Neural Processing Letters
https://doi.org/10.1007/s11063-022-11111-1

Improving the Polarity of Text through word2vec Embedding for Primary Classical Arabic Sentiment Analysis

Nour Elhouda Aoumeur[1,2] · Zhiyong Li[1,2] · Eissa M. Alshari[3]

Accepted: 10 December 2022

Abstract

Over the past decade, Sentiment analysis has attracted significant researcher attention. Despite a huge number of studies in this field, Sentiment analysis of authors' books (classical Arabic) with extracting the embedding features has not yet been done. The recent feature extraction of Arabic text depends on the frequency of the words within the corpus without extracting the relation between these words. This paper aims to create a new classical Arabic dataset CASAD from many art books by collecting sentences from several stories with human-expert labeling. Additionally, the feature extraction of those datasets is created by word embedding techniques equivalent to Word2vec that are able to extract the deep relation which means features of the formal Arabic language. These features are evaluated by several types of machine learning for classical Arabic, for example, support vector machines (SVM), Logistic Regression (LR), Naive Bayes (NB) K-Nearest Neighbors (KNN), Latent Dirichlet Allocation (LDA) and Classification And Regression Trees (CART). Moreover, statistical methods such as validation and reliability are applied to evaluate this dataset's label. Finally, our experiments evaluated the classification rate of the feature-extraction matrices in two and three classes using six machine-learning algorithms for tenfold cross-validation that showed that the Logistic Regression with Word2Vec approach is the most accurate in predicting topic-polarity occurrence.

Zhiyong Li and Eissa M. Alshari have been contributed equally to this work.

✉ Nour Elhouda Aoumeur
aoumeur89@gmail.com

Zhiyong Li
zhiyong.li@hnu.edu.cn

Eissa M. Alshari
alsharieissa@ibbunv.edu.ye

[1] College of Computer Science and Electronic Engineering, Hunan University, Lushan, Changsha 410082, Hunan, China

[2] Key Laboratory for Embedded and Network Computing of Hunan Province, Hunan University, Lushan, Changsha 410082, Hunan, China

[3] Computer Science, Ibb University, Ibb, Yemen

Published online: 23 January 2023

Keywords Arabic sentiment analysis · Authors · Classical Arabic · Feature extraction · Machine learning · Word2Vec

1 Introduction

Sentiment analysis (SA) is related to computational linguistics, natural language processing, and text mining [1]. It is the process of quantifying the emotional value in a series of words or text to gain an understanding of the expressed attitudes, opinions, and emotions. Sentiment Analysis also called Mood Analysis is a technique for determining the sentimental qualities of a given text based on the words it contains [2]. It is also called polarity detection, sentiment detection is one of several tasks of opinion mining [3], or sentiment classification which objective to classify sentiment data into polarity categories (e.g., positive, negative or neutral) [4, 5].

Sentiment analysis can be applied to various sectors, such as e-commerce, banking, and mining social-media websites like Facebook and Twitter [6]. It can be defined as a type of data mining that measures the inclination of people's opinions through natural language processing (NLP). SA has become one of the essential research fields whose application is clearly visible in a variety of domains such as politics, commerce, tourism, education, and health [7, 8].

There has been wide interest in sentiment analysis of the research community in the last decade. Much research has been done in this field for many purposes purposes [9], for example, working with movie scripts. Most studies on Arabic sentiment analysis used the most popular algorithms in machine learning, i.e., Naive Bayes (NB), Logistic Regression (LR), and support vector machines (SVM), because of their high accuracy rate regardless of the robustness of the user data and because of the ease with which they can be implemented [10]. We conducted binary sentiment classification using three classifiers: NB, SVM, and LR. Two corpora were used:the first was developed by these authors and is composed of two domain-specific datasets (movies and sports) [11, 12].The second is by Opinion Corpus Arabic (OCA); they developed a corpus of movie reviews [13].

The Arabic language is a difficult language for automatic processing. This is due to several intrinsic reasons such as Arabic multi-dialects, syntactic flexibility, and diacritics. Both machine learning and deep learning frameworks require large data sets for training to ensure accurate predictions. This leads to another challenge facing research using Arabic text; because high-quality Arabic text datasets are still scarce.

Arabic is the official language of 22 countries, with over 30 crore speakers in the world. There are 28 alphabets in Arabic, and they should be written from right to left. It is heavily inflected, with rich morphology and complex syntax [14]. It is also one of the popular languages in the world. As mentioned by statistical studies in 2019 [15] the language is spoken by almost 319 million people and ranks fifth among world languages after Chinese, Spanish and English.

There are three categories of Arabic: Dialectal Arabic (DA), Classical Arabic (CA) and Modern Standard Arabic (MSA). DA varies between countries and can also vary between areas of the same country [16, 17]. CA is more common in literature, science, and writing, while MSA is more commonly spoken. Most Arabic sentiment-analysis datasets that were used in the current sentiment-analysis approach were in MSA. There is a great scarcity in the study of CA for sentiment analysis of authors' texts and literature. The benefit of analyzing their texts, understanding their feelings when they wrote their books. It also helps to understand the situation in which they were living and even what they wanted to convey to the readers. It helps to understand their books and eliminate ambiguity because there are many problems, especially with the Arabic language movement that could change the meaning if these movements are not written [18, 19]. Many efforts have investigated the Arabic language, whether to analyze the text analyze sentiment [20] translate statements [21], or detect depression [22]; all these applications require the existence of comprehensive Arabic datasets. Building a dataset is not an easy task, as it requires tremendous effort, time and cost. Also, the recent application of machine learning and deep learning requires huge datasets that contain billions of records. Therefore, the creation of a dataset of literary works adds new horizons in the sentiment analysis of rigid texts.However, this does not deny the reality that Arabic is considered a highly ambiguous language, especially when trying to analyze, classify and process Arabic data automatically.

In this paper, the word2vec approach is implemented to extract the vector of features from the Classic Arabic Sentiment Analysis Dataset (CASAD) that are collected from several books. CASAD was judged by human experts with statistical-evaluation (validity and reliability) approaches. Next, feature- extraction approaches, such as count vectors, term frequency-inverse document frequency (TFIDF) and Word2Vec were applied to represent the vector weighting of CASAD features and thus these vector representations are used through training by machine-learning (ML) approaches. Finally, ML three different algorithms were implemented to analyze CASAD data with cross-validation evaluation.

Most related works in the areas of Arabic sentiment analysis are presented in Sect. 2. Section 3 outlines the data-collection methods, statistical evaluation tools, and feature-extraction and ML-training approaches. Section 4 shows the results and discussion, and Sect. 5 concludes the paper.

2 Related Works

Many researchers have applied machine learning algorithms for subjectivity and SA.A study in sentiment analysis was done by Al-Sabbagh et al. Crawled and annotated a general Twitter Arabic corpus for subjectivity and sentiment analysis, which was the first available corpus for SSA, as the researchers claimed [23]. An important amount of works has been done in Arabic sentiment analysis (ASA). A hybrid feature selection method used in the Arabic sentiment analysis to extract users' opinions of Saudi governmental applications for COVID-19. They developed a new Arabic dataset that includes 7759 reviews collected from Google Play and the app store. Different methods are applied to the dataset and the results show that the k nearest neighborhood (KNN) method generates the highest accuracy compared to other implemented methods [24]. In [25] They have distributed representations of documents modeled for Arabic Sentiment Analysis. They study the effect of various variable parameters' setup of Doc2Vec model on the four machine learning methods: LR, DT, SVM, and NB are used to detect sentiment with Two architectures of Doc2v applied to the datasets consist of 33K annotated reviews that are collected from different websites for movies, hotels, restaurants, and products. Doc2Vec with both large Dimensions and Negative Samples is better to achieve high effectiveness with classifiers.

A multi-facet sentiment analysis system was proposed in [26]. They first collected multi-domain datasets and lexicons. Since the manual construction and verification of a lexicon is time-consuming, They have performed classifi- cation based on the unsupervised; and the classical and deep neural supervised approaches. They have gained interest in data analytics, thus they have opted for existing and custom methods to optimize the characteristic vector of the opinionated reviews.

Several algorithms and machine learning approaches have been used to construct sentiment analysis systems in various domains [27], the number of publications in ASA has increased exponentially over the past few years.

A feature-based sentence-level approach to Arabic sentiment analysis was presented in another study [28, 29]. The new approach in this study used an Arabic idioms/saying-phrase lexicon as an important key to improving the detection of sentiment polarity in a sentence. The authors also used a number of novels and rich sets of linguistically motivated features (contextual intensifiers, contextual shifters, and negation handling), syntactic features for conflicting phrases to increase the accuracy rate of sentiment classification. In addition, they introduced an automatic expandable wide-coverage polarity lexicon of Arabic sentiment. Their experiment results with an SVM classifier gave high- performance levels, with an accuracy rate of over 95%.

Another study in sentiment analysis for stock-market purposes was per-formed in [30]. The leading stock-analysis software provider in GCC countries, with twitter for opinion-mining purposes have been studied. They extracted feedback from MUBASHER by designing a model suits sentiment analysis of Saudi Arabic tweets. Their new model combined a machine learning approach with an NLP approach to classify Arabic tweets into sentiment polarity classes:

Positive, negative, and neutral. They showed Arabic text classification using three different algorithms, Naıve Bayes, KNN, and SVM [31], and they claimed that the best accuracy rate was obtained by using KNN with a dataset of 1943 tweets that were collected and used in this study. Focusing on unemployment in Saudi Arabia as the problem, they proposed a method for sentiment analysis of Arab tweets depending on lexical normalization and supervised machine learning, specifically the SVM and NB algorithms, to set the polarity of each tweet's sentiment [32]. In [33] They presented a new Arabic sarcasm dataset. The dataset was created by re-annotation of the available datasets on Arabic sentiment. The new dataset contains sarcasm, sentiment and dialect labels.

3 Materials and Methods

The Arabic language is viewed as one of the top 10 main languages that are used on the web, but it is acknowledged as a poor content-language, unlike English, with very few web pages containing Arabic reviews and feedback [31].

In this section, the different steps of our architectural system that described in Fig. 1 are proposed. First, CASAD is initialed by collecting from the Arabic books. Next, the judges evaluate the dataset and thus the Cronbach's alpha is applied to evaluate the judge's scores.

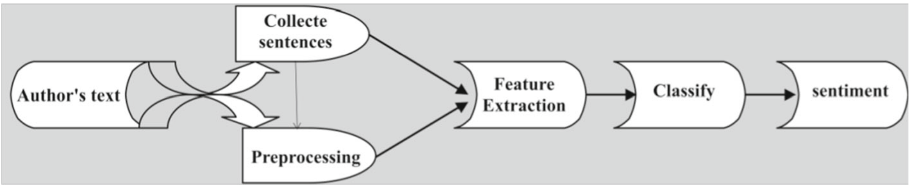


Fig. 1 Arabic sentiment analysis system architecture

Table 1 Reliability statistics

Cronbach's alpha	Cronbach's alpha based on standardized items	N of items
0.804	0.80	3

Table 2 Inter-item correlation matrix

	Judge1	Judge2	Judge3
Judge1	1000	0.397	0.511
Judge2	0.397	1000	0.808
Judge3	0.511	0.808	1000

Then, the feature-extraction vector is created by several methods (TFIDF, word embeddings). Finally, the CASAD feature-extraction vectors are trained using several ML algorithms, with cross-validation in the classification phase.

3.1 Data Collection

In this study, several Arabic books by the most famous authors were read, and each individual sentence was extracted and manually tagged.

- Data were randomly collected from a variety of books and manually from the Internet (human development, history, fairy tales, novels, and medicine) without being repeated. These books were collected from Large-Scale Arabic Book Review (LABR) [34, 35], a dataset of over 63,257 book reviews collected from www.goodreads.com.
- The collected data were separated into 9709 paragraphs.
- Paragraphs were sent to three Arabic native human experts with a Ph.D.

Degree in different faculties to express their opinions on the text.

- The statistical approaches (validity and reliability) were applied using SPSS to evaluate the experts' judgment. A Cronbach's Alpha reliability coefficient value of greater than 0.6 is suggested to be accepted (1).

Table 1 shows the measurement of reliability statistics, that is, Cronbach's alpha achieved 80.4%. This ratio reflects the overlapping similarity in Table 2 between the experts and, thus, the result is accepted because the value is in the acceptance area [2].

- Finally, Fig. 2 shows that the collected data were almost 10,000 paragraphs; therefore, the experts excluded around 3000 sentences and accepted the rest, which contained 454 negatives, 802 neutral, and 7047 positive entries. Table 3 shows the CASAD sample extracted from Arabic books as an example.

3.2 CASAD Preprocessing

Book sentences are sometimes written informally and contain many errors and shortcuts. Therefore, CASAD has to clean and filter. One of the main functions of NLP is to clean and reduce impurities to prepare the corpus for more processing. Most preprocessing steps that were used in this research are:

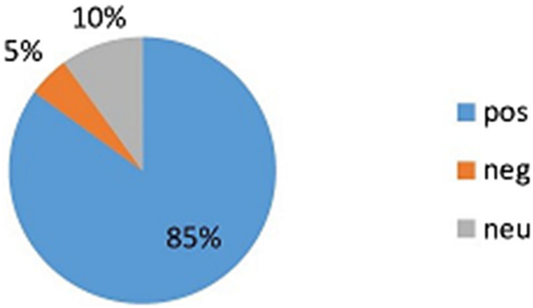
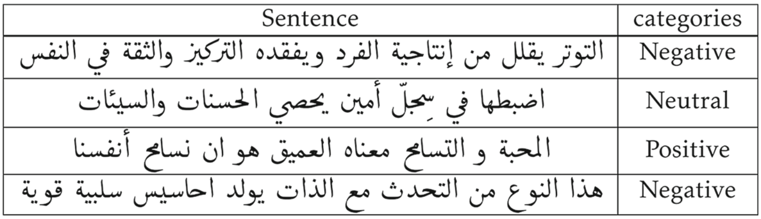


Fig. 2 Classification of book-author ratings

Table 3 Example of sentence extracted from Arabic books



Sentence	categories
التوتر يقلل من إنتاجية الفرد ويفقده التركيز والثقة في النفس	Negative
اضبطها في سِجلّ أمين يحصي الحسنات والسيئات	Neutral
المحبة و التسامح معناه العميق هو ان نسامح أنفسنا	Positive
هذا النوع من التحدث مع الذات يولد احاسيس سلبية قوية	Negative

3.2.1 Tokenization

For studying the polarity of each document or comment, it is necessary to divide the comment into sentences and then into words using NLP preprocessing. Each comment is divided into sentences using NLP processing. Then, the sentence is split up into words or terms by tokenization. Tokenization could be defined as one of the parsing tasks that discriminate a string of input characters to some other form or figure. All of these tokens are used in parsing, such as stemming and part-of-speech tagging on the one hand, or calculating the probability of word position in the text using word embeddings techniques on the other hand.

3.2.2 Stemming

Stemming is a process that reduces redundant tokens in a corpus by removing the suffix and prefix of words. Simply, it is a conversion of plurals to singulars or the derivation of a verb from the gerund form. There are other possibilities, such as deriving the root of the pattern word. For example, the root of the words علم (أعلام:Flags), (تعليم:Education), (معلم:Teacher), (العلم:Science), (معلوم:Known), (معلومة:Information), (عالم:Scientist). Stems are used in the feature selection methods. They are the roots of every similar word morphologically. In the Arabic language, roots or stems are mostly three or four letters. The importance of the steaming process is in the classification and index builders/searchers because it makes the operations less dependent on particular forms of words and reduces the potential size of vocabularies, which might otherwise have to contain all possible forms.

3.2.3 Part-of-Speech Tagging

Part-of-speech (POS) tagging is the process of marking up a word in a sentence corresponding to a particular part of speech, based on both its definition and its context. It has been used for a variety of SA tasks, such as SentiWordNet, and is extremely useful since it provides a linguistic signal on how a word is being used within the scope of a phrase, sentence, or document.

3.2.4 Stop-Word Filtering

Basically, stop words are a set of frequent words in any language; for example the words (“and:و”,”but:لكن”,”how:كيف”,”or:أو”, and”what: ماذا أو ما”). Therefore, stop words are usually removed from the corpus, and thus the proposed approach only focuses on the important words (nonstop words).

3.3 TFIDF Feature Extraction

The first step in this study was collecting a corpus of 8303 documents and human experts manually classifying them into three categories; positive, neg- ative, and neutral paragraphs. In this part, the features that are represented in the feature-extraction vector are the following:

Documents are represented as a vector of words, e.g., the success for every category of the collected dataset. This success *(النجاح)* can be denoted as $|V|$ using a count vector and a term frequency inverse term as:

1. Log normalization of the term frequent *TF* as:

$$\mathrm{TF} = \log f_{t,d} \tag{1}$$

   where $f_{t,\,d}$ is the raw count of a term t a document d.

2. Inverse document frequency *IDF* is a measure of how much information the word provides, i.e., if it is common or rare across all documents as:

$$IDF = \log \frac{N}{n_t} \tag{2}$$

   where N: Total number of documents in the corpus $N = |D|$ $N = |D|$

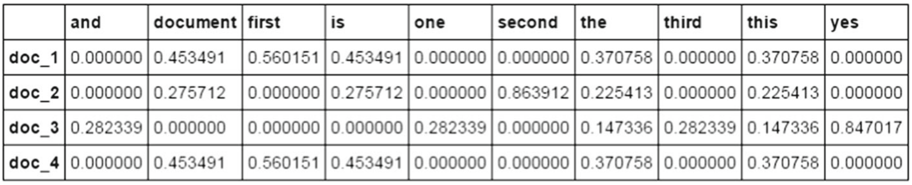
3. then, term frequency-inverse document frequency *TFIDF* is calculated by Eqs. 1 and 2 as:

$$\begin{aligned} \mathrm{TFIDF}(t, d, D) &= \mathrm{TF}(t, d) \cdot \mathrm{IDF}(t, D) \\ &= 1 + \log f_{t,d} . \log \frac{N}{n_t} \end{aligned} \tag{3}$$

   where N: Total number of documents in the corpus $N = |D|$ $N = |D|$

$|\{d \in D: t \in d\}|$ $|\{d \in D: t \in d\}|$:number of documents where the term t appears (i.e., $\mathrm{TF}(t, d) \neq 0$ $\mathrm{TF}(t, d) \neq 0$). If the term is not in the corpus, this will lead to a division by zero.It is, therefore, common to adjust the denominator to $1 + |\{d \in D: t \in d\}|$$1 + |\{d \in D: t \in d\}|$.

The Word2Vec and TFIDF with the TFIDF for each word are stored in document matrices, such as in Fig. 3. This feature-extraction vector is used in ML approaches to training the feature weight that aids in the significant classifying of the CASAD.



	and	document	first	is	one	second	the	third	this	yes
doc_1	0.000000	0.453491	0.560151	0.453491	0.000000	0.000000	0.370758	0.000000	0.370758	0.000000
doc_2	0.000000	0.275712	0.000000	0.275712	0.000000	0.863912	0.225413	0.000000	0.225413	0.000000
doc_3	0.282339	0.000000	0.000000	0.000000	0.282339	0.000000	0.147336	0.282339	0.147336	0.847017
doc_4	0.000000	0.453491	0.560151	0.453491	0.000000	0.000000	0.370758	0.000000	0.370758	0.000000

Fig. 3 Feature extraction sample from word2vec as document matrix

3.4 word2vec Feature Extraction

Sentiment Analysis involves a combination of natural language processing (NLP) and text mining. Many of the studies are dealing with SA for the English language. However, there are restricted numbers of studies about SA in Arabic [3]. To overcome the complexity of Arabic, Word embedding or word distributing approach is used to progress NLP in SA tasks.

Word embeddings is a technique capable of estimating the probability of the word domain position and the relationship between them within the text. The most important of these techniques is called Word2Vec [4, 5], which uses the techniques of deep learning in the extraction of relations between words. Word2vec model has achieved a good result in with SA Arabic language.

Word2Vec includes two models: the continuous bag-of-words model (CBOW) and the skip-gram (SG) model. The CBOW method uses the con- text to calculate the next word, and the SG model uses the word to predict the context. Word2vec is used in combination with other algorithms in order to accurately classify sentiments. The SG model and the CBOW models are opposites, but they are both effective architectures for allowing the neural networks to learn words and their context [6].

It is a two-layer neural network, one of these layers is called the input layer that is a textual corpus, while the outer layer is a set of vectors that is called feature vectors for words in that corpus as output. Then, the vocabulary and corpus will be trained using Word2vec models as the following algorithms:

The first component of the method deals with the discovery of the word representation based on Word2vec model. Given that a corpus *D* consists of a set of texts, D = {d_1, d_2, d_3,…, d_n} and a vocabulary T = {t_1, t_2, t_3,…, t_m} consists of unique terms extracted from D. Then, the word representation of the terms they are discovered by using the Skip-gram model of the word2vec [7] to calculate the probability distribution of other terms in context given t_i. In particular, t_i is represented by a vector $^{\rightarrow}v_i$ that is comprised of probabilistic values of all terms in the vocabulary.

This word embedding technique can discover semantic relationships among terms in the corpus. However, the resulting set of vectors for all terms in the corpus is high-dimensional and is inefficient for the classifier in the SA task. As a result, this first component discovers a set of vector.

$V_T = \{^{\rightarrow -}v_1, ^{\rightarrow -}v_2, ^{\rightarrow -}v_3, ..., ^{-}v^{\rightarrow}m\}$ representing the set of terms in the vocabulary T.

3.5 Machine-Learning Training

In this section, CASAD features are trained to estimate the probability of each feature (f_n) in the feature-extraction vector with each category in the dataset. A calculation process was performed to implement this step as follows:

- Computing the average of each category (C_j) in collecting dataset CASAD by dividing the number of each category by the total number of all categories in the collected dataset.
- Separately computing the probability of each word (w_n) within each category (C_j) in the CASAD by the following equation:

$$P(w_n|C_j) = \frac{X_n + 1}{X + |\mathrm{V}|} \tag{4}$$

where: $X = \{x_1, x_2, x_3,\ldots, x_n\}$ represents some features number of words (independent variables) in the given category C_j in the collected dataset, and X_n is the number of times the word occurs in each category C_j in the collected dataset.

- Using several ML approaches to evaluate the dataset and feature extraction. For example, one of the ML training approaches is Naive Bayes that computes the value of (V_{NB}) using the following equation:

$$(V_{NB}) = agrmax P(C_j) \prod_{X_n \in W} P(X_n|C_j) \tag{5}$$

- Finally, the confusion matrix of each ML approach is used to evaluate the CASAD in two scenarios, multi (3) and binary (2) classes.

4 Experimental Discussion and Analysis Results

In our approach, two feature-extraction methods, Word2Vec and TFIDF. The 5% of the dataset is used to solve the imbalance number of each class. In addition, six machine learning algorithms (LR, LDA, KNN, CART, NB, SVM, adat, Adapt real) were examined to evaluate the classification rate of the feature extraction matrices in two and three classes, were examined to evaluate the classification rate of the feature extraction matrices in multi-classes (Positive (Pos), Negative (Neg), and Neutral (Neu)) and binary (Positive and Negative) classes.

In information retrieval, natural language processing, and classification problems, the confusion matrix is generated to tabulate the performance of any classifier. This matrix shows the relation between correctly and wrongly predicted author texts. From this confusion matrix, different Performance evaluation parameter like Precision (6) are a measure of result relevancy, while Recall (7) is a measure of how many truly relevant results are returned.

Both precision and recall are Therefore, based on understanding and measuring relevance. These quantities are also related to the F-measure (8), which is defined as the harmonic mean of precision and recall [3].

Accuracy (9) is one metric for evaluating classification models and (Δ) is the feature value measured from the differences between Acc (TFIDF) and Acc (Wor2Vec) divides to Acc (TFIDF) (10) are calculated. The table of confusion matrix formation is shown in Table 4.

In this section, we compare manual and automatic classification results. In this comparison, we calculate precision, recall, F1, Accuracy and (Δ) for positive, negative, and neutral

Table 4 Confusion matrix

Predicted value	Actual value		
		True	False
	True	TP	FP
	false	FN	TN

classifications using precision and recall that are defined using the following formulas [44]:

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$F1 = \frac{2{*}Precion{*}Recall}{Precion + Recall} \tag{8}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

$$\Delta = \frac{Acc(TFIDF) - Acc(Wor2Vec)}{Acc(TFIDF)} \tag{10}$$

where the confusion matrix, TP (True Positive) represents the number of positive author text that are correctly predicted, whereas FP (False positive) gives the value for the number of positive author text that are predicted as negative by the classifier. Similarly, TN(True Negative) is the number of negative author text correctly predicted and FN (False Negative) is the number of negative author text predicted as positive by the classifier.

4.1 Multi class Experiments

In Tables 5 and 6 shows the evaluation matrix of several machine-learning approaches, which were applied to evaluate the proposed Arabic dataset that is discussed in the below section.

Table 5 shows the evaluation matrix of several machine-learning approaches for three classes (Positive, Negative, and Neutral). These approaches were applied to evaluate the proposed CASAD that is discussed in the above section.

Table 5 shows the evaluation matrix of several machine-learning approaches. These approaches were applied to evaluate the proposed Arabic dataset in three classes (positive, negative, and neutral) with TFIDF feature extraction had an average precision of 75% with LDA and average recall and F1 of 52% with LR and NB. And With Word2vec feature extraction, they had an average precision of 57% and average recall and F1 of 53% with LR.

Accuracy refers to the overall accuracies with the results indicate that the TFIDF occurrence with 52% gave almost the same result as between LR and NB, while Word2vec occurrence gave the highest accuracy 53%, with LR. This due to the word2vec can extract more hidden relations among the text.

4.2 Binary-Class Experiments

Table 6 shows the evaluation matrix of several machine-learning approaches that were applied to evaluate the proposed Arabic dataset in two classes (positive and negative).

Table 5 Comparison of TFIDF and Word2Vec feature extraction in three classes (Positive, Negative, and Neutral)

ML	Class	TFIDF				Wor2Vec				Δ
		Pre	Rec	F1	Acc	Pre	Rec	F1	Acc	
LR	Pos	0.62	0.40	0.49	52.381	0.65	0.39	0.49	53.333	0.018
	Neg	0.60	0.59	0.60		0.60	0.58	0.59		
	Neu	0.41	0.60	0.49		0.43	0.66	0.52		
	Avg	0.55	0.52	0.52		0.57	0.53	0.53		
LDA	Pos	1.00	0.03	0.05	38.095	0.00	0.00	0.00	33.81	– 0.112
	Neg	0.35	1.00	0.52		0.34	1.00	0.51		
	Neu	0.88	0.11	0.20		0.00	0.00	0.00		
	Avg	0.75	0.38	0.25		0.11	0.34	0.17		
KNN	Pos	0.38	0.84	0.53	40.0	0.38	0.86	0.53	39.524	– 0.012
	Neg	0.00	0.00	0.00		0.00	0.00	0.00		
	Neu	0.46	0.31	0.37		0.47	0.27	0.35		
	Avg	0.28	0.40	0.30		0.28	0.40	0.30		
CART	Pos	0.42	0.49	0.45	39.524	0.43	0.45	0.44	41.429	0.048
	Neg	0.47	0.38	0.42		0.43	0.34	0.38		
	Neu	0.29	0.29	0.29		0.39	0.45	0.42		
	Avg	0.40	0.40	0.39		0.42	0.41	0.41		
NB	Pos	0.51	0.47	0.49	52.381	0.50	0.42	0.45	49.524	– 0.055
	Neg	0.59	0.66	0.63		0.54	0.63	0.58		
	Neu	0.45	0.44	0.44		0.43	0.44	0.43		
	Avg	0.52	0.52	0.52		0.49	0.50	0.49		
SVM	Pos	0.00	0.00	0.00	29.524	0.00	0.00	0.00	29.524	0.000
	Neg	0.00	0.00	0.00		0.00	0.00	0.00		
	Neu	0.30	1.00	0.46		0.30	1.00	0.46		
	Avg	0.09	0.30	0.13		0.09	0.30	0.13		
Adapt	Pos	0.00	0.00	0.00	31.429	0.00	0.00	0.00	37.143	0.182
	Neg	0.80	0.06	0.11		0.35	0.96	0.51		
	Neu	0.30	1.00	0.46		0.62	0.16	0.26		
	Avg	0.36	0.31	0.17		0.30	0.37	0.25		
Ad_rel	Pos	0.40	0.88	0.55	42.381	0.40	0.83	0.54	42.381	0.000
	Neg	0.47	0.21	0.29		0.47	0.21	0.29		
	Neu	0.75	0.10	0.17		0.59	0.16	0.25		
	Avg	0.53	0.42	0.35		0.48	0.42	0.37		

Table 6 shows the evaluation matrix of several machine-learning approaches. With TFIDF feature extraction; there was an average precision of 74% and average recall and F1 of 71% with LR. And with Word2Vec feature extraction, there was an average precision of 73% and average recall and F1 of 70% with LR.

Table 6 Comparison between TFIDF and Word2Vec feature extraction in two classes (Positive and Negative)

ML	Class	TFIDF				Wor2Vec				Δ
		Pre	Rec	F1	Acc	Pre	Rec	F1	Acc	
LR	Pos	0.64	0.83	0.72	71.429	0.62	0.84	0.72	70.0	− 0.02
	Neg	0.81	0.62	0.71		0.82	0.58	0.68		
	Avg	0.74	0.71	0.71		0.73	0.70	0.70		
LDA	Pos	0.80	0.13	0.22	59.286	0.00	0.00	0.00	55.0	− 0.07
	Neg	0.58	0.97	0.72		0.55	1.00	0.71		
	Avg	0.68	0.59	0.50		0.30	0.55	0.39		
KNN	Pos	0.59	0.86	0.70	66.42	0.56	0.86	0.68	63.57	− 0.04
	Neg	0.81	0.51	0.62		0.80	0.45	0.58		
	Avg	0.71	0.66	0.66		0.69	0.64	0.62		
CART	Pos	0.52	0.68	0.59	57.143	0.53	0.63	0.58	58.571	0.02
	Neg	0.65	0.48	0.55		0.65	0.55	0.59		
	Avg	0.59	0.57	0.57		0.60	0.59	0.59		
NB	Pos	0.69	0.52	0.59	67.857	0.69	0.52	0.59	67.857	0.00
	Neg	0.67	0.81	0.73		0.67	0.81	0.73		
	Avg	0.68	0.68	0.67		0.68	0.68	0.67		
SVM	Pos	0.45	1.00	0.62	45.0	0.45	1.00	0.62	45.0	0.00
	Neg	0.00	0.00	0.00		0.00	0.00	0.00		
	Avg	0.20	0.45	0.28		0.20	0.45	0.28		
Adapt	Pos	0.46	0.83	0.59	48.571	0.46	0.83	0.59	47.857	− 0.01
	Neg	0.59	0.21	0.31		0.58	0.19	0.29		
	Avg	0.53	0.49	0.44		0.52	0.48	0.42		
Ad_rel	Pos	0.46	0.83	0.59	49.286	0.45	0.83	0.58	47.143	− 0.04
	Neg	0.61	0.22	0.32		0.56	0.18	0.27		
	Avg	0.54	0.49	0.45		0.51	0.47	0.41		

Accuracy refers to the overall accuracies with the results indicate that IFIDF occurrence gave the highest accuracy of 71.42% with LR, and the delta is accurate only 2.04%. However, the NB achieved the significant differences between TFIDF and Word2Vec reach of 7.8%.

On the other hand, Tables 5 and 6 shows that our approach is more significant for extracting the polarity in two classes compared with multi classes. This is due to the neutral words effect negatively in the ML training because there are several challenges to ignore the not effective neutral words from training model.

Figures 4, 5, 6 and 7 show the distribution of the evaluation values of train, develop and test are randomly distributed. To solve this problem, the 10-fold validation is used to measure the average of the results extracted from the machine learning models. However, the standard deviation of the 10-fold results reflects the model's stability. Therefore, the models (TFIDF or Word2vec) with binary or multiclass polarity are computed with a standard deviation. As a result, the variance helps to find the distribution of data in a population from a mean, and the

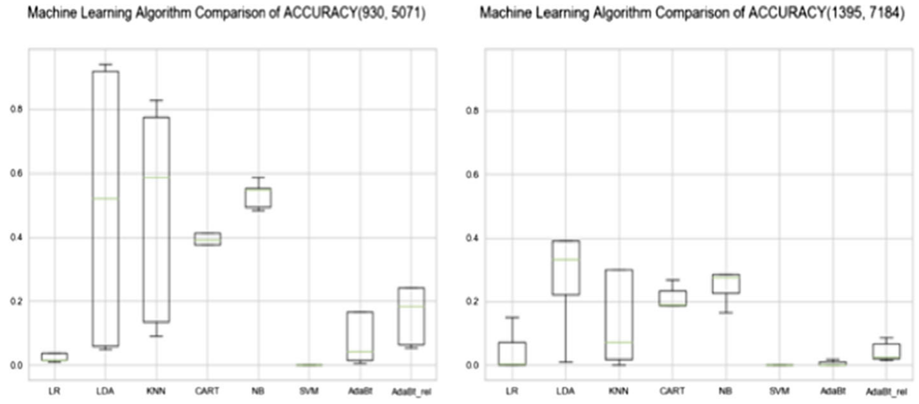


Fig. 4 Machine-learning accuracy

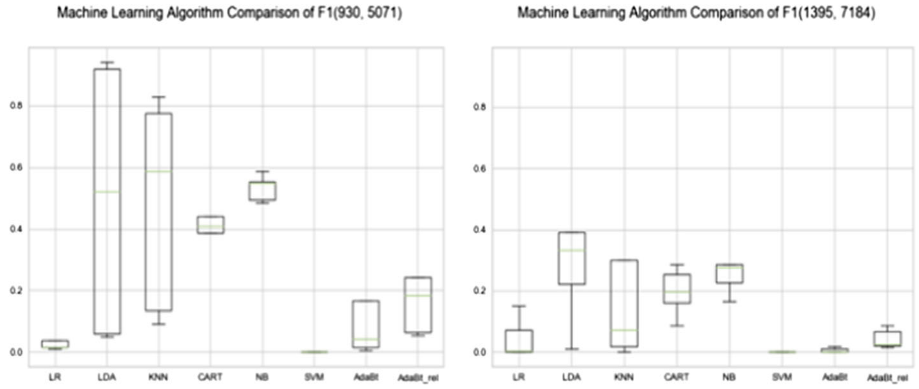


Fig. 5 Machine-learning F-score

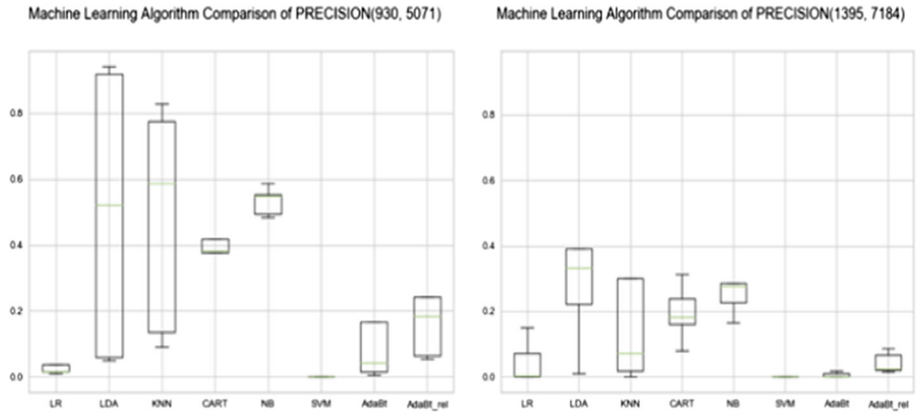


Fig. 6 Machine-learning precision

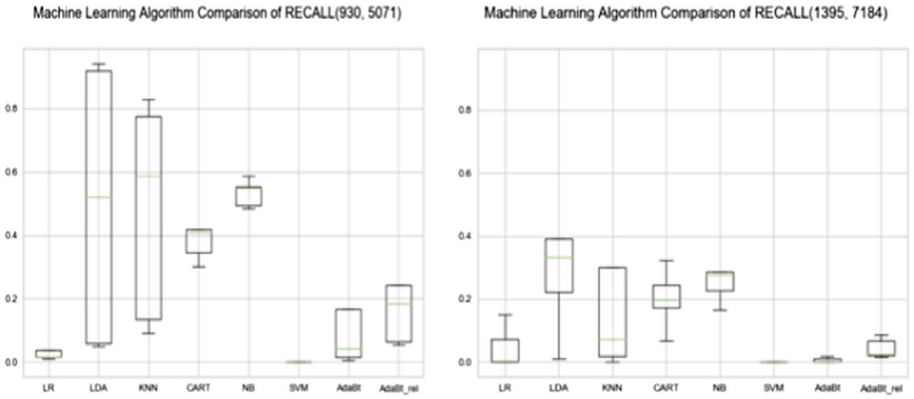


Fig. 7 Machine-learning recall

standard deviation also helps to know the distribution of data in a population. Nonetheless, the standard deviation in the figures clarifies the deviation from the mean. So the binary classification was outperformed by the multiclass classes once the 10-fold cross-validation with standard deviation was used. On the other hand, LDA achieved the best result when compared with the other machine-learning approaches for multi and binary classification.

5 Conclusions

In this work, we considered sentiment analysis of Arabic authors' books and a dataset crawled from Arabic books in all fields. A new classical Arabic dataset CASAD was created. It was statistically evaluated by an acceptable Cornbach alpha. Moreover, three and two classes feature extraction of CASAD (word2vec and TFIDF) was evaluated with six ML approaches. Although the results of the binary classes were better than those of the three classes, the accuracy of classical Arabic language needs further investigation to discover the ambiguity behind the writers' opinions. The accuracy score of Arabic text classification using the most popular ML algorithms, such as SVM, LR, and naive Bayes, was 71.42% by LR with a word2vec of binary classification (Pos and Neg). On the other hand, our approach is more important for extracting the polarity in two classes compared with multi-classes. This is due to the neutral words effect negatively in the ML training because there are several challenges to ignore the not effective neutral words from training model. Therefore, Deep Learning for CASAD classification provides a more accurate rate than standard feature extraction approaches. Therefore, this work will be addressed in the future work.

Data Availability The datasets during analyzed the current study are available from the corresponding author.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Mejova Y (2009) Sentiment analysis: an overview. University of Iowa, Computer Science Department
2. Min S, Park J (2019) Modeling narrative structure and dynamics with networks, sentiment analysis, and topic modeling. PloSone 14(12):e0226025
3. Attik M, Missen MMS, Coustaty M, Choi GS, Alotaibi FS, Akhtar N, Husnain M (2019) OpinionM-L—opinion markup language for sentiment representation. Symmetry 11(4):545
4. Chang YC, Yeh WC, Hsing YC, Wang CA (2019) Refined distributed emotion vector representation for social media sentiment analysis. Plosone 14(10):e0223317
5. Oueslati O, Cambria E, HajHmida MB, Ounelli H (2020) A review of sentiment analysis research in Arabic language. Futur Gener Comput Syst 112:408–430
6. Saxena D, Gupta S, Joseph J, Mehra R (2019) Sentiment analysis. Int J Eng Sci Math 8(3):46–51
7. Boudad N, Faizi R, Thami ROH, Chiheb R (2018) Sentiment analysis in Arabic: a review of the literature. Ain Shams Eng J 9(4):2479–2490
8. Ghallab A, Mohsen A, Ali Y (2020) Arabic sentiment analysis: a systematic literature review. Appl Comput Intell Soft Comput. https://doi.org/10.1155/2020/7403128
9. Ma Z, Nam J, Weihe, K (2016) Improve sentiment analysis of citations with author modeling. In: Proceedings of the 7th workshop on computational approaches to subjectivity, Sentiment and Social Media Analysis. pp 122–127
10. Marie-Sainte SL, Alalyani N, Alotaibi S, Ghouzali S, Abunadi I (2018) Arabic natural language processing and machine learning-based systems. IEEE Access 7:7011–7020
11. Mountassir A, Benbrahim H, Berrada I (2012) An empirical study to address the problem of unbalanced data sets in sentiment classification. In: IEEE international conference on systems. s.l. : IEEE, pp 3298–3303
12. Al-Badarneh A, Ali M, Ghaleb SM (2016) An improved classifier for arabic text. J Converg Inform Technol (JCIT) 11:69–84
13. Rushdi-Saleh M, Martín-Valdivia MT, Ureña-López LA, Perea-Ortega JM (2011) OCA: Opinion corpus for Arabic. J Am Soc Informa Sci Technol 62(10):2045–2054
14. Shahina KK, Jyothsna PV, Prabha G, Premjith B, Soman KP (2019) A sequential labelling approach for the named entity recognition in Arabic language using deep learning algorithms. In: 2019 International conference on data science and communication (IconDSC). s.l. : IEEE, pp 1–6
15. Duwairi R, Abushaqra F (2021) Syntactic-and morphology-based text augmentation framework for Arabic sentiment analysis. PeerJ Comput Sci 7:e469
16. Farha IA, Magdy W (2021) A comparative study of effective approaches for arabic sentiment analysis. Inform Process Manag 58(2):102438
17. Harrat S, Meftouh K, Smaili K (2019) Machine translation for Arabic dialects (survey). Inform Process Manag 56(2):262–273
18. Al-Azani S, El-Alfy ESM (2017) Using word embedding and ensemble learning for highly imbalanced data sentiment analysis in short arabic text. Procedia Comput Sci 109:359–366
19. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357
20. Oussous A, Benjelloun FZ, Lahcen AA, Belfkih S (2020) ASA: a framework for Arabic sentiment analysis. J Inform Sci 46(4):544–559
21. Al-Ibrahim R, Duwairi RM (2020) Neural machine translation from Jordanian Dialect to modern standard Arabic. In: 2020 11th International conference on information and communication systems (ICICS). IEEE, pp 173–178
22. Bataineh B, Duwairi R, Abdullah M (2019) ArDep: an Arabic lexicon for detecting depression. In: Proceedings of the 2019 3rd International conference on advances in artificial intelligence. pp 146–151
23. Al-Sabbagh R, Girju R (2012) Yadac: Yet another dialectal arabic corpus. In: Proceedings of the eighth international conference on language resources and evaluation (LREC'12), pp 2882-2889
24. Hadwan M, Al-Hagery M, Al-Sarem M, Saeed F (2022) Arabic sentiment analysis of users' opinions of governmental mobile applications. Comput Mater Continua 72(3):4675–4689
25. Alnawas A, Arici Nursal (2021) Effect of word embedding variable parameters on Arabic sentiment analysis performance. arXiv preprint arXiv:2101.02906.
26. Touahri I (2022) The construction of an accurate Arabic sentiment analysis system based on resources alteration and approaches comparison. Appl Comput Inform
27. Al-Ayyoub M, Khamaiseh AA, Jararweh Y, Al-Kabi MN (2019) A comprehensive survey of arabic sentiment analysis. Inform process Manag 56(2):320–342
28. Ibrahim HS, Abdou SM, Gheith M (2015) Sentiment analysis for modern standard Arabic and colloquial. arXiv preprint arXiv:1505.03105.

29. Pozzi F, Fersini E, Messina E, Liu B (2016) Sentiment analysis in social networks. Morgan Kaufmann, Burlington
30. Al-Rubaiee H, Qiu R, Li D (2016) Identifying Mubasher software products through sentiment analysis of Arabic tweets. In: 2016 International conference on industrial informatics and computer systems (CIICS). s.l. : IEEE, pp 1–6
31. Hamed AR, Qiu R, Li D (2015) Analysis of the relationship between Saudi twitter posts and the Saudi stock market. In: 2015 IEEE Seventh international conference on intelligent computing and information systems (ICICIS). s.l. : IEEE, pp 660–665
32. Alwakid G, Osman T, Hughes-Roberts T (2017) Challenges in sentiment analysis for arabic social networks. Procedia Comput Sci 117:89–100
33. Elhawary M, Elfeky M (2010) Mining Arabic business reviews. In: 2010 IEEE international conference on data mining workshops . s.l. : IEEE, pp 1108–1113
34. Aly M, Atiya A (2013) Labr: a large scale arabic book reviews dataset. In: Proceedings of the 51st annual meeting of the association for computational linguistics (Volume 2: Short Papers). vol 2, pp 494–498
35. Nabil M, Aly M, Atiya A (2014) Labr: a large scale arabic sentiment analysis benchmark. arXiv preprint arXiv:1411.6718.
36. Worsley KJ, Marrett S, Neelin P, Vandal AC, Friston KJ, Evans AC (1996) A unified statistical approach for determining significant signals in images of cerebral activation. Hum Brain Mapp 4(1):58–73
37. Alksher MA, Azman A, Yaakob R, Kadir RA, Mohamed A, Alshari E (2017) A framework for idea mining evaluation. In: New trends in intelligent software methodologies, tools and techniques. IOS Press, pp 550–559
38. Alnawas A, Arici N (2018) The corpus based approach to sentiment analysis in modern standard Arabic and Arabic dialects: a literature review. Politeknik Dergisi 21(2):461–470
39. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv:1301.3781v3. In: 2013 Proceedings of the international conference on learning representations (ICLR 2013), pp 1–12. ISSN (15324435) ISBN (1532–4435).
40. Alshari EM, Azman A, Doraisamy S, Mustapha N, Alkeshr M (2017) Improvement of sentiment analysis based on clustering of Word2Vec features. In: 2017 28th international workshop on database and expert systems applications (DEXA). IEEE, pp 123–126
41. Rong X (2014) word2vec parameter learning explained. arXiv preprint arXiv:1411.2738.
42. Mikolov T, Joulin A, Chopra S, Mathieu M, Ranzato MA (2014) Learning longer memory in recurrent neural networks. http://arxiv.org/abs/1412.7753
43. Guo S, Chen R, Li H (2017) Using knowledge transfer and rough set to predict the severity of Android test reports via text mining. Symmetry 9(8):161
44. Li N, Wu DD (2010) Using text mining and sentiment analysis for online forums hotspot detection and forecast. Decis Supp Syst 48(2):354–368

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.