LIVERPOOL JOHN MOORES UNIVERSITY

LJMU Research Online

Abd, DH, Khan, W, Khan, B, Alharbe, N, Al-Jumeily, D and Hussain, A

 Categorization of Arabic posts using Artificial Neural Network and hash features

http://researchonline.ljmu.ac.uk/id/eprint/21917/

Article

For more information please contact researchonline@ljmu.ac.uk

http://researchonline.ljmu.ac.uk/

Original article

Categorization of Arabic posts using Artificial Neural Network and hash features

Dhafar Hamed Abd [a,*], Wasiq Khan [b], Bilal Khan [c], Nawaf Alharbe [d], Dhiya Al-Jumeily [b], Abir Hussain [b,e,*]

[a] College of Computer Science and Information Technology, University of Anbar, Ramadi, Iraq
[b] Faculty of Engineering and Technology, Liverpool John Moores University, Liverpool L33AF, UK
[c] California State University San Bernardino, San Bernardino, CA 92407, United States
[d] Applied College, Taibah University, Saudi Arabia
[e] Department of Electrical Engineering, University of Sharjah, Sharjah, United Arab Emirates

ABSTRACT

Sentiment analysis is an important study topic with diverse application domains including social network monitoring and automatic analysis of the body of natural language communication. Existing research on sentiment analysis has already utilised substantial domain knowledge available online comprising users' opinion in various areas such as business, education, and social media. There is however limited literature available on Arabic language sentiment analysis. Furthermore, datasets used in majority of these studies have poor classification. In the present study, we utilised a primary dataset comprising 2122 sentences and 15,331 words compiled from 206 publicly available online posts to perform sentiment classification by using advanced machine learning technique based on Artificial Neural Networks. Unlike lexicon-based techniques that suffer from low accuracy due to their computational nature and parameter configuration, Artificial Neural Networks were used to classify people opinion posts into three categories including conservative, reform and revolution, accompanied by multiple hasher vector size to benchmark the performance of the proposed model. Extensive simulation results indicated an accuracy of 93.33%, 100%, and 100% for the classification of conservation, reform, and revolutionary classes, respectively.

1. Introduction

The use of online resources such as social media, personal blogs, online reviews has gained momentum in recent years (Abd et al., 2020) that allows users to share or present ideas, point of views, and comments with respect to various issues (Abd et al., 2020). Collection and automated analysis of these comments in real settings has recently been an important undertaking. For instance, customers in diverse range of retail business environment could be willing to know other users' opinions before purchasing a service or product (Medhat et al., 2014). Similarly, industries are util-

ising the body of collected knowledge based on existing customers' opinions to improve and adapt their products and services accordingly (Amarouche et al., 2015; Abd et al., 2019).

Generally, the pervasiveness of ideological websites, online social networks, and newspapers has increased significantly in various domains with the analyses of such resources to abstract reasonable information. The method of identifying the orientations from unstructured datasets is recognized by sentiment analysis (SA), sometimes known as sentiment orientation or opinion mining (Abd et al., 2020). It is the classification process that determines if the opinion or sentiment expressed in a statement or article is neutral, negative, or positive. Due to the majority of the datasets being limited to only English, with substantially limited amount of translations to other languages, it is critical to diversify sentiment analysis by including other languages for a wider availability, specifically, with larger audience such as Arabic language. It is noted that Arabic dialectal poses significant challenges in sentiment analysis due to its dissimilarity with English language.

Arabic language is regarded as the world's fifth most widely spoken language with its rich vocabulary exceeding 12 million words (Farghaly and Shaalan, 2009). Existing analysis on Arabic

* Corresponding authors.
E-mail addresses: Dhafar.hamed@uoanbar.edu.iq (D.H. Abd), abir.hussain@sharjah.ac.ae (A. Hussain).

context is minimal when compared to other studies that are carried out on attitudes, opinions, emotions, and other relevant sentiments in English language. Previous studies focused on specific aspect, such as business topics, sports articles, which were classified into the corresponding class. In this study, we used a primary dataset that intends to offer information on Arabic articles compiled and processed from social media, websites, and newspapers, to apply a new orientation recognition of polarity[1].

Arabic language required a well-established workflow due to the specific characteristics and language dissimilarities from other languages (Duwairi and El-Orfali, 2014). Highly ambiguous and morphologically complex Arabic language presents numerous challenges for sentiment analysis procedures and Natural Language Processing (NLP) approaches (Badaro, 2019). Arabic can be written in a variety of ways, including dialectal and classical representing the language of Islam's sacred book, and modern standard, which is the official language (Biltawi et al., 2016). Despite having a reasonably widespread usage across the globe, Arabic language has a few traits that poses significant challenge for sentiment analysis of other well-studied languages, such as English (Biltawi et al., 2016).

Generally, three separate classification levels are used in sentiment analysis (Liu, 2012) that are conducted at the levels of documents, sentences, and aspect levels. Our work focuses on the document level, with the goal to classify the post opinions. By talking about one article, the entire article is regarded as a basic information item. The goal is to examine articles in order to establish their opinion leanings. The selection was based on reformist, conservative, and revolutionary tendencies which are among those explored in this study.

Based on the document level sentiment analysis, there are further two techniques that include lexicon-based and machine learning (Mitra, 2020). There are several challenges associated with the lexicon-based approaches (2018). For example, lexicon-based methods contain limited words and phrases that express negative and positive emotions. Furthermore, because of word's opposite orientation of different domains using lexicon-based, sentiment categorization is inadequate. Machine learning on the other hand, outperformed lexicon-based strategies according to the parameter's versatility, ease of management, and higher accuracy (Velichkov, 2014; Nakov et al., 2013). These methods analyse the text as collection of words before transformation to numerous characteristics. This is followed by the classification algorithms such as Artificial Neural Networks (ANNs).

The novelty of the proposed study comprises three distinct aspects as follows:

- A primary dataset (i.e., corpus) comprising 2122 sentences and 15,331 words acquired from 206 publicly available posts in Arabic-language along with manual annotations is publicly available. In this case, detailed statistical aspects such as source information, overall number of sentences/articles, and punctuation are discussed. Dataset in the present study is divided into three categories including reform, revolution, and conservative. Since the focus of this study is Arabic language, to the authors' knowledge no other dataset provides such important categorization to Arabic posts
- Various hasher vector sizes are utilised for the proposed ANNs for optimisation.
- The application of ANN was used for classifying Arabic sentiment analysis with high accuracy.

The remaining study is organised as follows. Section 2 presents the recent research studies in the field. The suggested model is described in the Section 3. The experimental results and discussions on the outcomes are presented in Section 4. Section 5 contains the conclusion and the future recommendations

2. Related works

Prior research in this field could be classified based on sentiment classification methods including machine learning, lexicon-based, and hybrid method. There are several survey studies such as (Medhat et al., 2014; Balazs and Velásquez, 2016; Ravi and Ravi, 2015; Zhou et al., 2014; Sindhwani and Melville, 2008; Nasukawa and Yi, 2003) summarizing the sentiment analysis (SA) in various perspectives. SA approaches are generally divided into two categories that include machine learning or lexicon-based methods. The machine learning-based methods are further divided into unsupervised knowledge and supervised knowledge.

The application of machine-learning to a dataset comprising the movie critics is presented by Lee et al. (Pang et al., 2002). The application of Support Vector Machines (SVM), maximum entropy, and Nave Bayes (NB) enabled this historic breakthrough. Their portrayal of the reviews came in eight different formats, the simplest of which was shown as a unigram. The most effective representations (set of unigram features) and the most efficient machine learning induction approach (SVM) both had 82.9%accuracy. When utilizing a unigram-based feature representation, their NB classification attained an accuracy of 81%. The work is further extended towards implementation of a review system which reports that the review's objective sentences represent the plot of the film, whilst the subjective sentences express the reviewer's feelings about the story. The system was effective in extracting the most opinionated sentences from the reviews. The system resulted the extracts that are 60% of the original review's size with an accuracy level comparable to that of the entire text (Pang and Lee, 2004).

An alternative approach is presented by the Hatzivassiloglou et al. (Hatzivassiloglou and McKeown, 1997), and Turney et al. (Turney and Littman, 2002) that focuses on how to classify the orientation of words rather than entire chapter.

While the aforementioned literature comprises several approaches towards the SA of categorization of Arabic post, these works are limited in several aspects that mainly include: a) limited datasets specifically related to the Arabic contents, b) using only two classes instead of three categories as presented in this study, c) did not using hash vector as feature extraction with different number of features, and d) using ANN with different activation function to compare between them and select the best one. The proposed study instead, presents an intelligent SA of Arabic posts into conservative, revolutionary, and reform while utilising ANN with varying configurations and a larger and diverse corpus (Banan, 2020; Fan, 2020; Chen, 2022).

3. Proposed model

In this study, a new approach based on ANNs with hasher vectors of various sizes was proposed to detect the orientation of the articles in the context of Arabic contents. The proposed study followed the workflow of five stages including: a) primary dataset collection from multiple publicly available sources followed by the manual annotations with three labels (reform, conservative, and revolutionary), b) pre-processing to eliminate unnecessary content, c) feature extraction using multiple sizes of hasher vector, d) implementation of ANNs to classify processed data into three available classes (reform, conservative, and revolutionary), and e) performance evaluation of the trained model over purely unseen
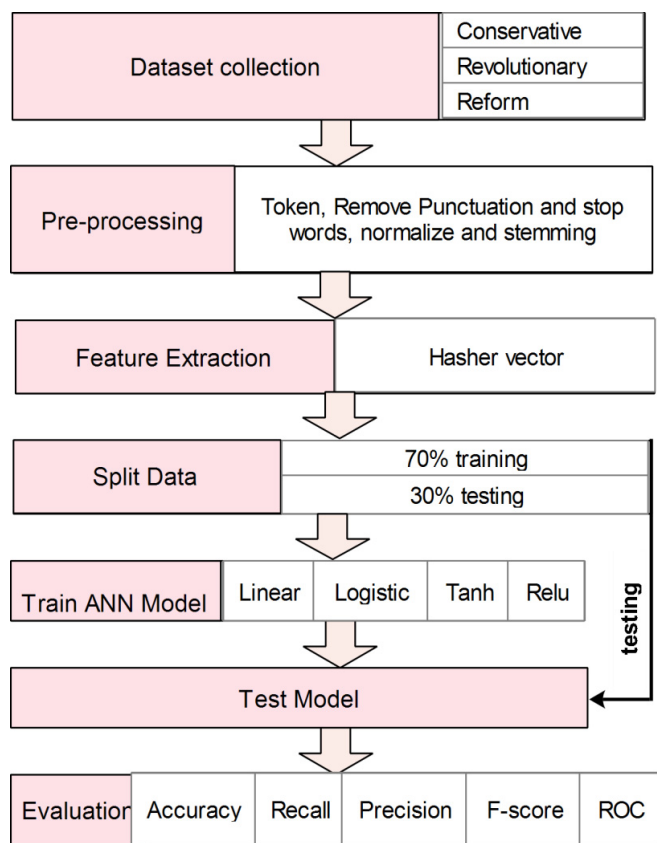

Fig. 1. Sequential procedure in the Proposed Model for Arabic Post.

Table 1
The PAAD dataset.

Statistic	Description	No. of. Instances	Percentage (%)
Class	Reform	80	38.834
	Conservative	58	28.155
	Revolutionary	68	33.009
Length post	Short	≤150	7.281
	Medium	≤250	17.242
	Long	>250	75.242
Number of sentences	Reform	738	34.778
	Conservative	511	24.081
	Revolutionary	873	41.140
All token	Reform	29,607	43.816
	Conservative	14,111	20.883
	Revolutionary	23,853	35.3
Unique token	Reform	6782	44.237
	Conservative	3246	21.172
	Revolutionary	5303	34.590
Highest posts token	Reform	1130	38.124
	Conservative	626	21.120
	Revolutionary	1208	40.755

instances. Fig. 1 shows the flow of the proposed approach utilizing the standard data science analysis for Arabic text classification in which the data was collected from various sources. Due to the raw and unstructured nature of the available dataset, the first step consisted of text tokenization. Punctuation and stop words were then removed for the normalization and stemming of data, which is critical for sentiment analysis via NLP.

3.1. Dataset

In this study, we collected raw data with 206 articles from various resources including online blogs, newspapers, and web sites (Abd et al., 2020). Table 1 summarises the dataset used in this work, which is built upon three class labels. Our PAAD (Arabic Articles Dataset) is available for free download from Mendeley repository under an open-source license. The PAAD dataset consists of statistic, description, number of instances, and percentage. Length of post is divided into short, medium, and long these belong to number of words in each post, also present number of sentences belong to each class, additionally show number of token(words) belong to each class, show number of unique token (word) this important for build the hash vector, finally show highest number of tokens belong to each class to see in results which class effect by the number of tokens.

3.2. Pre-processing

Pre-processing is a critical step specifically for the primary dataset when collected from various sources (Abd et al., 2019). For the proposed dataset, five steps are utilised which are tokenization, elimination of unwanted words, normalization, removal of stop word, and stemming. Tokenization breaks articles into sequence

of words to feed to other steps. Second phase involves the elimination of unwanted words such as English and punctuation. Normalization was applied to standardize the corpus, in other words, this step diacritic and converts alphabetic to other. Table 2 (b) illustrates the conversion of alphabetic to another where we convert nine alphabetic into four alphabetic, all shape of Alph ("إأآا") become as ("ا"), also both alphabetic (ؤ"") and (ئ"") become (ء"") these conversions do not change the meaning of word but help to but different shape of alphabetic in one shape.

The fourth step consists of eliminating stop words, that is used to remove the token when having the same word in list of stop words. This eliminates the need to remove the word that does not make decision and condensed the vector to improve the performance of the ANN. The final step is stemming, in Arabic language there are two types of stemming which are light stemming, and root stemming, in our work, light stemming is utilized which is description in details here (Abd et al., 2021).

3.3. Feature extraction

In this work, hash vector is used for feature extraction (Hasan et al., 2019) mainly due to their efficiency compared to alternative methods such as bag of words, Term Frequency (TF), and Inverse Document Frequency (IDF). These techniques pose computational challenges due to their nature of analysing complete words. Hash vector has lower memory footprint for the storage of words for analysis as illustrated in Fig. 2. It can extract features from list of articles to convert into matrix of tokens (occurrence). It works by mapping the tokens directly to column in a matrix with a predefined size. Fig. 2 shows an example of hash vector used in the present work. Since, hash vector utilizes smaller vectors, in which more focused words can be selected leading to improved accuracy. In the proposed work, various threshold values were investigated from 50 to 15,000 to determine the best word for Arabic text classification.

There are several challenges when dealing with larger dataset and can be resolved by using hash vector. For instance, (i) larger dataset needs large memory to save vocabulary, (ii) require fit data structure for dataset, (iii) word map require full dataset which is difficult to fit through the classifier, (iv) large vocabulary make learning model very slow, (v) it is difficult to split the vector into sub vocabulary (attributes). To overcome all these limitations, hash vector and count vectorization can be utilised as proposed in this study.

Table 2
(a) Example of word cleansing to eliminate the unnecessary diacritics (b) Converting alphabetic.

(a)

Arabic name	English name	Diacritic symbol	Keyboard press	Arabic example
فتحة	Fatha		Shift + Q	بَ
ضمة	Damma		Shift + E	بُ
كسره	Kasra		Shift + A	بِ
السكون	Sakon		Shift + x	بْ
تنوين الكسر	tanween kasra		Shift + S	بٍ
شده	Shada		Shift + Q	بّ
تنوين الفتح	Tanween fatha		Shift + W	بً
تنوين الضم	tanween damma		Shift + R	بٌ
همزء	Hamza	ء	X	بءء

(b)

Original alphabetic	اأآإ	ى	ؤ	ئ	ة
Converting to	ا	ي	ء	ء	ه
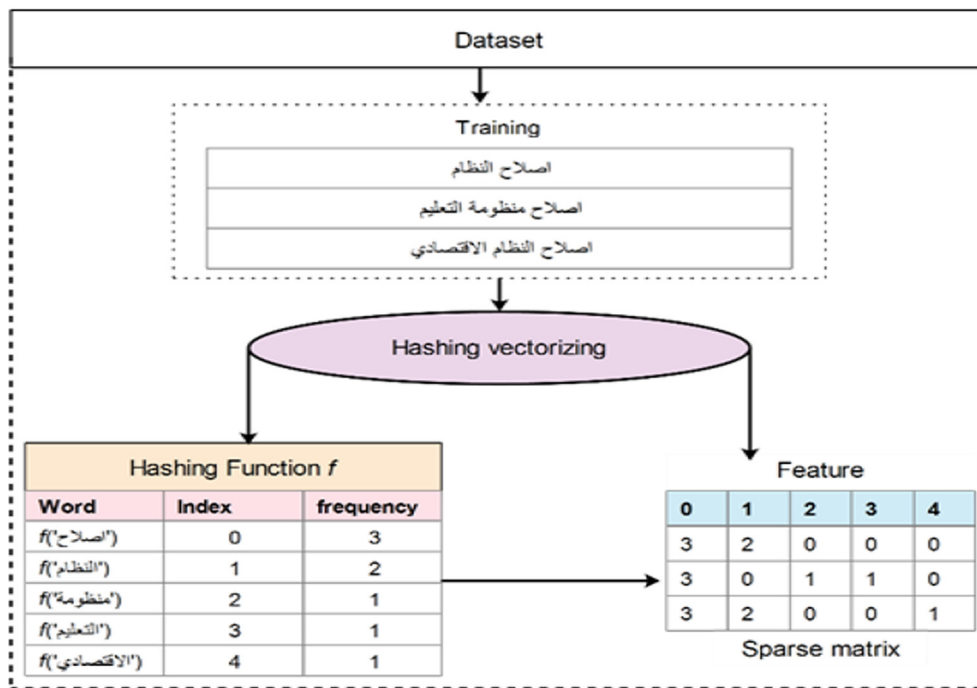
Fig. 2. Example of hashing vectorization providing idea about hash vector and how it builds the spars matrix used to train our ANN model.

3.4. Methodology

Multiple experiments were conducted using the aforementioned dataset and evaluation metrics to measure the performance of proposed model. The configuration for the ANN is set empirically and summarized in Table3, which is determined through trail and errors.

We partition the dataset into training and testing with 70% and 30% proportions, respectively. Hence, the total articles for training set are (144) articles consisting of 43, 46 and 55 articles for conservative, reform and revolution respectively and for testing set there are (62) articles with15, 22 and 25 articles for conservative, reform and revolution classes, respectively.

To evaluate the performance of the proposed model, we used various evaluation metrics that include Accuracy, Recall, F1-score, kappa, and Precision (Abd et al., 2019). Another quality measure is the AUC that provided a relation between true positives and false positives. In this respect, the higher the value of the AUC, the better the algorithm. However, the curve steepness, that determines the maximum of the genuine positive rate while reducing the false positive rate, is equally vital to consider.

3.5. Results and discussions

Various experiments were conducted in which parameter tuning was accomplished by recording an optimal hash vector size from 50 to 15,000 with four activation functions which are (linear, logistic, tanh, and relu). The best model with high accuracy can then be selected. From Table 4, the number (1000) of features showed the best selection with high accuracy of 98.383% for three activation functions (linear, logistic, and tanh). As illustrated in Table 4, logistic activation function showed high accuracy when benchmarked with linear, tanh and Relu functions. Table 5 shows the details for each activation function in which the number of features utilized is (1000).

Model performance was evaluated using well established metrics based on AUC, Recall, F-score, and precision as illustrated in Table 5. AUC at linear, tanh, and relu achieved high value (100%). Recall metric at conservative label for all activation functions achieved similar performance (i.e., 93%). For reform label, all activation functions achieved similar performance (i.e., 100%). For the revolutionary label, (linear, logistic, and tanh achieved same accuracy (100%) while relu function achieved 95%. The precision

Table 3
ANN algorithm parameters.

Parameters	Value	Details
Activation	Linear, logistic, relu, tahn	These Activation used for hidden layer.
Hidden layer	100	The number of hidden layers.
Number of neurons for each layers	100	The number of neurons in each hidden layer.
Solver	Limitedmemory Broyden–Fletcher–Goldfarb–Shanno (Lbfgs)	Lbfgs used for weight optimization when dataset not large, that will be fast and good performance
Alpha	0.0001	Alpha value used for regularization
Learning rate	7	Learning rate for weight updates using (integer number)
Max iteration	200	Number of iterations until training achieved good weights
Random state	Random value	Initialize random value for weight and bias
Max function	15,000	This number used for loss function. The stop training depends on max iteration or reach loss function number.
Tolerance	1e-4	Tolerance used for enhance loss function.

Table 5
Performance evaluation of proposed classification model (conservative, reform, revolutionary).

Activation function	Metric evaluation	Class (%)		
		Conservative	Reform	Revolutionary
Linear	Precision	100	0.96	100
	Recall	0.93	100	100
	F-score	0.97	0.98	100
	AUC	100	100	100
	Accuracy	93.33	100	100
Logistic	Precision	100	0.96	100
	Recall	0.93	100	100
	F-score	0.97	0.98	100
	AUC	0.98	0.98	100
	Accuracy	93.33	100	100
Tanh	Precision	100	0.96	100
	Recall	0.93	100	100
	F-score	0.97	0.98	100
	AUC	100	100	100
	Accuracy	93.33	100	100
Relu	Precision	100	0.93	100
	Recall	0.93	100	0.95
	F-score	0.97	0.96	0.98
	AUC	100	100	100
	Accuracy	93.33	100	93.33

metric for conservative and revolutionary labels is similar (100%) for all activation functions while for reform label, linear, logistic, and tanh achieved 96%. In terms of F-score, revolutionary label achieved higher value (%100) than other labels (conservative and reform) using three activation functions which are linear, logistic, and tanh. The Kappa value for Relu function was of 0.95 relatively lower than the same reported value of 0.97 for Linear, Logistic and Tanh.

The kappa metric was used to evaluate proposed model for each activation function. This metric can provide details about the proposed classification technique reliability. It can be noticed that e linear, logistic, and tanh achieved higher value (97.52%) and for relu achieved (95.03%). Fig. 3 shows the learning rate for activation function with size 1000 for feature extraction.

The precision and recall for each label (conservative and revolutionary) using all activation functions achieved 100%, where reform label for all activation function achieved 0.99%. As shown in Table 4, three activation functions work very well when the number of features to build hash vector is small, these functions

are linear, logistic, and tanh achieving high accuracy at number (1000). Relu activation function works better than other activation functions when the number of features is high as seen in Table 4 with the number of features at (9000 and 12000) achieved accuracy higher than other around (98.3%). Table 5 indicates that the reform and conservative classes outperform for all activation functions. On the other side, revolutionary class performs better for three activation function which are (linear, logistic, and tanh) with (100% accuracy) while comparatively lower for the relu (0.95%). Precision metric for revolutionary and conservative label achieved higher value for all activation function. Finally, the best label performance from the proposed model is achieved for the revolutionary label (100%) for all metrics and activation functions.

Likewise, the best performing activation functions are logistic, linear, and tanh indicating over 90% accuracy for all labels. Table 6 shows the accuracy of our proposed model for 10 folds using four activation functions and best 1000 features.

In our work we utilized accuracy, precession, recall, f-score, and ROC-AUC with two statistic metrics which are mean and standard

Table 4
Number of feature selection with four activation function to select best number with best activation function.

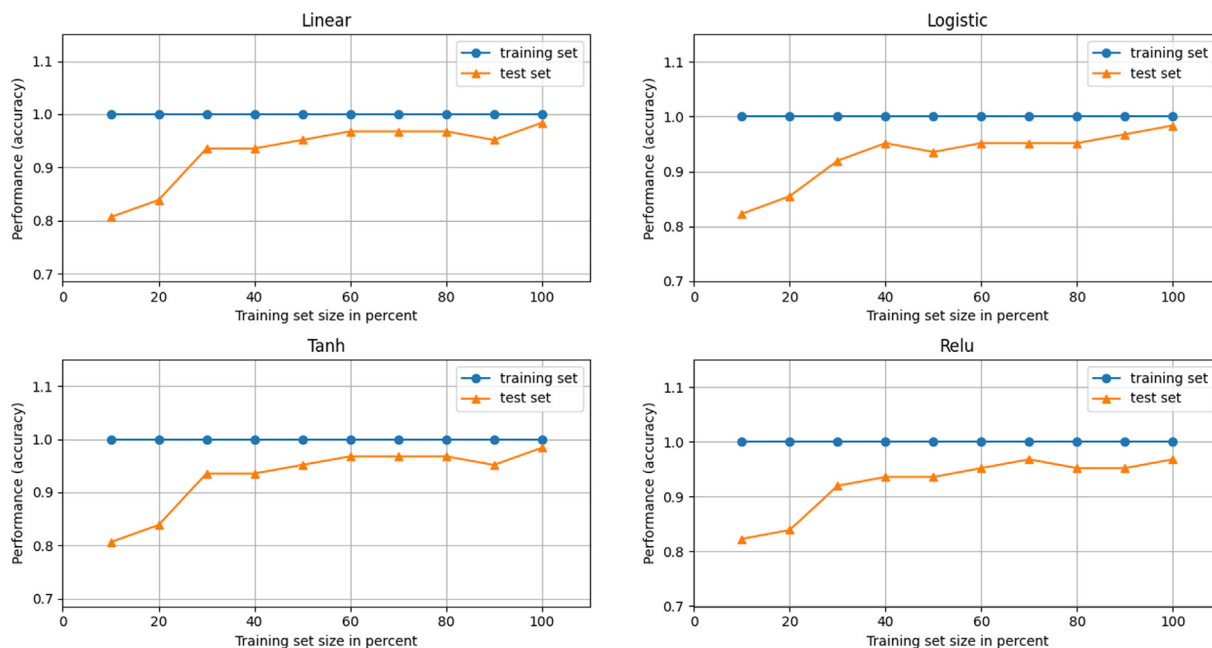| No. of features	Activation function (Accuracy%)				Selection of best activation function
	Linear	Logistic	Tanh	Relu	
50	80.645	**82.258**	79.032	80.645	Logistic
100	**93.548**	90.322	**93.548**	88.709	Linear and tanh
150	88.7	88.7	88.7	88.7	=
250	**93.5**	**93.5**	**93.5**	91.9	Linear, Logistic, and tanh
400	95.1	95.1	95.1	**96.7**	Relu
700	91.9	**93.5**	91.9	91.9	Logistic
1000	**98.3**	**98.3**	**98.3**	96.7	Linear, Logistic, and tanh
1500	96.7	96.7	96.7	96.7	Linear, Logistic, and tanh
2000	**96.7**	**96.7**	**96.7**	95.1	Linear, Logistic, and tanh
2500	**95.1**	**95.1**	**95.1**	93.5	Linear, Logistic, and tanh
3000	95.1	95.1	95.1	**96.7**	Relu
4000	**96.7**	**96.7**	**96.7**	95.1	Linear, Logistic, and tanh
5000	**95.1**	**95.1**	**95.1**	93.5	Linear, Logistic, and tanh
7000	**95.1**	**95.1**	**95.1**	93.5	Linear, Logistic, and tanh
9000	95.1	95.1	96.7	**98.3**	Relu
11,000	95.1	95.1	95.1	**96.7**	Relu
12,000	96.7	96.7	96.774	**98.3**	Relu
15,000	**96.7**	**96.7**	**96.7**	95.1	Linear, Logistic, and tanh

Fig. 3. Learning rate using four activation functions for size 1000 feature extraction.

Table 6
K-fold 10 with four activation functions.

Activation function	Accuracy		Recall		Precession		F-score		ROC-AUC	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
Linear	0.937	0.064	0.937	0.064	0.95	0.045	0.934	0.073	0.993	0.016
Logistic	0.937	0.064	0.927	0.034	0.949	0.041	0.934	0.07	0.992	0.018
Tanh	0.937	0.064	0.937	0.064	0.95	0.045	0.934	0.073	0.993	0.015
Relu	0.942	0.052	0.942	0.052	0.954	0.036	0.94	0.056	0.994	0.015

deviation (STD). As it can be shown from Table 6, with 10 K-fold, the model achieved good accuracy and 0.064 using the standard deviation. This means that activation functions (linear, logistic, Tanh) can provide high accuracy.

Table 7 shows the benchmark with various techniques available in the literature. As it can be shown, our proposed solution indicated improved accuracy for light steaming.

As it can be noted, there are various techniques available in the literature for Arabic text classification including SVM (Abd et al., 2019), KNN (Alwan et al., 2021), decision trees (Abd et al., 2020), and Rough set theory (Alwan et al., 2021), our proposed method utilized smaller feature set with 1000 words which provided improved computational complexity and memory utilization as well as accuracy.

Table 7
Benchmark with the available works in the –literature.

Ref	objective	data/methd	accuracy	limitation
(Al-Radaideh and Al-Khateeb, 2015)	Categorize medical Arabic document in disease class	Using set of associative rules.	90.6	Not opinion Arabic post
(Abooraig et al., 2018)	Categorize resources into Arab nationalist, Islamic Shia, socialist, liberal, and brotherhood	Using Bag of Words (BOW) and machine learning	90.17	Not categorize science in three class as we did
(Al-Radaideh and Al-Abrat, 2019)	categorize Arabic text into (Art, Economy, Health, Law, Literature, Religion, Sport, and Technology)	Using rough set theory with Term frequency – inverse document frequency (TF-IDF)	94	Not opinion Arabic
(Abd et al., 2019)	Classify Arabic article	Support vector machine with TF-IDF	95.161	Low accuracy and time consuming because use SVM
(Abd et al., 2019)	Classify Arabic documents	Four different nave bayes with Term Frequency (TF) feature extraction	96.77	Word order information is lost in TF
(Alwan et al., 2021)	Categories Arabic article	Using rough set theory with lexicon and five grams. Best accuracy with unigram	85.483	Low accuracy
(Abd et al., 2020)	Classify Arabic articles	Using Decision Tree (DT) and KNearest Neighbor (KNN) with TF	DT = 87.096 KNN = 77.419	Both algorithms are low accuracy
Propsoed model	Categorize Arabic post	Using hash vector with four activation function in ANN	**98.3**	Using only light stemming, in feature use root stemming

4. Conclusions

This study presents an automated categorization approach for Arabic posts using ANN and a primary dataset acquired from various online sources. The experimental outcomes showed that an artificial neural network classifier is able to classify Arabic posts into three categories {conservative, reform, revolutionary}. Our goal was to interrogate the impact of the size of PAAD dataset on model classification accuracy and efficiency by accompanying hash vectors with ANN model. Utilizing hash vector with various transfer functions including linear, logistic, tanh, and sigmoid activation functions provided reasonable accuracy of ∼ 98.3 % with size vector of 1000.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jksus.2023.102733.

References

Abd, D.H., Abbas, A.R., Sadiq, A.T., 2020. Analyzing sentiment system to specify polarity by lexicon-based. Bull. Electrical Eng. Informatics 10 (1), 283–289.

Abd, D.H., Sadiq, A.T., Abbas, A.R., 2019. 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, pp. 79-94.

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

Abd, D.H., Sadiq, A.T., Abbas, A.R., 2020. Political Arabic Articles Classification Based on Machine Learning and Hybrid Vector. In: 2020 5th International Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA), IEEE, pp. 1-7.

Abd, D.H., Khan, W., Thamer, K.A., Hussain, A.J., 2021. Arabic Light Stemmer Based on ISRI Stemmer. In: International Conference on Intelligent Computing, Springer, pp. 32-45.

Abd, D.H., Sadiq, A.T., Abbas, A.R., 2020. PAAD: Political Arabic articles dataset for automatic text categorization. Iraqi J. Computers Informatics 46 (1), 1–10.

Abooraig, R., Al-Zu'bi, S., Kanan, T., Hawashin, B., Al Ayoub, M., Hmeidi, I., 2018. Automatic categorization of Arabic articles based on their political orientation. Digit. Investig. 25, 24–41.

Al-Radaideh, Q.A., Al-Abrat, M.A., 2019. An Arabic text categorization approach using term weighting and multiple reducts. Soft. Comput. 23 (14), 5849–5863.

Al-Radaideh, Q.A., Al-Khateeb, S.S., 2015. An associative rule-based classifier for Arabic medical text. Int. J. Knowledge Eng. Data Mining 3 (3–4), 255–273.

Alwan, J.K., Hussain, A.J., Abd, D.H., Sadiq, A.T., Khalaf, M., Liatsis, P., 2021. Political Arabic articles orientation using rough set theory with sentiment lexicon. IEEE Access 9, 24475–24484.

Amarouche, K., Benbrahim, H., Kassou, I., 2015. Product opinion mining for competitive intelligence. Procedia Comput. Sci. 73, 358–365.

Badaro, G. et al., 2019. A survey of opinion mining in Arabic: a comprehensive system perspective covering challenges and advances in tools, resources, models, applications, and visualizations. ACM Trans. Asian Low-Resource Language Information Processing (TALLIP) 18 (3), 1–52.

Balazs, J.A., Velásquez, J.D., 2016. Opinion mining and information fusion: a survey. Information Fusion 27, 95–110.

Banan, A. et al., 2020. Deep learning-based appearance features extraction for automated carp species identification. Aquac. Eng. 89, 102053.

Biltawi, M., Etaiwi, W., Tedmori, S., Hudaib, A., Awajan, A, 2016. Sentiment classification techniques for Arabic language: a survey, in: 2016 7th International Conference on Information and Communication Systems (ICICS), IEEE, 2016, pp. 339-346.

Chen, C.C. et al., 2022. Forecast of rainfall distribution based on fixed sliding window long short-term memory. Eng. Appl. Comput. Fluid Mech. 16 (1), 248–261.

Duwairi, R., El-Orfali, M., 2014. A study of the effects of preprocessing strategies on sentiment analysis for Arabic text. J. Inf. Sci. 40 (4), 501–513.

Fan, Y.J. et al., 2020. Spatiotemporal modeling for nonlinear distributed thermal processes based on KL decomposition, MLP and LSTM network. IEEE Access 8, 25111–25121.

Farghaly, A., Shaalan, K., 2009. Arabic natural language processing: challenges and solutions. ACM Trans. Asian Language Information Processing (TALIP) 8 (4), 1–22.

Hasan, M., Islam, I., Hasan, K.A., 2019, Sentiment analysis using out of core learning. In: 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), IEEE, pp. 1-6.

Hatzivassiloglou, V., McKeown, K.R., 1997. Predicting the semantic orientation of adjectives. In: Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the European chapter of the association for computational linguistics, Association for Computational Linguistics, pp. 174-181.

Liu, B., 2012. Sentiment analysis and opinion mining. Synthesis lectures on human language technologies 5 (1), 1–167.

Medhat, W., Hassan, A., Korashy, H., 2014. Sentiment analysis algorithms and applications: a survey. Ain Shams Eng. J. 5 (4), 1093–1113.

Mitra, A., 2020. Sentiment analysis using machine learning approaches (Lexicon based on movie review dataset). J. Ubiquitous Comput. Commun. Technologies (UCCT) 2 (03), 145–152.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., 2013. A. Ritter, and T. Wilson, "Task 2: Sentiment analysis in twitter. In: Proceedings of the 7th International Workshop on Semantic Evaluation, Atlanta, Georgia.

Nasukawa, T., Yi, J., 2003. Sentiment analysis: Capturing favorability using natural language processing. In: Proceedings of the 2nd international conference on Knowledge capture, ACM, pp. 70-77.

Pang, B., Lee, L., 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd annual meeting on Association for Computational Linguistics, Association for Computational Linguistics, p. 271.

Pang, B., Lee, L., Vaithyanathan, S., 2002. Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing. Vol. 10, Association for Computational Linguistics, pp. 79-86.

Ravi, K., Ravi, V., 2015. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. Knowl.-Based Syst. 89, 14–46.

Sadia, A., Khan, F., Bashir, F., 2018. An overview of lexicon-based approach for sentiment analysis. In: 2018 3rd International Electrical Engineering Conference (IEEC 2018), pp. 1-6.

Sindhwani, V., Melville, P., 2008. Document-word co-regularization for semi-supervised sentiment analysis. In: 2008 Eighth IEEE International Conference on Data Mining, IEEE, pp. 1025-1030.

Turney, P.D., Littman, M.L., 2002. Unsupervised learning of semantic orientation from a hundred-billion-word corpus," arXiv preprint cs/0212012.

Velichkov, B. et al., SU-FMI: System Description for SemEval-2014 Task 9 on Sentiment Analysis in Twitter. In: SemEval@ COLING, Citeseer 2014, pp. 590-595.

Zhou, S., Chen, Q., Wang, X., 2014. Fuzzy deep belief networks for semi-supervised sentiment classification. Neurocomputing 131, 312–322.