect. Comput. Syst. (EECS)*, Lefke, Cyprus, Nov. 2010, pp. 118–123.
[20] A. K. Uysal, "On two-stage feature selection methods for text classification," *IEEE Access*, vol. 6, pp. 43233–43251, 2018.
[21] M. A. Mesleh, "Chi square feature extraction based SVMs Arabic language text categorization system," *J. Comput. Sci.*, vol. 3, no. 6, pp. 430–435, 2007.
[22] Z. Jianqiang and G. Xiaolin, "Comparison research on text preprocessing methods on twitter sentiment analysis," *IEEE Access*, vol. 5, pp. 2870–2879, 2017.
[23] M. M. Syiam, Z. T. Fayed, and M. B. Habib, "An intelligent system for Arabic text categorization," *Int. J. Intell. Comput. Inf. Sci.*, vol. 6, no. 1, pp. 1–19, 2006.
[24] S. A. Yousif, V. W. Samawi, and I. Elkabani, "Enhancement of Arabic text classification using semantic relations with part of speech tagger," *Adv. Elect. Comput. Eng.*, 2015, pp. 195–201.
[25] R. Duwairi, M. N. Al-Refai, and N. Khasawneh, "Feature reduction techniques for Arabic text categorization," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 60, no. 11, pp. 2347–2352, 2009.
[26] T. Kanan and E. A. Fox, "Automated Arabic text classification with P-Stemmer, machine learning, and a tailored news article taxonomy," *J. Assoc. Inf. Sci. Technol.*, vol. 67, no. 11, pp. 2667–2683, 2016.
[27] R. Mamoun and M. Ahmed, "Arabic text stemming: Comparative analysis," in *Proc. Conf. Basic Sci. Eng. Stud. (SGCAC)*, Feb. 2016, pp. 88–93.
[28] B. Al-Shargabi, "A comparative study for Arabic text classification algorithms based on stop words elimination," in *Proc. Int. Conf. Intell. Semantic Web-Services Appl.*, 2011, p. 11.
[29] R. M. Sallam, H. M. Mousa, and M. Hussein, "Improving Arabic text categorization using normalization and stemming techniques," *Int. J. Comput. Appl.*, vol. 135, no. 2, pp. 38–43, 2016.
[30] S. Bahassine, A. Madani, and M. Kissi, "Arabic text classification using new stemmer for feature selection and decision trees," *J. Eng. Sci. Technol.*, vol. 12, no. 6, pp. 1475–1487, 2014.
[31] A. Nehar, D. Ziadi, and H. Cherroun, "Rational kernels for Arabic root extraction and text classification," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 28, no. 2, pp. 157–169, 2016.
[32] S. Oraby, Y. El-Sonbaty, and M. A. El-Nasr, "Exploring the effects of word roots for Arabic sentiment analysis," in *Proc. 6th Int. Joint Conf. Natural Lang. Process.*, Oct. 2013, pp. 471–479.
[33] Q. W. Bsoul and M. Mohd, "Effect of ISRI stemming on similarity measure for Arabic document clustering," in *Information Retrieval Technology* (Lecture Notes in Computer Science), vol. 7097. Berlin, Germany: Springer, 2011, pp. 584–593.
[34] S. Bahassine, A. Madani, M. Al-Sarem, and M. Kissi, "Feature selection using an improved Chi-square for Arabic text classification," *J. King Saud Univ.-Comput. Inf. Sci.*, to be published. doi: 10.1016/j.jksuci.2018.05.010.
[35] A. Hotho, A. Nürnberger, and G. Paaß, "A brief survey of text mining," *J. Comput. Linguistics Lang. Technol.*, vol. 20, no. 1, pp. 19–62, 2005.
[36] I. Hmeidi, M. Al-Ayyoub, N. A. Abdulla, A. A. Almodawar, R. Abooraig, and N. A. Mahyoub, "Automatic Arabic text categorization: A comprehensive comparative study," *J. Inf. Sci.*, vol. 41, no. 1, pp. 114–124, 2015.
[37] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*. Amsterdam, The Netherlands: Elsevier, 2011.
[38] S. Al-Harbi, A. Almuhareb, A. Al-Thubaity, M. S. Khorsheed, and A. Al-Rajeh, "Automatic Arabic text classification," Tech. Rep., Aug. 2008, pp. 77–84.
[39] L. S. Larkey, L. Ballesteros, and M. E. Connell, "Improving stemming for Arabic information retrieval," in *Proc. 25th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2002, pp. 275–282.
[40] G. Salton and C. Buckley, "Improving retrieval performance by relevance feedback," *J. Amer. Soc. Inf. Sci.*, vol. 41, no. 4, pp. 288–297, 1999.
[41] C. D. Manning, P. Raghavan, and H. Schütze, "Introduction to information retrieval," *Natural Lang. Eng.*, vol. 16, no. 1, pp. 100–103, 2010.
[42] S. Khoja and R. Garside, "Stemming Arabic text," Dept. Comput., Lancaster Univ., Lancaster, U.K., Tech. Rep. 1999 Sep 22, 1999.
[43] M. S. Khorsheed and A. O. Al-Thubaity, "Comparative evaluation of text classification techniques using a large diverse Arabic dataset," *Lang. Resour. Eval.*, vol. 47, no. 2, pp. 513–538, 2013.
[44] A. M. Ezzeldin, Y. El-sonbaty, and M. H. Kholief, "Exploring the effects of root expansion, sentence splitting and ontology on Arabic answer selection," *Natural Lang. Process. Cogn. Sci., Proc.*, Oct. 2014, p. 273.

[45] M. N. Al-Kabi, "Towards improving Khoja rule-based Arabic stemmer," in *Proc. IEEE Jordan Conf. Appl. Elect. Eng. Comput. Technol. (AEECT)*, 2013, pp. 1–6.
[46] F. Thabtah, M. A. H. Eljinini, M. Zamzeer, and M. Hadi, "Naïve Bayesian based on chi square to categorize Arabic data," in *Proc. 11th Int. Bus. Inf. Manage. Assoc. Conf. Innov. Knowl. Manage. Twin Track Econ. (IBIMA)*, Cairo, Egypt, Jan. 2009, pp. 4–6.

YOUSIF A. ALHAJ received the B.Sc. degree in engineering and information technology from the University of Modern Sciences, Sana'a, Yemen, in 2010, and the M.Sc. degree in computer science and technology from the Wuhan University of Technology, Wuhan, China, in 2016, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Technology. His main areas of research interest are machine learning, text mining, and Arabic natural language processing.

JIANWEN XIANG received the Ph.D. degrees from Wuhan University, Wuhan, China, and the Japan Advanced Institute of Science and Technology, Nomi, Japan, in 2004 and 2005, respectively. He was a Researcher with NEC Corporation. He is a currently a Professor with the School of Computer Science and Technology, Wuhan University of Technology. His research interests include dependable computing, formal methods, and software engineering.

DONGDONG ZHAO received the Ph.D. degree from the University of Science and Technology of China, in 2016. He is currently a Lecturer with the School of Computer Science and Technology, Wuhan University of Technology. His research interests include dependable computing, information security, privacy protection, and biometrics.

MOHAMMED A. A. AL-QANESS received the B.S., M.S., and Ph.D. degrees in information and communication engineering from the Wuhan University of Technology, in 2010, 2014, and 2017, respectively. He is currently an Assistant Professor with the School of Computer Science, Wuhan University, Wuhan, China. His current research interests include wireless sensing, mobile computing, machine learning, and Signals & image processing.

MOHAMED ABD ELAZIZ received the B.S. and M.S. degrees in computer science and the Ph.D. degree in mathematics and computer science from Zagazig University, Egypt, in 2008, 2011, and 2014, respectively. From 2008 to 2011, he was an Assistant Lecturer with the Department of Computer Science. Since 2014, he has been a Lecturer with the Mathematical Department, Zagazig University. He has authored more than 30 articles. His research interests include machine learning, signal processing, and image processing.

ABDELGHANI DAHOU was born in Algiers, Algeria, in 1990. He received the B.S. and M.S. degrees in computer science and intelligent systems from the University of Ahmad Draia, Adrar, Algeria, in 2012 and 2014, respectively. He is currently pursuing the Ph.D. degree in computer science with the Wuhan University of Technology, Wuhan, Hubei, China. His research interests include deep learning, artificial intelligence, data mining, and Arabic natural language processing.