Solving the twitter sentiment analysis problem based on a machine learning-based approach

Article *in* Evolutionary Intelligence · September 2020

DOI: 10.1007/s12065-019-00301-x

Fatemeh Zarisfi Kermani
Shahid Bahonar University of Kerman
5 PUBLICATIONS **30** CITATIONS

Faramarz Sadeghi
Shahid Bahonar University of Kerman
7 PUBLICATIONS **32** CITATIONS

Evolutionary Intelligence
https://doi.org/10.1007/s12065-019-00301-x

RESEARCH PAPER

Solving the twitter sentiment analysis problem based on a machine learning-based approach

Fatemeh Zarisfi Kermani[1] · Faramarz Sadeghi[1] · Esfandiar Eslami[2]

Received: 9 January 2019 / Revised: 29 July 2019 / Accepted: 9 October 2019

Abstract

Twitter Sentiment Analysis (TSA) as part of a text classification task has been widely attended by researchers in recent years. This paper presents a machine learning approach to solving the TSA problem in three phases. In the second phase, a suitable value for representing each feature in the Vector Space Model is determined through the weighted combination of the values obtained from four methods (i.e., Term Frequency and Inverse Document Frequency, semantic similarity, sentiment scoring using SentiWordNet, and sentiment scoring based on the class of tweets). In this manner, finding the percentage of contributions or weights of each method is defined as an optimization problem and solved using a genetic algorithm. Also, the weighted values obtained from four methods are combined based on the Einstein sum as an important T-conorm method. Finally, the performance of the proposed method is tested based on the accuracy of support vector machine and multinomial naïve Bayes classification algorithms on four famous Twitter datasets, namely the Stanford testing dataset, STS-Gold dataset, Obama-McCain Debate dataset, and Strict Obama-McCain Debate dataset. The obtained results show the high superiority of the proposed method in comparison with the other methods.

Keywords Twitter sentiment analysis · Genetic algorithm · Einstein T-conorm · Support vector machine · Multinomial Naïve Bayes

1 Introduction

In recent years, the web has been known as a place that suitably allows users to communicate and share information with each other regardless of distance in the real world [1]. One web-based technology that converts communication between users into an interactive dialogue is social media [2]. The growth of social media and micro-blogging platforms like Facebook, Twitter, and YouTube, expressing the feelings, opinions, and thoughts of users about various topics is easy [3].

Twitter is the most popular microblogging service for Internet users [3], averaging 326 million active users[1] monthly who send around 500 million tweets per day and around 200 billion tweets per year.[2] Twitter registers users to read and post short messages, so-called tweets, about various topics such as events, movies, products, politicians, governmental organizations, and so on [4]. Retrieving and processing this huge amount of exchanged information about a special topic can help in answering technological and sociological questions [5]. In other words, opinions expressed on Twitter are essential information that represents the sentiment of people and requires analysis. For this reason, these posts are applied to measure the general sentiment of users by a field of data mining, called Sentiment Analysis (SA).

The task of SA as a part of text classification, the main issue in text mining, is the classification of opinions or sentiments of people about different topics expressed in texts. SA can identify the interest of people in a specific topic by

✉ Fatemeh Zarisfi Kermani
fzarisfi@math.uk.ac.ir

Faramarz Sadeghi
farsad@uk.ac.ir

Esfandiar Eslami
esfandiar.eslami@uk.ac.ir

[1] Department of Computer Science, Faculty of Mathematics and Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran

[2] Department of Pure Mathematics, Faculty of Mathematics and Computer Science, Shahid Bahonar University of Kerman, Kerman, Iran

[1] https://zephoria.com/twitter-statistics-top-ten/.

[2] https://www.internetlivestats.com/twitter-statistics/.

Published online: 22 October 2019

determining the positive or negative polarity of their emotions [6]. Twitter Sentiment Analysis (TSA) can be a challenging task, because each tweet, with the maximum 140 characters, is defined by a combination of slang, abbreviations, acronyms, emoticons, URLs, and hashtags [7].

Based on [8], the approaches that are most commonly used in SA can be grouped into two categories with multi-different subcategories. The first group, the machine learning-based approaches with two different branches of supervised and unsupervised learning, requires a set of linguistic or nonlinguistic features to formulate SA. In recent years, various methods have been published. For example, in [7], a TSA method was proposed in which its authors extract features using a unigram, and they are handled with all of the tweets as bags of words. A fuzzy thesaurus to represent each feature using sentiment similarity has also been produced based on the concept of information retrieval. Finally, these features are used to train three known classifier algorithms, namely Support Vector Machines (SVM), Multinomial Naïve Bayes (MNB), and Bernoulli Naïve Bayes (BNB). The authors of the method proposed in [9] converted tweets into bags of words and represented each of them using a sentiment scoring module based on SWN lexicons. Then, each of the represented tweets is analyzed using rough set theory (RST)-based algorithms for rule induction. In [10], a method has been suggested with which a new feature set is extracted based on the combination of bigram, information gain, and object-orientated words, and each is represented based on binary occurrences for training two classification algorithms (namely SVM and naïve Bayes). The method proposed in [11] focuses on extracting a set of 11 different features from preprocessed tweets and clusters each of tweets using the k-means algorithm. The cuckoo search is utilized as an optimization algorithm for finding the suitable centers of clusters. The method proposed in [12], is based on the label propagation on a graph constructed using users, tweets, word unigrams, word bigrams, and hashtags.

The second group comprises lexicon-based approaches. They require sentiment lexicons which include the sentiment words together with their sentiment scores to determine the sentiment polarity of a text. Applying one of the lexicon-based approaches for Twitter sentiment analysis needs a suitable sentiment lexicon. In [13], a method is proposed that has three various modules, each of which is implemented using WordNet, SentiWordNet (SWN), or emoticon lexicons. In [14], a method is proposed that is used at sentence and document levels. By considering the different parts of a tweet, like slangs, emoticons, the existent sentiment words in SWM, and other non-existent words in SWN, this method improves the performance of the TSA system. In fact, by calculating the sentiment polarity of each part based on the predefined sentiment lexicons or based on the extracted information from the tweets, the sentiment orientation of each tweet is determined. The method proposed in [15], because it constructs a novel lexicon, namely, SentiCircle, is known as a lexicon-based approach. In SentiCircle, the sentiment polarities of a word are fixed and constructed based on the contextual semantics of the word. Another suggested method that uses SWN lexicons to produce a novel sentiment lexicon is named SentiMI [16]. In this method, SWN is utilized as a training corpus, and a SentiMI lexicon is constructed by calculating the mutual information based on POS tagging for each term extracted from SWN. The results of these studies show that the assignment of a suitable sentiment value to each term in the predefined lexicons can greatly affect the performance of lexicon-based approaches. For this reason, in [6], determining these suitable values has been defined as an optimization problem, and the Genetic Algorithm (GA) has been used to solve it.

In general, in the second group, the predefined sentiment lexicons (such as SentiWordNet (SWN) [17], AFINN [18], and SO-CAL [19] are domain-independent; building a comprehensive and domain-dependent lexicon is not only expensive, but also very time-consuming. Using the first group, the SA is performed by explaining the sentiment of the context of the tweets instead of only counting the sentiment words in the second group [7]. Therefore, this paper focuses on presenting a new method in the first group.

In machine learning-based approaches, tweets must be properly described in order to achieve an acceptable accuracy. Therefore, the critical steps are extracting features and determining an appropriate weight for them. Feature engineering is known as an important phase in these methods. In this paper, the features of a set of tweets are extracted based on the bag-of-words (BOW) model, and a novel hybrid method based on a machine learning algorithm is proposed to represent the extracted features more suitably than previous methods. This method was inspired by four previous methods, namely the semantic similarity method [7], the Term Frequency-Inverse Document Frequency method (TFIDF), the method of sentiment scoring using SWN [9], and the method of sentiment scoring based on the class of tweets in the corpus [14]. Each of these methods is known in the field of text classification especially SA, and has advantages and disadvantages that are mentioned below:

- In the first examined method (the semantic similarity method), the author has represented each of the extracted features based on a custom fuzzy thesaurus that incorporates the semantic similarity of every feature with every word in a tweet. Therefore, the sentiment polarity of the features is not considered in any representation. Nevertheless, this proposed method can confront the sarcastic expressions in tweets. In addition, this method can produce a less sparse representation matrix.

- In the second examined method (TFIDF), each extracted feature is represented based on the importance of the feature in a given tweet rather than the feature frequency. For features that are not in the given tweet or features that appear in all tweets, this method offers zero value. Therefore, it suffers from the dispersion of the representation matrix. In addition, the sentiment polarity of the features is not considered in any representation. Nevertheless, this method can detect the relevance features of each tweet that occur not only many times in that tweet, but also within a small number of tweets.
- In the third examined method (sentiment scoring using SWN), only the available features in the SentiWordNet (SWN) lexicon are recognized as words with sentiment polarity that have a non-zero value in the representation matrix. In fact, the authors have utilized the sentiment polarity of extracted features independently from the content of the tweets to construct the representation matrix for the corpus of tweets in different subjects. Thus, it is possible for a feature to have a negative score based on the content, whereas according to the SWN lexicon, a positive score has been assigned to it. In addition, it is possible for some features to be considered in the sentiment analysis of a given tweet, while they have not been mentioned in the SWN lexicon. Also, the constructed representation matrix will be as a sparse matrix. Nevertheless, this method well highlights the features with pure emotional tendencies within each tweet. In addition, this method well distinguishes between the emotional tendencies of each word in the various POS tags.
- In the fourth examined method (sentiment scoring based on the class of tweets in the corpus), the authors have used the information extracted from the corpus of tweets to assign a sentiment score to each feature. In general, the frequency-based probabilities of a feature in positive and negative tweets and the importance of that feature in a given tweet are the information used in computing the sentiment score of the feature. Since the importance of features in each tweet has been calculated based on the TFIDF method, the constructed representation matrix will be a sparse matrix. In addition, the general sentiment polarity of each feature is not considered independently of the context. Nevertheless, in this proposed method, the sentiment scores of features are estimated independently of a predefined sentiment lexicon. As a result, a sentiment score is dependent on the content of tweets, and all features in a given tweet have a sentiment score.

Clearly, each of the constructed matrices discussed above has advantages and disadvantages that can affect the accuracy and performance of the used classifier. It seems that combining all matrices could compound the advantages of all four methods and cover their disadvantages. For this reason, the main goal of this paper is to construct a representation matrix based on the hybrid of all four matrices. The four matrices are aggregated using a fusion operator (Einstein T-conorm). Because each method has certain benefits, considering the same contribution for all of them in constructing the proposed matrix cannot be a good idea; it is better to specify a weight for each method. Here, determining a suitable weight can be defined as an optimization problem, and the GA as a global search method can solve it. In GA, the efficiency of each proposed solution is evaluated according to a fitness function. In this paper, a set of selected weights for each matrix is defined as a solution of GA. Also, the accuracy of a specific classification algorithm, like MNB or SVM, is considered as the fitness function of GA.

It has been claimed that the performance of the proposed method is better than that of other previous methods. To prove this claim, the performance of the proposed method was evaluated on four benchmark collections in the field of TSA (the Stanford testing dataset STS-Gold dataset, Obama-McCain debate (OMD), and strict OMD datasets). The results showed that the accuracy and F-measure of the proposed hybrid method can be significantly better than that of state-of-the-art methods.

The main contributions of this research are as follows:

- The extracted features of tweets are represented in the form of a matrix constructed using the proposed hybrid method based on the Einstein T-conorm.
- Because the participation percentage of each of the four matrices in making the final representation matrix is different, the problem of determining these percentages is defined as an optimization problem and solved using GA.
- Applying a specific classification algorithm as a black box in the proposed hybrid method is its major specification.
- Analysis showed that the proposed method has a better accuracy value compared with other methods.

The rest of this paper is organized as follows: A brief summary of background information and the requirements of the proposed method are described in Sect. 2. The proposed method is discussed in Sect. 3, and the experimental results are shown in Sect. 4. Finally, the paper's conclusion along with suggestions for future works are mentioned in Sect. 5.

2 Preliminaries

In this section, the requirements of the proposed method are briefly described.

2.1 The four previous methods for feature representation

Here, the four previous methods for feature representation, namely semantic similarity [7], TFIDF, sentiment scoring using SWN [9] and, sentiment scoring based on the class of tweets in the corpus [14] are briefly described.

2.1.1 Semantic similarity

Representation of features based on semantic similarity was proposed by Ismail et al. [7]. They utilized Eq. (1) to represent the extracted features in the form of a matrix that illustrates the semantic similarity between each feature and the whole tweets as follows:

$$\mu_{F_i d} = 1 - \prod (1 - Cf_{ij}) \tag{1}$$

where $\mu_{F_i d}$ is the semantic similarity between a feature F_i and a given tweet d. Cf_{ij} is the semantic similarity between the feature F_i and each available word W_j in tweet d that is calculated according to the custom fuzzy thesaurus.

Ismail et al. constructed the custom fuzzy thesaurus using Eqs. (2) (3), and (4). In fact, they computed the semantic similarity between every two distinct words in the Twitters by using the suggested method in [7].

$$C_{ij} = \sum_{x \in V(w_i)} \sum_{y \in V(w_j)} \frac{1}{d(x, y)} \tag{2}$$

$$nC_{ij} = \frac{C_{ij}}{\left|V\left(w_i\right)\right| * \left|V\left(w_j\right)\right|} \tag{3}$$

$$Cf_{ij} = \frac{\sum_{m=1}^{k} nC_{ij}}{k} \tag{4}$$

In Eq. (2), $d(x, y) = |postiion(x) - position(y)| + 1$ is known as a distance correlation factor in the field of information retrieval. This distance factor must be computed for all positions of the two words w_i and w_j that are listed in two sets, namely $V\left(w_i\right)$ and $V\left(w_j\right)$, respectively. After these computations, the frequency of co-occurrence of two specified words in the single tweet C_{ij} is obtained. Finally, Cf_{ij} (as the semantic similarity between two words) is achieved by dividing the sum of normalized values of C_{ij} (i.e., nC_{ij}) into k (as the number of common tweets between two words w_i and w_j).

2.1.2 Term Frequency-Inverse Document Frequency (TF-IDF)

In many fields of natural language processing, such as text classification, in which feature extraction is based on BOW, the TFIDF method is one of the most popular and simplest methods for representation of the extracted features. Because TSA is part of text classification, TFIDF is used as a common feature representation method and is calculated as such:

$$w_{ij} = tf_{ij} * idf_j \tag{5}$$

$$idf_j = \log \frac{|D|}{n_{j,D} + \xi} \tag{6}$$

where $|D|$ is the total number of tweets in the corpus D, $n_{j,D}$ is the number of tweets in D in which term j occurs, and ξ is a fixed value to solve the division-by-zero problem when term j does not occur in the corpus D (here, $\xi = 1$). The parameter tf_{ij} represents the number of occurrences of term j in the ith tweet.

2.1.3 Sentiment scoring using SWN

In the sentiment scoring module of the method proposed by Asghar et al. in [9], a sentiment score is assigned to every word using SentiWordNet (SWN). This well-known sentiment lexicon has more than 60,000 synsets obtained from WordNet. SentiWordNet (SWN) by $senti^{+}$, $senti^{-}$, and $senti^{0}$ icons appoints three scores (positive, negative, and neutral/objective) in the range of 0.0–1.0 to each synset of every word.

Every word has various meanings in each of its Part-Of-Speech (POS) tags, so the correct sentiment scores of each are obtained using the computed average values as follows:

$$senti\_ave^{+}(w, P) = \sum_{i=1}^{numSyn} senti^{+}(w_i)/numSyn \tag{7}$$

$$senti\_ave^{-}(w, P) = \sum_{i=1}^{numSyn} senti^{-}(w_i)/numSyn \tag{8}$$

$$senti\_ave^{0}(w, P) = \sum_{i=1}^{numSyn} senti^{0}(w_i)/numSyn \tag{9}$$

where $senti\_ave^{+}$, $senti\_ave^{-}$, and $senti\_ave^{0}$ are the average values of the sentiment scores of the $numSyn$ different meanings of a word w in the POS tag P. After this computation, every word has three scores in each sentiment polarity. Finally, to make an exact diagnosis of the sentiment polarity for a special word in a particular POS category, the following is done:

$$Senti^{SWN}(w,P) = \left\{ \begin{array}{ll} senti\_ave^{0}(w,P) & if senti\_ave^{0}(w,P) = 1 \\ \max(senti\_ave^{+}(w,P) senti\_ave^{-}(w,P)) & else \end{array} \right\} \tag{10}$$

2.1.4 Sentiment scoring based on the class of tweet

Applying an enhancement of the delta scoring technique that is available in [20] for assigning a sentiment score to every extracted word is part of the modules that are proposed by Asghar et al. [14]. In this module, the sentiment score of a word is computed without a common sentiment lexicon such as SWN and using the information extracted from the corpus. This information is formulated as below:

$$Senti(w) = \left\{ \begin{array}{ll} tf * idf * P(w,T_{+}) & P(w,T_{+}) > P(w,T_{\_}) \\ tf * idf * (w,T_{\_}) & P(w,T_{\_}) > P(w,T_{+}) \end{array} \right\} \tag{11}$$

where *tf* and *idf* as term frequency and inverse document frequency are obtained using Eq. (5) and Eq. (6), respectively. Also, $P(w,T_{+})$ and $P(w,T_{\_})$ as the frequency-based probabilities are the probabilities of word w occurring in positive and negative tweets of the training set that are computed using Eq. (12) and Eq. (13), respectively:

$$P(w,T_{+}) = \frac{count(w \in T_{+})}{|T_{+}|} \tag{12}$$

$$P(w \in T_{\_}) = \frac{count(w \in T_{\_})}{|T_{\_}|} \tag{13}$$

where T_{+} and $T_{\_}$ are positive and negative tweets in the corpus, respectively. Thus, $|T_{+}|$ and $|T_{\_}|$ are the total number of positive and negative tweets, respectively.

2.2 Einstein T-conorm

Information fusion is a very important problem which can be solved with difficulty when multiple sources of information suffer from uncertainty. One approach is to represent this uncertainty as fuzzy sets. The fuzzy set operators that are taken from fuzzy logic can be used to perform the information fusion [21]. These operators, such as conjunction and disjunction, can be applied directly on the fuzzy sets. The family of T-norms and T-conorms are respectively determined as the generalization of the conjunctive "AND" operator and the disjunctive "OR" operator of the Boolean logic connectives [22]. T-conorms (also called S-norms) are utilized to present the union in fuzzy set theory. The Einstein sum, an example of a T-conorm is defined as a function $S = [0,1] \times [0,1] \rightarrow [0,1]$ such that:

(Definition 1) [23]

$$S(x,y) = \frac{x+y}{1+x \cdot y} \quad \forall (x,y) \in [0,1] \tag{14}$$

2.3 Genetic algorithm

The base idea of GA as one of the evolutionary algorithms was first introduced by John Holland in 1960 based on the concept of Darwin's theory of evolution and natural selection; Goldberg extended GA in 1989 [24]. This global search method can search in a large-scale, complex search space without any knowledge about it. Thus, it is known as a suitable method for solving optimization problems. This iterative algorithm starts with a population of chromosomes, each called an individual, represented as a candidate solution in the search space that has been randomly generated. By applying a set of operators to the individuals in each iteration, a new population is created. The selection, crossover, and mutation have been defined as three main operators for constructing new chromosomes. First, the selection operator is executed and two chromosomes (as the parents) are selected based on a fitness function. Second, the crossover operator is applied in order to construct two offspring that inherit some characteristics of their parents. Third, the mutation operator is performed that can change some characteristics of each offspring. It is expected that the new population in the current iteration will suggest better solutions than the previous population in the former iteration. This procedure is performed until one of the stop criteria like, a fixed number of iterations reached, is met. In each iteration, the quality of the produced candidate solution via each individual is measured by a fitness function. Figure 1 shows the flowchart of the GA algorithm.

3 Proposed method

The aim of this paper is to provide a new method based on machine learning for solving the TSA problem. The new method is divided into three major phases: (1) data pre-processing, (2) feature engineering (including feature extraction and feature representation), and (3) classification. In machine learning-based approaches, the proper engineering of features can clearly affect the performance of the classification algorithm used; thus, special attention is given to improving the second phase. In this section, a novel hybrid method is proposed to represent each feature extracted from tweets. The proposed method is a combination of results

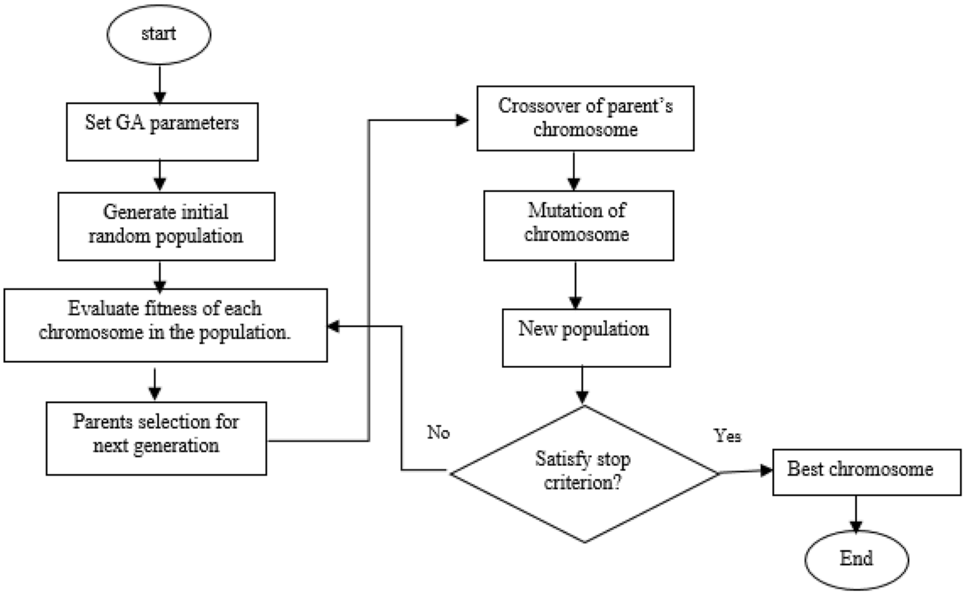

start
Set GA parameters
Generate initial random population
Evaluate fitness of each chromosome in the population.
Parents selection for next generation
Crossover of parent's chromosome
Mutation of chromosome
New population
Satisfy stop criterion?
No
Yes
Best chromosome
End

Fig. 1 The flowchart of the Genetic Algorithm

obtained from the four previous methods. In this paper, the union operator in fuzzy set theory and GA are utilized to achieve an effective combination. The details of the proposed method are presented in the form of a general scheme in Fig. 2. In the rest of this section, the details of each of the mentioned cases are discussed.

3.1 Data pre-processing

Applying structures such as abbreviations, slangs, punctuation signs, words with repeated letters and spelling errors, and some twitter specific vocabulary like URLs, hashtags, and emoticons are usually common among Twitter users. As a result, from the point of view of TSA researchers, the pre-processing phase plays a bold role in the extraction of suitable and relevant features and final correct classification of tweets [5, 25]. In the proposed method, the three steps of filtering, replacing, and modifying are used to implement the pre-processing phase.

3.1.1 Filtering

In this step, all tokens that are not useful for our application (i.e., has no information about the sentiment of the tweets) are known as the noise elements and filtered through the regular expression used in [11]. These useless tokens are *URLs*, *usernames* (i.e., a phrase, including a name of a person or organization that starts with the '@' symbol), *redundant punctuation marks,* and *whitespaces*.

3.1.2 Replacing

The use of special structures in tweets is not only very usual among users, but also has a remarkable impact on indicating the sentiment of the tweets. In this step, first all uncommon structures are detected using the reformed regular expressions adopted from Christopher Potts' tokenizing script.[3] Then the various standard dictionaries are used to replace them with legal phrases. A list of the special structures that must be replaced mentioned in [25] includes *hashtags* (i.e., a word or phrase without space in tweets that starts with the '#' symbol), *emoticons*[4] (i.e., pictorial expressions including punctuation marks, numbers, and letters), *slangs,*[5] *acronyms,*[6] *elongated words* (i.e., a word with a sequence of letters repeated more than twice), and *contractions* (i.e., contracted form of some phrases like "don't", "won't", etc.).

3.1.3 Modifying

The existence of many misspelled words in tweets, as well as the usage of different word forms, can have an unpleasant effect on the accuracy of sentiment classification. For this reason, they are modified by *spell checking* and *lemmatization*, respectively.

[3] http://sentiment.christopherpotts.net/tokenizing.html.

[4] https://en.wikipedia.org/wiki/List_of_emoticons.

[5] https://www.noslang.com/dictionary.

[6] https://www.netlingo.com/acronyms.php.

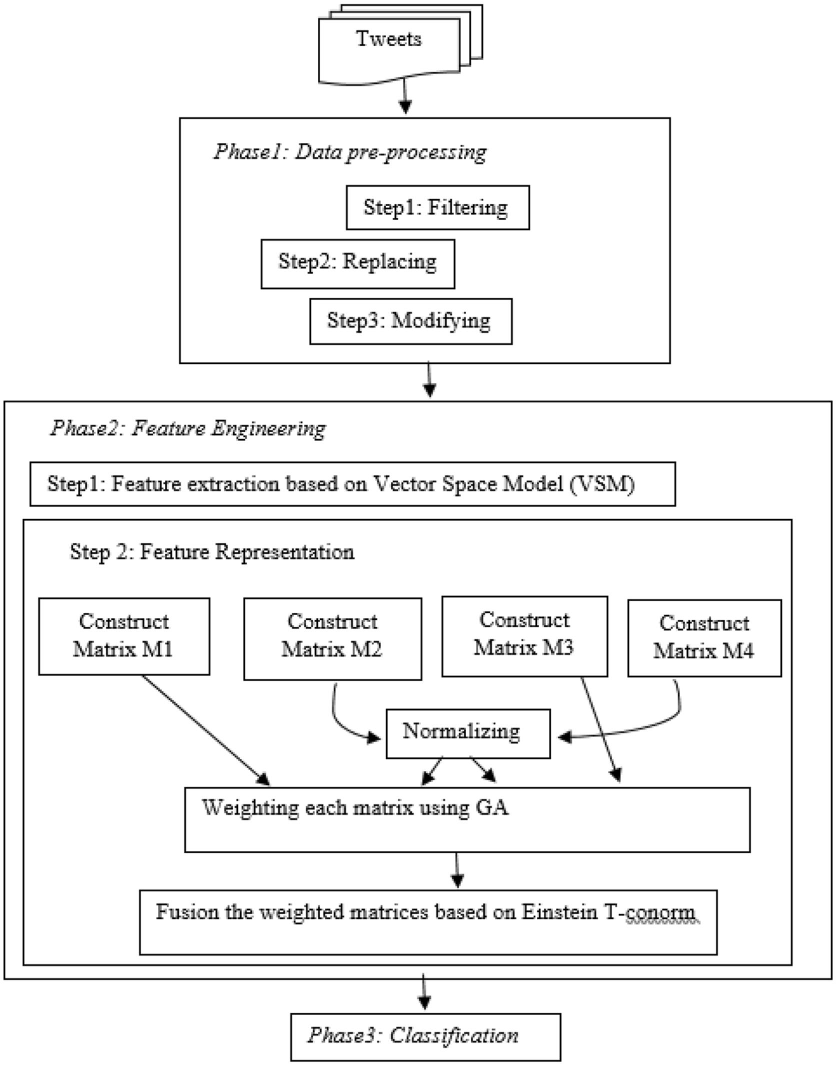

Tweets
Phase1: Data pre-processing
Step1: Filtering
Step2: Replacing
Step3: Modifying
Phase2: Feature Engineering
Step1: Feature extraction based on Vector Space Model (VSM)
Step 2: Feature Representation
Construct Matrix M1
Construct Matrix M2
Construct Matrix M3
Construct Matrix M4
Normalizing
Weighting each matrix using GA
Fusion the weighted matrices based on Einstein T-conorm
Phase3: Classification

Fig. 2 the general scheme of the proposed method

3.2 Feature engineering based on the proposed hybrid method

Feature engineering includes two steps, namely feature extraction and feature representation. In the first step, the Vector Space Model (VSM) [26] is used, and every single word in the tweets is extracted as a feature. By ignoring the grammar structures and word order, the corpus of the tweets is converted into a bag/set of individual words, called BOW. In the second step, every feature must have a representation value in all tweets. In this paper, this step is carried out according to a novel hybrid method that has four stages: *constructing*, n*ormalizing*, w*eighting*, and *combining*.

- Stage 1: Constructing

The four methods described in Sect. 2 are considered. Then each of them is individually implemented on a given dataset of tweets. As a result, the four separate matrices are constructed with a different representative value for each feature.

- Stage 2: Normalizing

The range of representative values for each feature is different in the four matrices, thus normalizing them on a common scale is attempted. The values in the matrix obtained from the semantic similarity method [7] are set into a range of 0 and 1. The reason for this event is the existence of a normalization step embedded in the method. Also, the nature of the used SWN lexicon in the method of sentiment scoring using SWN [9] has caused the values in its matrix to be set into the same range of 0 and 1. The ranges of the two other matrices obtained from the TFIDF method and the method of sentiment scoring based on the class of tweets in the corpus [14], however, are different. For this reason, the Min–Max normalization technique is used to change them into the range of 0 and 1. Min–Max normalization is known

Table 1 The setting of the GA parameters

Parameter	Value	Description
Population size	100	–
Length of each chromosome (i.e., the number of genes)	4	Equal with the number of available matrices
Max. iteration	150	–
λ_1, λ_2	0.5	Two coefficient of the average crossover.
p_c^{low}	0.6	The lower bound of the crossover rate
p_c^{high}	0.9	The upper bound of the crossover rate
p_m^{low}	0.005	The lower bound of the mutation rate
p_m^{high}	0.05	The upper bound of the mutation rate

as a linear transformation performed on the original data. Suppose that $\min_A$ and $\max_A$ are the minimum and maximum values of a feature called *A*. This technique converts the value *v* of feature *A* to value *v'* in the range [0,1], according to the following formula [27]:

$$v' = \frac{v - \min_A}{\max_A - \min_A} \tag{15}$$

- Stage 3: Weighting

According to the flowchart of GA in Fig. 1, the specific parameters of GA are first set based on the values mentioned in Table 1. Then, in the first iteration of GA, a population of individuals is randomly initialized. Every gene of the individual represents the participation weight of its corresponding matrix. Therefore, a random real value in the interval 0 and 1 is assigned to each of the genes, so that the sum of all genes is one. Utilizing the combination stage upon each individual, a representation matrix can be made. As a result, the efficiency of the constructed matrices is considered as the fitness value of the individuals. Because this efficiency can be seen in the performance of the used classifier algorithm, the accuracy is utilized as a good criterion to evaluate the performance. This measure is defined as the total number of correctly classified examples over the total number of examples [28].

Twitter sentiment analysis is known as a binary classification problem. Based on the two classes, positive and negative, a confusion matrix is obtained that it divides every classified tweet into four categories [27].

Based on the confusion matrix shown in Table 2, for a tweet β, TP increases 1 when β is predicted as class positive and if the actual class of β is Positive; if the true class label of β is negative, FP increases 1. Also, TN increases 1 when β is predicted as class negative and if the actual class of β is negative; if the true class label of β is positive, FN increases 1. Therefore, the accuracy measure is calculated as follows:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} * 100 \tag{16}$$

After measuring the quality of the individuals, a selection operator called the roulette wheel is used to choose two individuals as parents based on the fitness function. Then, to produce two children from them, a special case of the convex crossover, called an average crossover, is executed [29]. In the average crossover, each child is the weighted average of the parents and is calculated as follows:

$$child_1 = \lambda_1 * parent_1 + \lambda_2 * parent_2 \tag{17}$$

$$child_2 = \lambda_2 * parent_1 + \lambda_1 * parent_2 \tag{18}$$

where $child_1$ and $child_2$ are children obtained from two parents. Also, λ_1 and λ_2 are two coefficients with the restrictions $\lambda_1 + \lambda_2 = 1, \lambda_1 > 0, \lambda_2 > 0_2$. Finally, the exchange mutation operator is performed upon some of the children. The exchange operator is executed in such a way that two genes of the selected child are randomly chosen and their values are exchanged. As a result, the constraint of total 1 for each individual of the population will be preserved. In this paper, the maximum number of iterations is considered as a stop criterion. This operation is repeated until the stop criterion is achieved. Table 3 shows the pseudo-code of using GA in the proposed method.

Note that in this paper, neither the crossover rate nor the mutation rate is fixed in all iterations. Here, two linear equations are used to determine both the crossover rate and the mutation rate in each iteration. These equations that are dependent on iteration are defined as follows:

$$P_m(iter) = P_m^{high} - \left(P_m^{high} - P_m^{low}\right)^{iter}/_{Max\_iter} \tag{19}$$

$$P_c(iter) = P_c^{high} - \left(P_c^{high} - P_c^{low}\right)^{iter}/_{Max\_iter} \tag{20}$$

Table 2 Confusion matrix

	Actual positive	Actual negative
Real positive	True positive (TP)	False positive (FP)
Real negative	False negative (FN)	False negative (FN)

Table 3 Pseudo-code of using GA in the proposed method

```
Algorithm 1 pseudo-code of using GA in the proposed method.
Begin
Divide dataset into a training set and testing set using 10-fold cross-validation.
Initialize the population of $N$ agent (randomly).
For $i$=1 to a maximum number of iterations ($IT$):
    For j=1 to population size ($N$):
        For $k$=1 to a total number of tweets in the training set ($T$):
            For $l$=1 to a total number of extracted features ($F$):
                Compute Einstein T-conorm.
            End.
        End.
        Train the classification algorithm and compute the accuracy value.
        Find the best solution (i.e., a set of weights with high accuracy).
    End.
    Select two parents in the population using the roulette wheel operation.
    Generate two offspring using the average crossover operation between two parents.
    Mutate some offspring using the exchange mutation operator.
End.
Return the best set of weights.
```

where $P_m(iter)$ and $P_c(iter)$ are the mutation rate and crossover rate in *iter* iteration, respectively. According to these equations, in the primitive iterations of the algorithm where the necessity of high exploration is felt, both the crossover rate and the mutation rate are tuned in the highest value. In return, both rates reach the lowest possible value in the final iterations of the algorithm, where the necessity of high exploitation is felt.

- Stage 4: Combination

In each of the four normalized matrices, a real value has been assigned to a feature that can be considered as the degree of membership of that feature in each of the tweets. These assigned values are different in each matrix. Thus, for every feature, a set of degrees of membership $\{m_1(d,f), m_2(d,f), m_3(d,f), m_4(d,f)\}$ is available. It means that the value appropriated into every feature in the proposed matrix (i.e., $M(d,f)$) must be produced through the combination of all of them. From the mathematical point of view, this problem means finding a proper mapping $g : [0,1]^4 \rightarrow [0,1]$ such that:

$$M(d,f) = g(m_1(d,f), m_2(d,f), m_3(d,f), m_4(d,f)) \quad (21)$$

In this paper, the Einstein T-conorm, mentioned in Sect. 2, is considered in order to perform this combination. Therefore, the value of feature f in tweet d of the proposed matrix M (i.e., $M(d,f)$) is computed based on the following equations:

$$M(d,f) = S(S_1(m_1(d,f), m_2(d,f)), S_2(m_3(d,f), m_4(d,f))) \quad (22)$$

$$S_1(m_1(d,f), m_2(d,f)) = \frac{w_1 * m_1(d,f) + w_2 * m_2(d,f}{1 + (w_1 * m_1(d,f) * w_2 * m_2(d,f))} \quad (23)$$

$$S_2(m_3(d,f), m_4(d,f)) = \frac{w_3 * m_3(d,f) + w_4 * m_4(d,f}{1 + (w_3 * m_3(d,f) * w_4 * m_4(d,f))} \quad (24)$$

where $m_i(d,f)$ is known as the assigned degree of membership of the ith *matrix* into feature f in tweet d for i = 1,2,3,4, and w_i is the participation weight of ith *matrix* in the construction of the proposed matrix M specified in the weighting stage.

Table 4 Datasets

Dataset	Total number of tweets	Number of positive tweets	Number of negative tweets
Stanford twitter sentiment (STS) testing set	359	182	177
STS-Gold	2032	632	1400
Obama-McCain Debate (OMD)	1906	710	1196
Strict Obama McCain Debate (SOMD)	916	347	569

3.3 Classification

After constructing the feature vectors of tweets and representing each of them appropriately, two classification algorithms, SVM and MNB are applied to determine the sentiment orientation of each tweet (positive or negative)

SVM as a supervised learning algorithm is widely used in both classification and regression tasks. This method was first proposed in 1995 by Vapnik [30] and can be applied for classifying not only linear, but also non-linear datasets with high dimensions. Since TSA as a part of text classification includes the feature space with high dimension and linearly separable characteristics, SVM is a recommended classification algorithm for it.

MNB as a type of naive Bayes classifier is a probabilistic classification model based on the Bayes' rule with the naïve independence assumption between the features. Since MNB is based on multinomial distribution, it can be used well for data with discrete and easily countable data. For this reason, MNB is a proper classification algorithm for text classification tasks like TSA.

Table 5 the size of the feature vector of each dataset

Datasets	Total number of features	Total number of Tweets
STS	1291	359
STS-Gold	3677	2034
OMD	2789	1906
SOMD	1968	916

4 Experimental results

In this section, the experimental results of the various feature representation methods are compared to demonstrate the performance of the method proposed herein. The benchmark datasets, evaluation measures, and evaluation results of the proposed method in comparison with other methods are also discussed.

Table 6 the comparison of the obtained results of the proposed method with the existing methods on four datasets. (the percentage of accuracy ± SD: Standard Deviation)

Methods	SVM				MNB			
	Accuracy ± SD	Recall	Precision	F-score	Accuracy ± SD	Recall	Precision	F-score
STS dataset								
M1	82.74 ± 3.74	0.849	0.822	0.832	77.18 ± 2.40	0.832	0.759	0.791
M2	81.35 ± 3.10	0.870	0.779	0.820	82.46 ± 3.36	0.854	0.808	0.828
M3	83.02 ± 3.42	0.858	0.823	0.838	86.00 ± 4.54	**0.895**	**0.836**	**0.863**
M4	68.81 ± 4.31	0.780	0.671	0.720	73.28 ± 3.64	0.819	0.687	0.744
The proposed method	**85.52** ± 2.81	**0.873**	**0.853**	**0.861**	**86.23** ± 3.21	0.892	0.828	0.855
STS-Gold dataset								
M1	82.54 ± 3.30	0.701	0.727	0.712	66.27 ± 3.52	0.735	0.457	0.561
M2	79.99 ± 2.65	0.672	0.681	0.673	79.25 ± 2.34	**0.771**	0.588	0.663
M3	82.35 ± 3.74	0.737	0.683	0.705	80.40 ± 2.81	0.704	0.645	0.671
M4	76.65 ± 3.87	0.670	0.549	0.600	72.96 ± 3.11	0.601	0.568	0.581
The proposed method	**85.92** ± 2.10	**0.740**	**0.743**	**0.740**	**84.16** ± 2.50	0.686	**0.795**	**0.733**
OMD dataset								
M1	73.29 ± 3.33	0.606	0.712	0.655	64.32 ± 4.23	0.579	0.522	0.546
M2	85.33 ± 3.65	0.781	0.813	0.793	81.50 ± 3.71	**0.794**	0.843	0.817
M3	68.29 ± 4.21	0.581	0.689	0.627	66.89 ± 3.98	0.763	0.826	0.792
M4	73.51 ± 3.84	0.711	0.719	0.713	72.30 ± 2.86	0.733	0.789	0.758
The proposed method	**87.75** ± 3.02	**0.824**	**0.845**	**0.829**	**84.11** ± 3.55	0.792	**0.864**	**0.824**
SOMD dataset								
M1	79.58 ± 4.12	0.717	0.720	0.716	69.23 ± 4.62	0.633	0.528	0.573
M2	81.65 ± 3.73	**0.764**	0.740	0.748	76.31 ± 3.76	**0.755**	0.663	0.706
M3	68.66 ± 4.50	0.515	**0.781**	0.618	67.04 ± 4.95	0.586	0.616	0.597
M4	75.98 ± 3.34	0.599	0.690	0.640	75.00 ± 3.69	0.677	**0.784**	**0.723**
The proposed method	**83.18** ± 2.16	0.754	0.780	**0.765**	**80.23** ± 2.39	0.669	0.780	0.716

Table 7 state-of-the-art comparison for all testing datasets

Author/Reference	Year	Technique	Mean accuracy
STS dataset			
Speriosu et al. [12]	2011	Label Propagation	82.56
Keshavarz and Abadeh [6]	2017	The sum of the sentiment scores	83.34
Pandey et al. [11]	2017	K-means	78.17
Asghar et al. [14]	2018	The sum of the sentiment scores of words, slang terms, and emoticons	82.37
Ismail et al. [7]	2018	SVM	83.46
		MNB	82.79
the proposed method	–	SVM	85.52
		MNB	**86.23**
STS-gold dataset			
Speriosu et al. [12]	2011	Label Propagation	79.20
Keshavarz and Abadeh [6]	2017	The sum of the sentiment scores	84.50
Pandey et al. [11]	2017	K-means	85.64
Asghar et al. [14]	2018	The sum of the sentiment scores of words, slang terms, and emoticons	83.48
Ismail et al. [7]	2018	SVM	81.00
		MNB	82.17
the proposed method	–	SVM	**85.92**
		MNB	84.16
OMD dataset			
Speriosu et al. [12]	2011	Label Propagation	80.41
Keshavarz and Abadeh [6]	2017	The sum of the sentiment scores	79.25
Pandey et al. [11]	2017	K-means	83.32
Asghar et al. [14]	2018	The sum of the sentiment scores of words, slang terms, and emoticons	78.10
Ismail et al. [7]	2018	SVM	84.48
		MNB	80.81
the proposed method	–	SVM	**87.75**
		MNB	84.11
SOMD dataset			
Speriosu et al. [12]	2011	Label Propagation	80.54
Keshavarz and Abadeh [6]	2017	The sum of the sentiment scores	**83.34**
Pandey et al. [11]	2017	K-means	82.40
Asghar et al. [14]	2018	The sum of the sentiment scores of words, slang terms, and emoticons	82.00
Ismail et al. [7]	2018	SVM	80.11
		MNB	78.97
the proposed method	–	SVM	83.18
		MNB	80.23

4.1 Dataset description

Four datasets (Stanford Twitter Sentiment (STS),[7] STS-Gold,[8] original Obama-McCain Debate (OMD),[9] and Strict Obama-McCain Debate (SOMD)) that are known in the field of Twitter sentiment analysis and are made from tweets of users were used in this study [31]. The Stanford Twitter Dataset with a training dataset of 1.6 million tweets and testing dataset of 359 tweets was presented by Go et al. [32]. In the training dataset, all tweets were automatically labeled based on emoticons, whereas all tweets in the testing dataset were manually labeled. Although the auto-labeling of tweets is fast, its accuracy is not adequate [31]. Therefore, based on previous research such as [6, 7], the testing dataset including 359 tweets was used in this paper. The STS-Gold Twitter dataset presented by Saif et al. [31] consists of 3000 tweets that were manually labeled by three graduate students. Each tweet was

[7] Stanford dataset official page: http://help.sentiment140.com/forstudents.

[8] STS-Gold dataset: https://github.com/pollockj/world_mood/blob/master/sts_gold_v03/sts_gold_tweet.csv.

[9] OMD dataset: https://github.com/pmbaumgartner/text-feat-lib.

Table 8 Statistical test between the proposed method and the other methods for all testing datasets. (SVM)

Methods	*P* value	T-test	Hypothesis
STS dataset			
M1	0.0579	=	Not Rejected
M2	0.0075	+	Rejected
M3	0.1005	=	Not Rejected
M4	5.4781E−0.9	+	Rejected
Speriosu et al. [12]	0.0135	+	Rejected
Keshavarz and Abadeh [6]	0.0450	+	Rejected
Pandey et al. [11]	3.389E−07	+	Rejected
Asghar et al. [14]	0.0058	+	Rejected
Ismail et al. [7]	0.02613	+	Rejected
	0.00154	+	Rejected
STS-gold dataset			
M1	0.0158	+	Rejected
M2	3.1482E−05	+	Rejected
M3	0.0183	+	Rejected
M4	3.2449E−06	+	Rejected
Speriosu et al. [12]	8.937E−08	+	Rejected
Keshavarz and Abadeh [6]	0.0541	=	Not Rejected
Pandey et al. [11]	0.473	=	Not Rejected
Asghar et al. [14]	0.0145	+	Rejected
Ismail et al. [7]	7.705E−06	+	Rejected
	4.6875E−09	+	Rejected
OMD dataset			
M1	7.71117E−09	+	Rejected
M2	0.1169	=	Not Rejected
M3	6.0816E−10	+	Rejected
M4	3.5864E−08	+	Rejected
Speriosu et al. [12]	8.0955E−07	+	Rejected
Keshavarz and Abadeh [6]	3.4456E−07	+	Rejected
Pandey et al. [11]	3.2209E−04	+	Rejected
Asghar et al. [14]	1.7273E−08	+	Rejected
Ismail et al. [7]	0.0247	+	Rejected
	4.3579E−04	+	Rejected
SOMD dataset			
M1	0.0064	+	Rejected
M2	0.2814	=	Not Rejected
M3	3.4989E−08	+	Rejected
M4	2.0806E−05	+	Rejected
Speriosu et al. [12]	0.0063	+	Rejected
Keshavarz and Abadeh [6]	0.8569	=	Not Rejected
Pandey et al. [11]	0.3963	=	Not Rejected
Asghar et al. [14]	0.1922	=	Not Rejected
Ismail et al. [7]	0.0025	+	Rejected
	0.00147	+	Rejected

labeled into one of five classes: negative, positive, neutral, mixed, or other. Based on [7], only two classes of tweets, negative and positive, with a total of 2032 tweets were used in this paper. The original Obama-McCain Debate (OMD) dataset as a public dataset for TSA consists of 3238 tweets, all of which were posted during the first U.S. presidential candidate TV debate in September, 2008. At least three annotators, positive, negative, mixed, or others, were assigned to each tweet using Amazon Mechanical Turk by some voters. Since the voters had contradictory votes about these tweets, only two-thirds of the tweets upon which the voters agreed were considered. The result was a set of tweets with 1196 negative and 710 positive tweets. The Strict Obama-McCain Debate (SOMD) dataset, another version of the OMD dataset, contains 569 negative and 347 positive tweets. Since their votes are unanimous, these datasets are known as strict. The characteristics of these datasets are shown in Table 4.

4.2 Evaluation measures

According to the confusion matrix in Table 2, several performance measures can be defined. Among them, accuracy, recall, precision, and F-score are well known to researchers [27] and are the most used in research in the field of TSA, like [1, 6, 7]. Therefore, in this article, the performance of all methods on all datasets is evaluated based on these four performance measures. The accuracy measure that is utilized as a fitness function for the proposed method has been defined in (16), and the other performance measures are computed as follows:

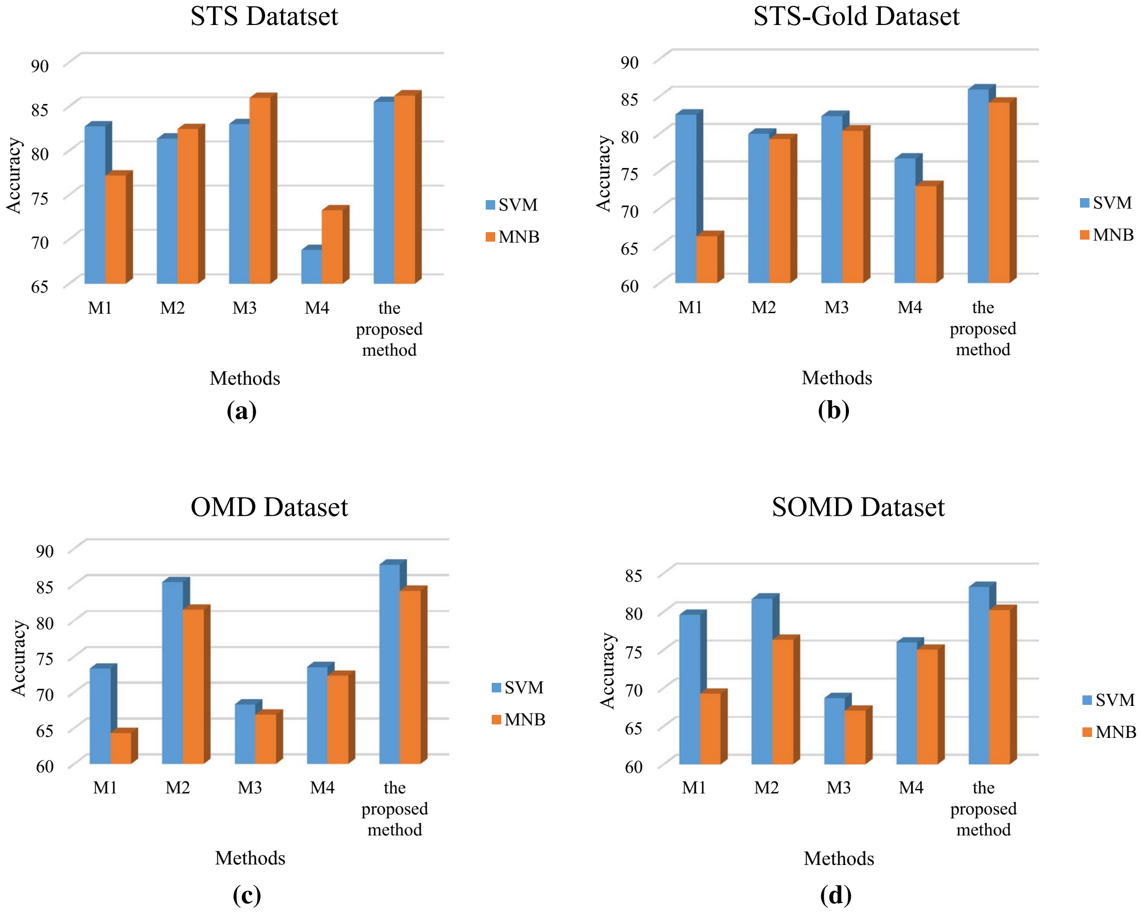
$$recall = \frac{TP}{TP + FN} \tag{25}$$

$$precision = \frac{TP}{TP + FP} \tag{26}$$

$$F - score = \frac{2 * precision * recall}{precision + recall} \tag{27}$$

4.3 Evaluation Results

This section presents the performance of the proposed method on four different twitter datasets. Here, the training and test sets of each dataset are determined based on the 10-fold cross-validation mechanism. The proposed hybrid method produces a representation matrix through the weighted combination of four representation matrices obtained from the previous methods. To prove the efficiency of the matrix produced by the proposed method, the results obtained with it are compared with those of the other matrices. Tables 4, 5, 6, 7 and 8 show this comparison with the basic matrices. In addition, the performance of the proposed method is compared with other state-of-the-art methods, and the results are shown in Table 9. All of the methods mentioned in this section are implemented on a system with characteristics such as 4 GB RAM and Intel Core i5 processor



(a) (b) (c) (d)

Fig. 3 the comparison of the accuracy of both classification algorithms in all methods and on all datasets (respectively, **a** STS dataset, **b** STS-Gold dataset, **c** OMD dataset, and **d** SOMD dataset)

that runs at 2.5 GHz. Moreover, the source codes of all of them were developed using the several packages and toolkit in Python software (e.g., NLTK, re, sklearn, and so on).

As mentioned, the pre-processing phase is as the first phase in the proposed method. Table 5 shows the results of implementing this phase on all of the datasets. Table 6 shows the experimental results of the proposed method on four datasets (STS, STS-Gold, OMD, and SOMD datasets), respectively. In this table, the results of two classification algorithms (linear SVM and MNB) for recognizing the class of the tweets in the test set are shown. Based on the 10-fold cross-validation mechanism, each classifier is trained and tested 10 times, each tweet being used 9 times for training and once for testing, and the average values of each evaluation criterion (accuracy, recall, precision, and F-score) are mentioned. Note that in Table 4; M1, M2, M3, and M4 indicate the matrices produced by semantic similarity method, TFIDF, sentiment scoring using SWN, and sentiment scoring based on the class of tweets, respectively.

The average value of accuracy along with the standard deviation mentioned in Table 6 shows that combining these four matrices as in the proposed method positively affects the accuracy of the classification algorithms. A comparison of the results based on the SVM classification algorithm shows that this algorithm can produce not only the best accuracy, but also the best F-score for all datasets, while the MNB classification algorithm demonstrates this excellence only in the value of the accuracy criterion for all datasets. Therefore, the MNB classification algorithm can only present the best result based on the F-score criterion for two datasets (STS-Gold and OMD).

The comparison of the obtained accuracy from two classification algorithms in Fig. 3 shows the superiority of one classifier over the other. In three datasets, the accuracy of the linear SVM classification algorithm in all methods, especially the proposed method, is better than the accuracy of the MNB classification algorithm. Only in the STS dataset did the MNB classification algorithm in all methods except the semantic similarity method perform better.

As described in the previous section, the suitable value for representing each feature in the proposed method is produced based on the weighted combination of values obtained from the other matrices. Here, using Fig. 4, the claim that the weighting stage using GA in the performance of the

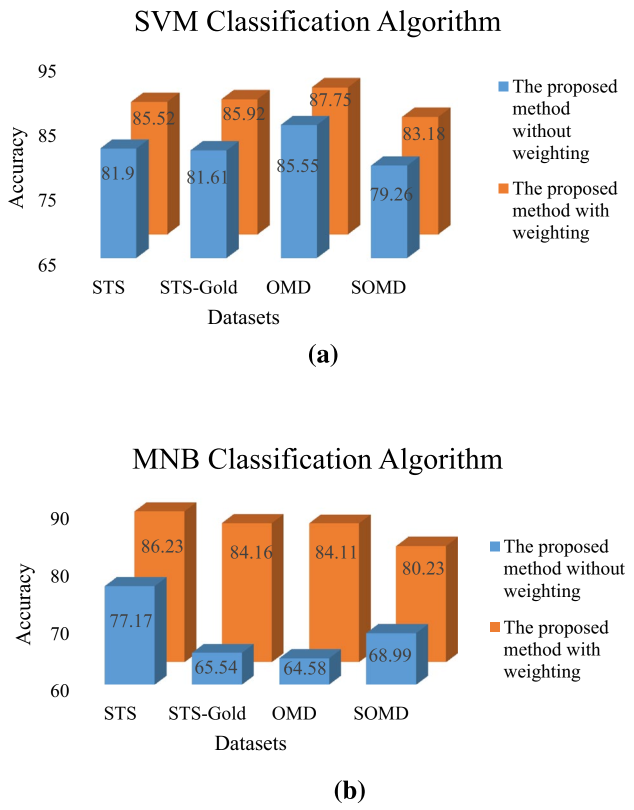


Fig. 4 the effect of the weighting stage using GA in the performance of both classification algorithms for the correct recognition of the class of tweets in testing sets

proposed method has an important role is proven. The results also demonstrate that using this stage increases the accuracy of both classification algorithms in all datasets. This figure also shows that the weighting phase increases the accuracy of the MNB classifier algorithm far more than the SVM classifier algorithm. In general, the positive effect of using a particular classification algorithm as a black box in the weighting stage is clearly seen in Fig. 4.

It is clear that the nature of the GA as a heuristic method in searching the problem space is random. For this reason, the convergence diagram is used herein to show how the genetic algorithm can find a near-optimal solution in the final iterations. Diagram of the GA convergence done to maximize the accuracy of the SVM and MNB classification algorithms are shown in Figs. 5 and 6, respectively.

As the final conclusion and in order to further test its performance the accuracy of the proposed method was compared with the accuracy of state-of-the-art methods. All five methods [6, 7, 11, 12, 14] mentioned as state-of-the-art methods were implemented on similar datasets and in the same software and hardware. Previously, these five methods have been performed not only based on different datasets, but also on systems with different hardware and software. In Table 7, the results obtained with the proposed method are compared with the results of both lexicon-based and machine learning-based approaches. As can be seen, in some datasets, like the STS dataset, lexicon-based approaches perform better than the machine-learning approaches. The reverse of this conclusion is also correct, and in some datasets, like the OMD dataset, machine learning-based approaches perform better than lexicon-based ones. The proposed method in all but the SOMD dataset performed much better than the other approaches. The results in Table 7 indicate that the method presented herein has a positive influence on the performance of the classification algorithm.

4.4 Statistical results

In order to specify which one of the generated model for TSA is identified as the best one, a test of statistical significance must be applied. Supposing that *k*-fold cross-validation has been used for each one of the models, and the average value of accuracy criterion has been calculated as the performance of each model in 10 times. Consequently, the obtained results for each model would follow a *t* distribution with *k-1* degrees of freedom. As a result, this allows us to do hypothesis testing based on *T* test (or student's *T* test) and considering it as the significance test. In this test, the null hypothesis is that the two models are the same. If we can reject it with the α significance level, then we can deduce that two models are statistically different. In other words, the model with the higher accuracy value could be selected as the efficient model for TSA. The *t* test is one of the parametric statistical tests, which can be used with power for the normal distribution data. Due to this reason, the Shapiro–Wilk test is used in order to demonstrate the obtained results distribution from the above-mentioned methods. The results of this test were indicated in Appendix A, and are presenting that the data distribution could be considered as the normal distribution.

Here, we used10-fold cross-validation for each one of the above-mentioned methods, and also examined their performance for 10 independent runs. Since in each 10-fold cross-validation round, a single test set has been used for methods, also pairwise Student's *T* test is performed as a statistical significance test. The degree of freedom in the *T* test and its significance level are selected as 9 and 0.05, respectively.

Table 8 displays the results of the statistical test between the proposed method and others. In this table, each of the sign has a meaning, consequently "+" means that the proposed method classification performance is significantly better in comparison with the other methods, and "=" indicates that both methods are as same as each other.

4.5 Time Complexity

Utilizing the GA to construct the representation matrix in *phase 2* of the proposed method is the most important part of time complexity computing. Since the computational

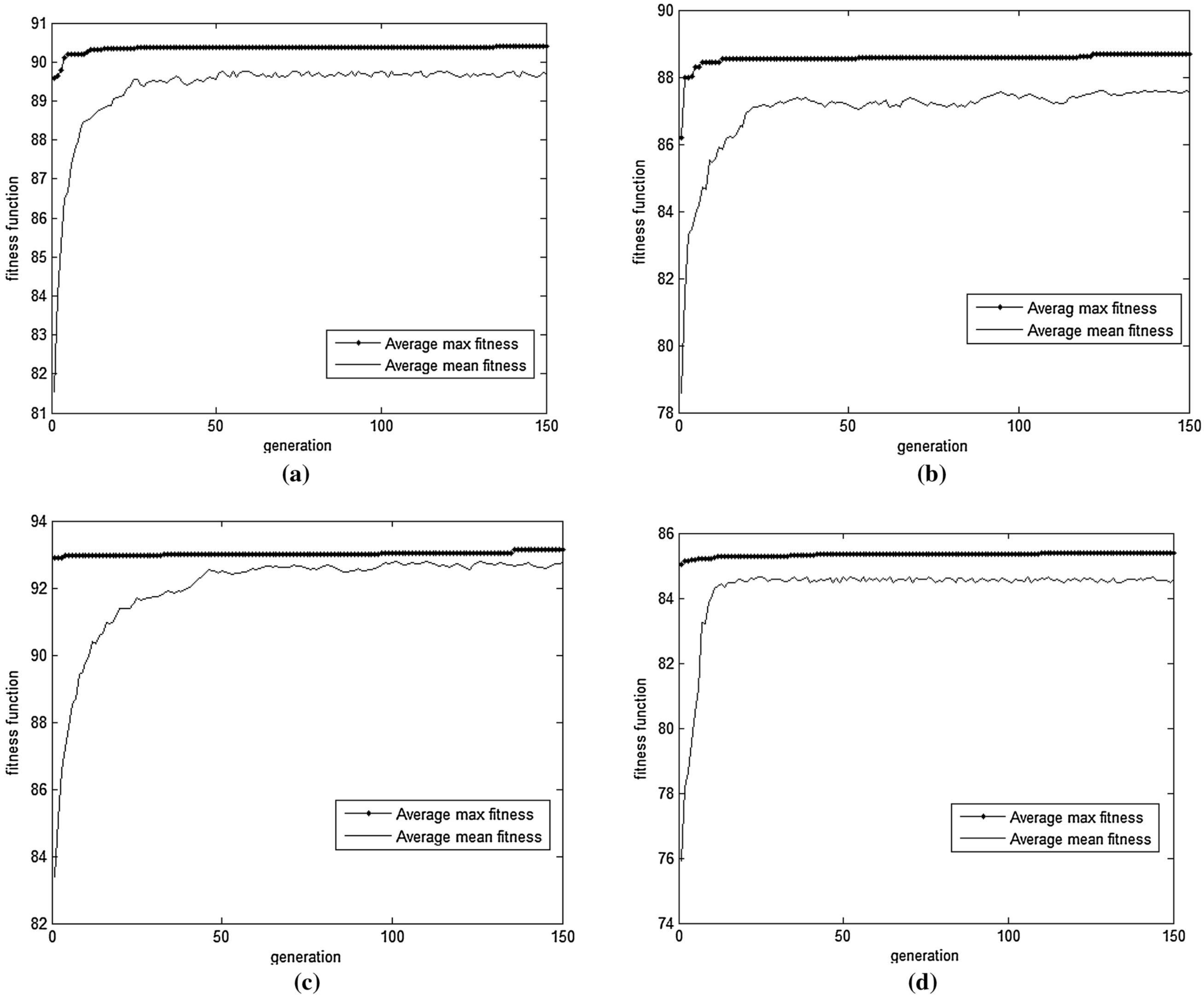
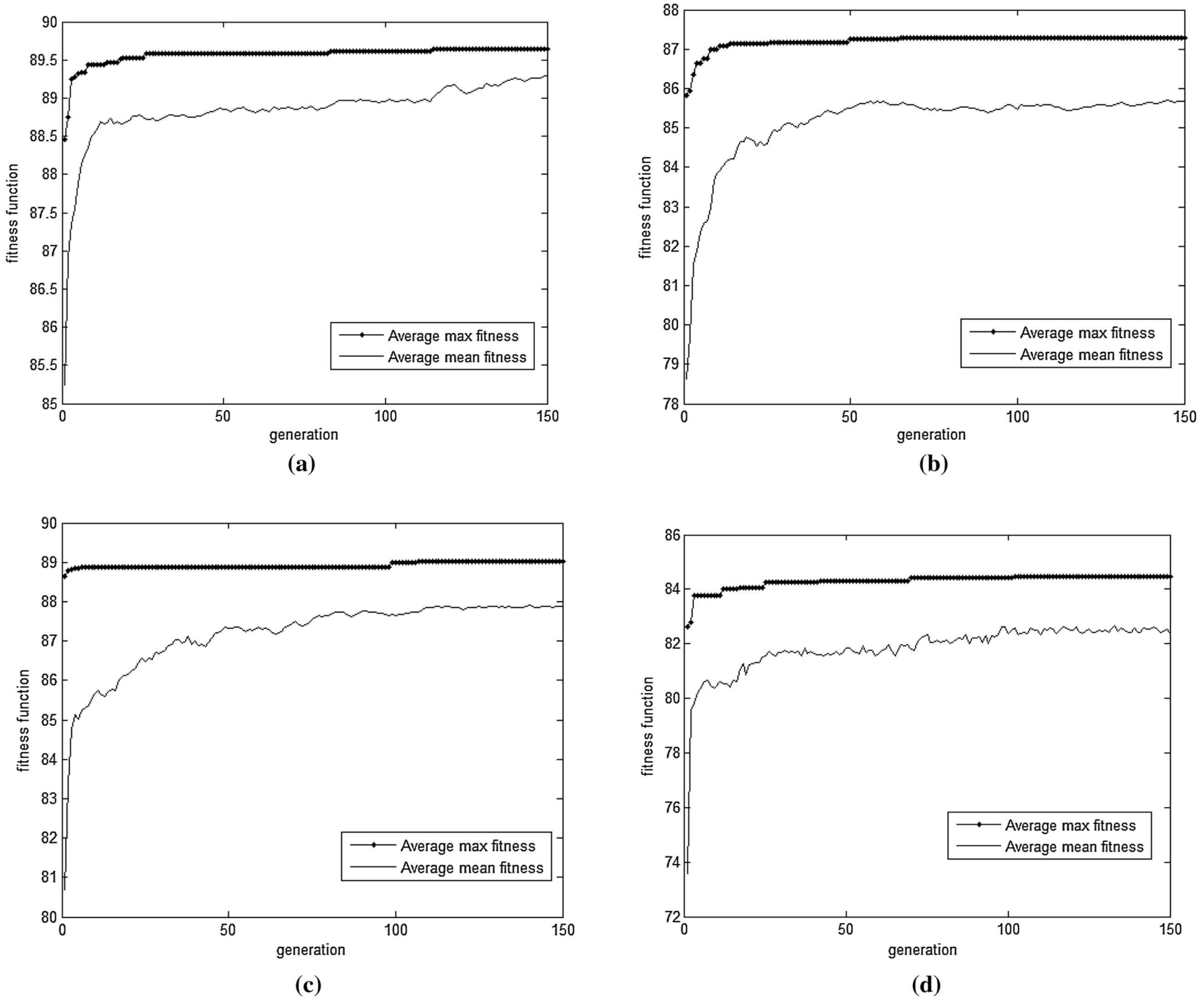


Fig. 5 the convergence diagram of GA solutions. (the fitness function is the accuracy of the SVM classification algorithm and the datasets are **a** STS dataset, **b** STS-Gold dataset, **c** OMD dataset, and **d** SOMD dataset, respectively)

complexity of the fitness function used in GA is much larger than the genetic operations (crossover, mutation, etc.), these operators were ignored in computing the time complexity of the proposed method. Thus, the time complexity of the GA is directly related to the calculation of the fitness of each chromosome (individual) in each iteration. To compute the fitness value of each chromosome, first, the *combination stage* must be executed on the four normalized matrices; then, the performance of the obtained matrix must be examined based on the accuracy criterion of the specific classification algorithm. Based on the pseudo-code in Table 3, the time complexity of the proposed method is approximated as $O(IT * N * T * F)$, where IT represents the maximum number of iterations, N is the population size, T is the total number of tweets in a dataset, and F is the total number of extracted features. According to this order of time complexity, the running time of the proposed method is increased by growing the size of the dataset.

5 Conclusion and future works

In this paper, a TSA method is proposed which includes three phases, the second of which, i.e., feature engineering, is the most important one. In fact, this paper has suggested a method for representing the features of tweets in the vector space model based on four available methods, namely, the semantic similarity method, TFIDF method, semantic scoring using SWN method, and semantic scoring based on the class of tweets method. In this phase, first, the representation matrix of each of the mentioned methods is obtained. Then, GA is applied in order to assign the percentage of participation of each method



(a)

(b)

(c)

(d)

Fig. 6 the convergence diagram of GA solutions. (the fitness function is the accuracy of the MNB classification algorithm and the datasets are **a** STS dataset, **b** STS-Gold dataset, **c** OMD dataset, and **d** SOMD dataset, respectively)

in constructing the final representation matrix. Finally, the value of each feature in the final representation matrix is achieved by a weighted combination of corresponding values in the four mentioned matrices and using the Einstein sum.

Testing the proposed method on four famous Twitter datasets (Stanford testing, STS-Gold, OMD, and SOMD datasets) and comparing it with not only the four mentioned methods, but also five state-of-the-art methods has proven the superiority of the proposed method over other methods. Although the proposed method has the best performance, the time complexity of it is not very acceptable, especially for a dataset with a high number of tweets and features. This is a disadvantage of the proposed method.

In future work, attempts will be made to correct this disadvantage by one of the following actions:

- Adding the feature selection or sample reduction phase on the dataset before the *weighing stage* in order to reduce its size by removing irrelevant features or redundant tweets.
- Improving the optimization problem by correcting the fitness function.
- Improving the genetic algorithm or replacing it with another heuristic algorithm.

Appendix A

The Shapiro–Wilk test is a normality test in statistic science and was published in 1965. At a time that the size of the sample is small, this test can be considered as an appropriate

Table 9 The results of the Shapiro–Wilk test on all methods mentioned in this paper

Methods	*P*-value	Methods	*P*-value
STS Dataset		*OMD Dataset*	
M1	0.510648	M1	0.354665
M2	0.754191	M2	0.18882
M3	0.088694	M3	0.695303
M4	0.100794	M4	0.203914
Speriosu et al. [12]	0.150088	Speriosu et al. [12]	0.067837
Keshavarz and Abadeh [6]	0.978864	Keshavarz and Abadeh [6]	0.755015
Pandey et al. [11]	0.344628	Pandey et al. [11]	0.907508
Asghar et al. [14]	0.188041	Asghar et al. [14]	0.180683
Ismail et al. [7]	0.661005	Ismail et al. [7]	0.731798
	0.843651		0.571325
The proposed method (SVM)	0.267694	The proposed method (SVM)	0.21445
STS-Gold Dataset		*SOMD Dataset*	
M1	0.077301	M1	0.753037
M2	0.498604	M2	0.170448
M3	0.545481	M3	0.575855
M4	0.815653	M4	0.586099
Speriosu et al. [12]	0.3215	Speriosu et al. [12]	0.80429
Keshavarz and Abadeh [6]	0.86736	Keshavarz and Abadeh [6]	0.667438
Pandey et al. [11]	0.874299	Pandey et al. [11]	0.299714
Asghar et al. [14]	0604009	Asghar et al. [14]	0.77676
Ismail et al. [7]	0.595255	Ismail et al. [7]	0.05105
	0.941257		0.24785
The proposed method (SVM)	0.066436	The proposed method (SVM)	0.321924

alternative. Handling the small samples (n < 20) is identified as one of this test advantages [33]. In this test, the null hypothesis is what the population is normally distributed. This hypothesis is rejected with the significant level of α, if the data tested has not been distributed normally. Table 9 indicates the results distribution is the normal distribution (the significance level 0.05), which was mentioned above in this research.

References

1. Supriya BN, Kallimani V, Prakash S, Akki CB (2016) Twitter sentiment analysis using binary classification technique. In: International conference on nature of computation and communication ICTCC 2016: nature of computation and communication pp 91–396
2. Haque MdA, Rahman T (2014) Sentiment analysis by using fuzzy logic. Int J Comput Sci Eng Inf Technol (IJCSEIT) 4:33–48
3. Shirdastian H, Laroche M, Richard M-O (2019) Using big data analytics to study brand authenticity sentiments: the case of starbucks on twitter. Int J Inf Manage 48:291–307
4. Mansour R, Hady MFA, Hosam E, Amr H, Ashour A (2015) Feature selection for twitter sentiment analysis: an experimental study. In: International conference on intelligent text processing and computational linguistics CICLing computational linguistics and intelligent text processing, pp 92–103
5. Bao Y, Quan Ch, Wang L, Ren F (2014) The role of pre-processing in twitter sentiment analysis. In: International conference on intelligent computing ICIC: intelligent computing methodologies, pp 615–624
6. Keshavarz H, Abadeh M-S (2017) ALGA: adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs. Knowl-Based Syst 122:1–16
7. Ismail H-M, Belkhouche B, Zaki N (2018) Semantic twitter sentiment analysis based on a fuzzy thesaurus. Soft Comput 22:6011–6024
8. Medhat W, Hassan A, Korashy H (2014) Sentiment analysis algorithms and applications: a survey. Ain Shams Eng J 5:1093–1113
9. Asghar M-Z, Khan A, Khan F, Kundi F-M (2018) RIFT: a rule induction framework for twitter sentiment analysis. Arabian J Sci Eng 43:857–877
10. Le B, Nguyen H (2015) Twitter sentiment analysis using machine learning techniques. In: Advanced computational methods for knowledge engineering AISC: advances in intelligent systems and computing, pp 279–289
11. Pandey A-Ch, Rajpoot D-S, Saraswat M (2017) Twitter sentiment analysis using hybrid cuckoo search method. Inf Process Manage 53:764–779
12. Speriosu M, Sudan N, Upadhyay S, Baldridge J (2011) Twitter polarity classification with label propagation over lexical links and the follower graph. In: Conference on empirical methods in natural language processing, UK, pp 53–63
13. Masud F, Khan A, Ahmad S, Asghar M-Z (2014) Lexicon-based sentiment analysis in the social web. J Basic Appl Sci Res 4(6):238–248

14. Asghar M-Z, Kundi F-M, Ahmad Sh, Khan A, Khan F (2018) T-SAF: twitter sentiment analysis framework using a hybrid classification scheme. Exp Syst 35:1–19
15. Saif H, He Y, Fernandez M, Alani H (2016) Contextual semantics for sentiment analysis of Twitter. Inf Process Manage 52:5–19
16. Khan F-H, Qamar U, Bashir S (2016) SentiMI: introducing point-wise mutual information with SentiWordNet to improve sentiment polarity detection. Appl Soft Comput 39:140–153
17. Esuli A, Sebastiani F (2006) Sentiwordnet: a publicly available lexical resource for opinion mining. In: Proceedings of the fifth international conference on language resources and evaluation, pp 417–422
18. Nielsen F-A (2011) A new ANEW: evaluation of a word list for sentiment analysis for microblogs. In: Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': big things come in small packages, pp 93–98
19. Taboada M, Brooke J, Tofiloski M, Voll K, Stede M (2011) Lexicon-based methods for sentiment analysis. Comput Lingust 37:267–307
20. Paltoglou G, Thelwall M (2010) A study of information retrieval weighting schemes for sentiment analysis. In: Proceedings of the 48th annual meeting of the association for computational linguistics: association for computational linguistics, pp 1386–1395
21. Yager RR, Kelman A (1996) Fusion of fuzzy information with considerations for compatibility, partial aggregation, and reinforcement. Int J Appr Reason 15:93–122
22. Appel O, Chiclana F, Carter J, Fujita H (2016) a hybrid approach to the sentiment analysis problem at the sentence level. Knowl-Based Syst 108:110–124
23. Gassert H (2018) Operators on fuzzy sets: zadeh and einsteinations on fuzzy sets properties of T-Norms and T-Conorms. https://pdfs.semanticscholar.org/a045/52b74047208d23d77b8aa9f5f334b59e65ea.pdf. Accessed 8 Dec 2018
24. Goldberg D-E (1989) Genetic algorithms in search optimization and machine learning. Addition Wesley, Massachusetts
25. Effrosynidis D, Symeonidis S, Arampatzis A (2017) A comparison of pre-processing techniques. In: International conference on theory and practice of digital libraries TPDL: research and advanced technology for digital libraries, pp 394–406
26. Salton G, Wong A, Yang C-S (1975) A vector space model for automatic indexing. Commun ACM 18:613–620
27. Han J, Kamber M (2006) Data mining: concepts and techniques, 2nd edn. University of Illinois at Urbana-Champaign, printed on Elsevier Inc
28. Vierira S-M, Mendonca L-F, Farinha G-J, Sousa J-M-C (2013) Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients. Appl Soft Comput 13:3494–3504
29. Gen M, Cheng R (1997) Genetic algorithms and engineering design, printed on Wiley
30. Vapnik V-N (1995) The nature of statistical learning theory. Springer, New York
31. Saif H, Fernande M, Alani YHH (2013) Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the STS-Gold. In: 1st interantional workshop on emotion and sentiment in social and expressive media: approaches and perspectives from AI (ESSEM 2013), Turin, Italy, pp 9–21
32. Go A, Bhayani R, Huang L (2010) Twitter sentiment classification using distant supervision. Technical report Stanford University
33. Shapiro SS, Wilk MB, Chen HJ (1968) A comparative study of various tests for normality. J Am Stat Assoc 63(324):1343–1372

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