Twitter sentiment analysis using hybrid cuckoo search method

Avinash Chandra Pandey*, Dharmveer Singh Rajpoot, Mukesh Saraswat

Jaypee Institute of Information Technology, Noida, India

ARTICLE INFO

ABSTRACT

Sentiment analysis is one of the prominent fields of data mining that deals with the identification and analysis of sentimental contents generally available at social media. Twitter is one of such social medias used by many users about some topics in the form of tweets. These tweets can be analyzed to find the viewpoints and sentiments of the users by using clustering-based methods. However, due to the subjective nature of the Twitter datasets, metaheuristic-based clustering methods outperforms the traditional methods for sentiment analysis. Therefore, this paper proposes a novel metaheuristic method (CSK) which is based on K-means and cuckoo search. The proposed method has been used to find the optimum cluster-heads from the sentimental contents of Twitter dataset. The efficacy of proposed method has been tested on different Twitter datasets and compared with particle swarm optimization, differential evolution, cuckoo search, improved cuckoo search, gauss-based cuckoo search, and two n-grams methods. Experimental results and statistical analysis validate that the proposed method outperforms the existing methods. The proposed method has theoretical implications for the future research to analyze the data generated through social networks/medias. This method has also very generalized practical implications for designing a system that can provide conclusive reviews on any social issues.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

The unrivalled increase in the acceptance as well as penetration of social media platforms, such as Facebook, Twitter, Google plus, etc., in a day to day life, have changed the pattern of online communication of people. Formally, user's online access was highly restricted to professional contents such as news agencies or corporations. However, these days they can seamlessly interact with each other in a more concurrent way by creating their own content within a network of peers. According to Howard (2011), "We use Facebook to schedule the protest, Twitter to coordinate, and YouTube to tell the word". Social media has emerged as a vital platform of representing people's sentiment, boosting the requirements of data mining in the field of the sentiment analysis.

In the sentiment analysis, the raw data is the online text that is exchanged by users through social media (Tang, Tan, & Cheng, 2009). Twitter, which is one of such social medias, has become the prominent source to exchange the online text, providing a vast platform of sentiment analysis. Twitter is a very popular social networking website that allows registered users to post short messages, also called tweets, up to 140 characters. Twitter database is one of the largest database having

---

* Corresponding author.
E-mail addresses: avish.nsit@gmail.com (A. Chandra Pandey), dharamveer.rajpoot@jiit.ac.in (D. Singh Rajpoot), saraswatmukesh@gmail.com (M. Saraswat).
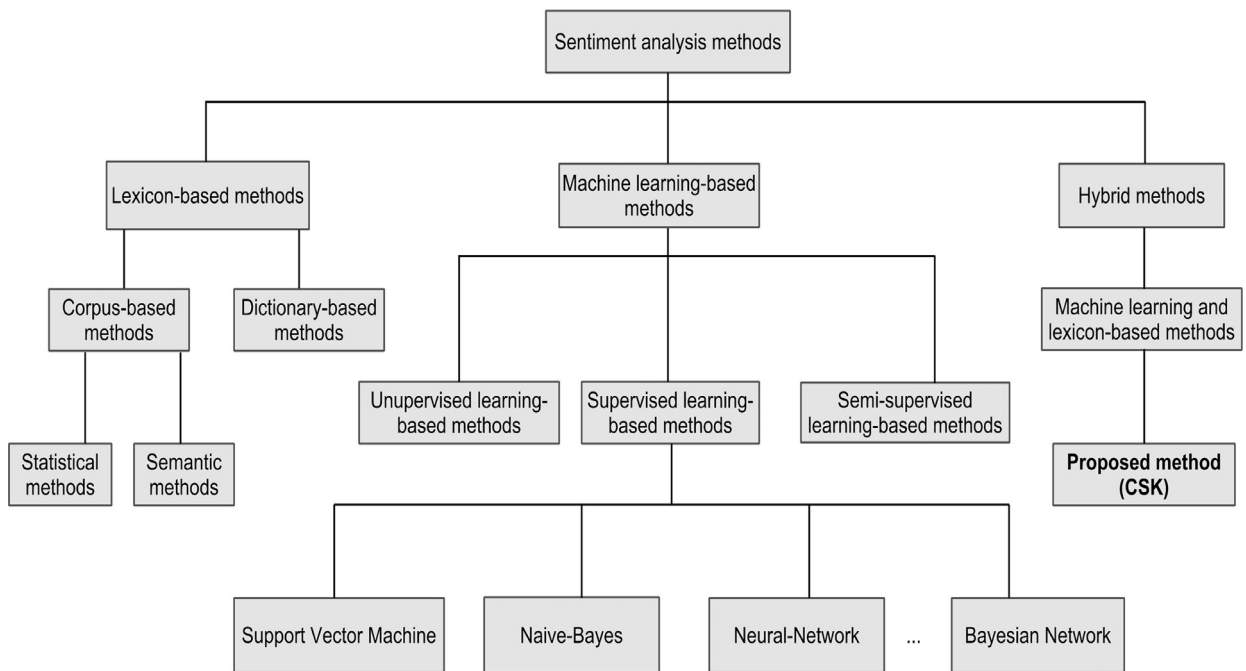
Fig. 1. Sentiment classification methods.

200 million users who post 400 million messages/tweets in a day (Ritter, Clark, Etzioni et al., 2011). At Twitter, users often share their personal opinion on different subjects such as acceptance or rejection of politicians and viewpoint about products, talk about current issues and share their personal life events. However, users post their tweets with fewer characters by using a short form of words and symbols such as emoji. Therefore, analysis of these tweets can be used to find strong viewpoints and sentiments for any topic. Twitter data has already been used by different people to predict stock market prediction (Bollen, Mao, & Zeng, 2011), box office revenues for movies (Asur & Huberman, 2010), identify the clients with negative sentiments (Thet, Na, & Khoo, 2010), etc. The main aim of sentiment analysis is to determine the attitude of users on a particular topic. Therefore, this paper proposes a novel clustering method for sentiment analysis on Twitter dataset.

Sentiment analysis methods can be broadly categorized into lexicon-based methods, machine learning-based methods, and hybrid methods (Medhat, Hassan, & Korashy, 2014) which can be further classified into sub-category as depicted in Fig. 1. Lexicon-based methods require predefined sentiment lexicon to determine the polarity of any document. However, the accuracy of lexicon-based method is reduced drastically in the presence of emoticons and short hand texts, as they are not the part of predefined sentiment lexicon (Khan, Atique, & Thakare, 2015). Emoticons are the visual emotional symbols used by the users at social medias (Hu, Tang, Gao, & Liu, 2013a). Hu, Tang, Tang, and Liu (2013b) proposed a novel method of sentiment analysis that considers the short texts like "gud nite" and emotional symbols such as ":)", in a unified framework. The performance of this method does not show stability on some of the emotional signals, such as emoticons, when used on datasets from different domains (Hu et al., 2013a). This problem can be resolved by examining the contributions of other emotion indication information existing in social media, like product ratings, restaurant reviews, and other emotion correlation information (Hu et al., 2013a; Yusof, Mohamed, & Abdul-Rahman, 2015) such as correlation between two words in a post. Emotion indication represents the sentiment polarity of a post and further, it is classified into post level emotion indication (emoticons) and world level emotion indication (publicly available sentiment lexicons) (Hu et al., 2013a). Moreover, emotion correlation for posts are usually represented by a graph in which nodes represent the data points and edge represent correlation between the words. Further, Canuto, Gonçalves, and Benevenuto (2016) proposed a new sentiment-based meta-level features for effective sentiment analysis. This method has a capability to utilize the information from the neighborhood effectively and efficiently to capture important information from highly noise data.

Bravo-Marquez, Mendoza, and Poblete (2013) introduced a novel supervised method to combine strengths, emotions, and polarities for improving the Twitter sentiment analysis process. Kontopoulos, Berberidis, Dergiades, and Bassiliades (2013) proposed ontology-based sentiment analysis of tweets. In this method, a sentiment grade has been assigned for every distinct notion in the tweets. Further, Mohammad, Zhu, Kiritchenko, and Martin (2015) analyzed US presidential electoral tweets by using supervised automatic classifiers and identified the emotional state, emotion stimulus, and intent of these tweets. Coletta, da Silva, Hruschka, and Hruschka (2014) combined the strength of SVM classifier with a cluster ensemble for refining the tweet classification. SVM classifier is executed first to classify tweets, thereafter C3E-SL algorithm has been used to enhance the classification of tweets.

Agarwal, Mittal, Bansal, and Garg (2015) introduced a new sentiment analysis model based on common-sense information mined from ConceptNet-based ontology and context knowledge. ConceptNet-based ontology is used to discover the domain specific concepts which is further used to obtain the domain specific important features. Saif, He, Fernandez, and Alani (2016a) proposed a SentiCircle method which assigns context-specific sentiment orientation to words. SentiCircle method has been introduced to update the sentiment strength of many terms dynamically. Kranjc, Smailović, Podpečan, Grčar, Žnidaršič and Lavrač (2015) has used active learning on data streams for sentiment analysis. In this method a web service, based on support vector machine, has been developed to build and update sentiment analysis models. A workflow component which uses web services has also been developed, to provide a built-in interface for tweets labeling and further constructed a new active learning sentiment analysis method. Xia, Xu, Yu, Qi, and Cambria (2016) presented a polarity shift detection elimination and ensemble (PSDEE) model, also known as cascade model, to deal with the polarity shift problem in document-level sentiment analysis. Further, Qiu, Liu, Bu, and Chen (2009) introduced a double propagation approach based on semi-supervised method for the opinion lexicon expansion and target extraction problems.

Moreover, Pandarachalil, Sendhilkumar, and Mahalakshmi (2015) have given an unsupervised and distributed method for sentiment analysis based on three domain-independent sentiment lexical resources namely; Senti-WordNet, SenticNet, and SentislangNet. Fernández-Gavilanes, Álvarez-López, Juncal-Martínez, Costa-Montenegro, and González-Castaño (2016) introduced a novel unsupervised method based on linguistic sentiment propagation model to predict the sentiments in informal texts. Due to unsupervised nature, this method does not require any training and uses linguistic content for sentiment analysis. K-means is also one of the popular unsupervised sentiment analysis methods (Boiy, Hens, Deschacht, & Moens, 2007). However, K-means method has its own limitations like data size, shape, balance, etc. For the same, overlapping clustering methods (Bello-Orgaz, Menéndez, & Camacho, 2012; Yokoyama, Nakayama, & Okada, 2009) are being used to improve the accuracy and to reduce the limitations of K-means.

Recently, sentiment analysis methods have used natural language processing (NLP) to add semantics in feature vector which improves the accuracy of the classifiers (Kanakaraj & Guddeti, 2015; Saif, Ortega, Fernández, & Cantador, 2016b). To illustrate certain facets of natural language semantics, Altınel and Ganiz (2016) proposed a novel semantic smoothing kernels which is used by class term matrices, a new type of vector space models (VSM), to extract class specific semantics. Bravo-Marquez, Frank, and Pfahringer (2016) expanded the capability of opinion lexicons methods by combining information from automatically annotated tweets with existing opinion lexicons. Further, Muhammad, Wiratunga, and Lothian (2016) introduced a lexicon-based sentiment classification system which uses textual neighborhood (local context) interaction and text category (global context) for social media genres. Moreover, Appel, Chiclana, Carter, and Fujita (2016) presented a hybridized method which uses NLP and fuzzy sets to determine semantic polarity and its intensity for posts.

Furthermore, Cambria (2016) discussed merits and limitations of various sentiment analysis methods such as knowledge-based, statistical, and hybrid. Shah et al. (2016) presented a multimedia summarization system to analyze online user-generated contents (UGCs) from multiple modalities. For the same, they have used the EventBuilder system for semantics understanding and EventSensor system for sentics understanding. Chen, Xu, He, Xia, and Wang (2016) introduced a document-level sentiment analysis method using sequence modeling-based neural network. Further, Sulis, Farías, Rosso, Patti, and Ruffo (2016) investigated the effect of figurative linguistic phenomena in twitter to separate the tweets with tag #irony, #sarcasm and #not using psycholinguistic and emotional features.

Sarcasm detection in Twitter dataset, is a recent area of research. YourDictionary (sar, 2016) defines sarcasm as "an ironic or satirical remark that seems to be praising someone or something but is really taunting or cutting". Due to this nature, it is very difficult to decide whether a statement is sarcastic or not. Twitter is one of the social media where people express their feelings in the form of tweets, which may also be sarcastic. Carvalho, Sarmento, Silva, and De Oliveira (2009) used linguistic features such as interjection, punctuation, etc. for sarcasm detection. González-Ibánez, Muresan, and Wacholder (2011) had studied how unigrams and emoticons can be used to identify sarcastic tweets. Reyes, Rosso, and Buscaldi (2012) had given a hashtag-based method to decide whether a tweet is sarcastic or not. Further, Joshi, Sharma, and Bhattacharyya (2015) presented a computational system which uses context incongruity as a basis for sarcasm detection. Joshi, Bhattacharyya, and Carman (2016) presented a survey in which they discussed many approaches for sarcasm identification on different datasets.

Metaheuristic-based methods have also been used for sentiment analysis. Basari, Hussin, Ananta, and Zeniarja (2013) proposed a hybrid method based on support vector machine (SVM) and particle swarm optimization (PSO) to categorize a movie into watchable and non-watchable. However, the SVM-PSO based method does not perform well for multiclass sentiment classification. Further, Gupta, Reddy, Ekbal et al. (2015) proposed a PSO-Asent method for selecting features from text and sentiment classification using PSO for aspect-based sentiment analysis. The accuracy of PSO-Asent method depends upon reduced set of features and sometimes suffers in case of unlabeled product reviews. Further, Zhu, Wang, and Mao (2010) proposed a hybrid method based on genetic algorithm (GA) and conditional random forest (CRF) to classify sentiments.

Due to the above mentioned limitations of traditional as well as metaheuristic-based clustering methods, this paper introduces a novel metaheuristic method (CSK) which is being used to cluster the sentimental contents. The proposed method, which is based on cuckoo search (CS) (Yang & Deb, 2009) and K-means (Žalik, 2008), optimizes the cluster-heads of sentimental datasets. Moreover, the performance of the proposed method has been compared with cuckoo search (CS), improved cuckoo search algorithm (ICS) (Valian, Mohanna, & Tavakoli, 2011), Gauss based cuckoo search algorithm (GCS) (Zheng & Zhou, 2012), particle swarm optimization algorithm (PSO) (Kennedy & Eberhart, 1995), differential evolution (DE) (Storn & Price, 1997), and two n-grams (basic baseline method). The rest of the paper is organized as follows: Section 2 de-

scribes preliminaries namely; K-means and CS method. The proposed method is explained in Section 3. Section 4 discusses experimental results and Section 5 concludes the paper.

2. Preliminaries

The proposed method uses two existing methods namely; K-means and CS which are described in the following sections.

2.1. K-means method

K -means (Kogan, Teboulle, & Nicholas, 2005; Žalik, 2008) data clustering method groups *n* data points in *K* clusters by iteratively minimizing the distance of data points from the *K* cluster-heads. Distance can be calculated either by using Euclidean distance (Danielsson, 1980) or cosine measure (Wilkinson & Hingston, 1991). *K*-means method gets a partition that minimizes the squared error between data points and empirical mean of a cluster. In *K*-means clustering, deciding the value of *K* for a dataset having unknown number of classes is a challenging task. In such cases, elbow method (Nugent, Dean, & Ayers, 2010), information criterion method (Jain, 2010), silhouette method (Chiang & Mirkin, 2010), etc. can be used to select *K* value. The procedure of K-means clustering method (Žalik, 2008) is described in the Algorithm 1.

Algorithm 1 K-means method.

1. Initialize the *k* cluster-heads randomly
2. Assign each data points to closest cluster
3. New cluster-head C_i of each cluster is calculated using following formula:

$$C_i = \frac{1}{n_i} \sum_{\forall d_i \in S_i} d_i, i = 1, 2, \ldots, k$$

where d_i denotes the data points that belong to the cluster S_i; n_i is the number of documents in cluster S_i
4. Repeat Steps 2 and 3 until convergence

2.2. Cuckoo search

CS has been developed by Yang and Deb (2009) which is a meta-heuristic optimization method and based on the obligate brood parasitic conduct of cuckoos. CS method uses the following three idealized rules:

1. Each cuckoo lays one egg at a time and dumps it, in a randomly chosen nest.
2. Nest with high quality of eggs will be the best nest and will carry over the next generations.
3. The number of available host nest is fixed, and the egg laid by a cuckoo is discovered by the host bird with a probability $P_a \in [0, 1]$. If the host bird discovers alien egg, it either throws the eggs away or abandons the nest and built the new one. In short, this rule can be approximated as fraction P_a of worse eggs which are replaced by new eggs.

Basic steps of CS method is represented by Algorithm 2 (Yang & Deb, 2009). In the CS method new solutions $x(t + 1)$ for a cuckoo *i* is generated using Eq. (1).

$$x_i(t + 1) = x_i(t) + \alpha \oplus Levy(\lambda) \tag{1}$$

here α is a step size scaling factor that scales step size produced by Brown, Liebovitch, and Glendon (2007); Pavlyukevich (2007). Eq. (1) is used for a random walk, where next state/location depends on the current location and transition probability P_a. The product $\oplus$ represents entry wise multiplications. Search space is explored by using Lévy flight as its step length is much longer in the long run. Fraction P_a of the worse nest is abandoned and new ones are built using a biased random walk.

3. Proposed method

The proposed method (CSK) clusters the input tweets in three phases; (i) preprocessing of the tweets, (ii) feature extraction, and (iii) hybrid clustering using K-means and cuckoo search. The detailed flow chart of the proposed method has been shown in Fig. 2.

3.1. Preprocessing

The raw tweets, collected from Twitter, have noise in terms of unwanted and fuzzy words, URLs, stopwords etc., which are needed to be reduced before feature extraction. Therefore, the proposed method uses the following preprocessing method in two phases before extracting the features:

Algorithm 2 Cuckoo search.

Set the initial parameters:
$-P_a$ (the probability of worse nests)
$-MaxGeneration$ (the maximum number of iterations)
$-n$ (the size of population)
Objective function $f(x), x = (x_1, \ldots, x_d)^T$
Generate initial population of n host nests, $x_i (i = 1, 2, \ldots, n)$
$counter = 1$
while $counter <= MaxGeneration$ or $stoppping - criterion$ **do**
   Move a randomly selected cuckoo (x_i) by Levy flights and termed it as new solution (x_{new})
   Evaluate its fitness $F_{(x_{new})}$
   Randomly choose a nest x_j among n available nests and evaluate its fitness $(F_{(x_j)})$
   if $F_{(x_{new})} > F_{(x_j)}$ **then**
     Replace x_j by the new solution(x_{new})
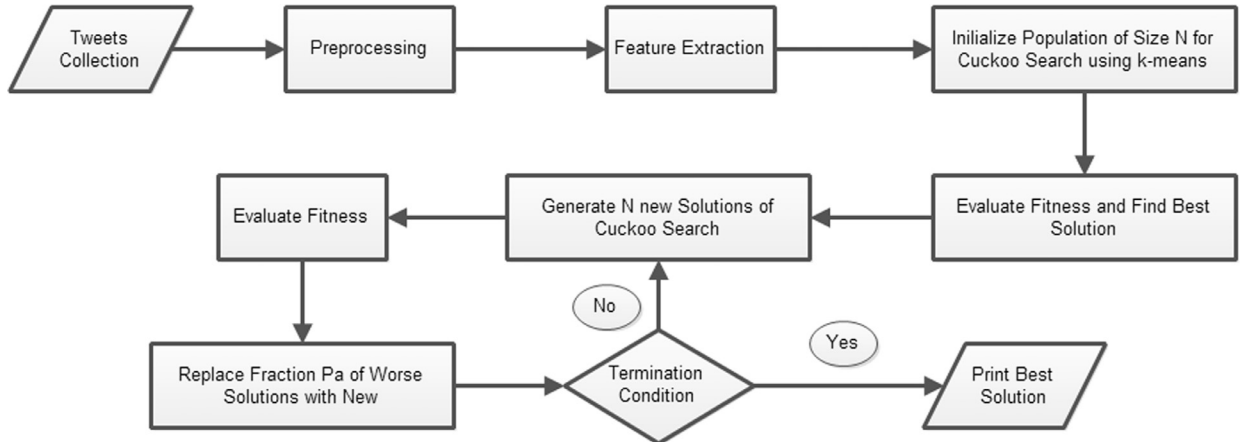   **end if**
   Fractions (P_a) of worse nests are abandoned and new ones are built using biased random
   walk
   Compare the worse nests with new ones and keep the better solutions
   Rank the solutions and find the current best
end while

Fig. 2. Flowchart of the proposed hybrid cuckoo search method (CSK).

3.1.1. Phase 1

This phase eliminates unwanted noise elements from the Twitter data set using the following steps:

1. Eliminate all the URLs via regular expression matching. A regular expression is a textual pattern that defines a search pattern for strings/text. It can be used to search for URLs, email address etc. The list of regular expression used in this paper is shown in Appendix A.
2. Replace "@Username" with "usr" using regular expression matching.
3. Since "hash-tag(#)" provides some useful information, therefore remove only #, keeping the word as it is. *viz.*, " #Lee" is replaced with "Lee".
4. Remove parenthesis, forward slash (/), backward slash (\), and dash from tweets.
5. Replace multiple white spaces with single white space.

3.1.2. Phase 2

In this phase, two dictionaries namely; stop word (sto, 2015) and acronym (Acronym dictionary, 2015) have been deployed to improve the precision of resultant Twitter dataset of Phase 1. The steps of Phase 2 are as follows:

1. Convert all the words of tweets into lowercase.
2. Remove all the stop words such as, a, is, the, etc. by comparing them with stop word dictionary (sto, 2015).
3. Replace sequence of repeated characters (three or more) in a word by one character *viz.*, "hellooooo" is converted to "Hello".
4. Eliminate words which do not start with an alphabet.

Table 1
Considered Twitter datasets.

Sr.No.	Dataset	Number of Instances	Number of classes	Positive	Negative	Neutral	Date Range	Topics Covered
1.	Testdata.manual.2009.06.14	498	3	182	177	139	May 11, 2009 to Jun 14, 2009	Google, Obama, Kindle, China, etc.
2.	Twitter-sanders-apple2	479	2	163	316	–	Oct 15, 2011 to Oct 20, 2011	Apple, Google, Microsoft, Twitter
3.	Twitter-sanders-apple3	988	3	163	316	509	Oct 15, 2011 to Oct 20, 2011	Apple, Google, Microsoft, Twitter
4.	Twitter dataset	2000	2	1000	1000	–	Nov 17, 2014 to Dec 10, 2014	Sports, Saints, Funny Images, etc.

Table 2
Details of extracted features from Twitter dataset.

Sr.No.	Name of Feature	Testdata.manual.2009.06.14		Twitter-sanders-apple2		Twitter-sanders-apple3		Twitter dataset	
		Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation
1.	Total Characters	0.4525	0.2201	0.5601	0.2082	0.5402	0.2013	0.3633	0.1612
2.	Positive Emoji	0.0302	0.1274	0.0250	0.1564	0.0111	0.0771	0.0020	0.0446
3.	Negative Emoji	0.0261	0.1598	0.0188	0.1359	0.0070	0.0839	0.0020	0.0446
4.	Neutral Emoji	0.0423	0.2014	0.0008	0.0910	0.0040	0.1560	0.0030	0.0547
5.	Positive Exclamation	0.0060	0.0775	0.0146	0.1201	0.0041	0.0635	0.0005	0.0223
6.	Negative Exclamation	0.0081	0.0894	0.0229	0.1499	0.0171	0.1300	0.0025	0.0499
7.	Negation	0.0654	0.2457	0.0553	0.1698	0.0470	0.1560	0.0530	0.1618
8.	Positive Words	0.1159	0.2463	0.0501	0.1357	0.0404	0.1207	0.0246	0.0691
9.	Negative Words	0.0241	0.0879	0.0438	0.1556	0.0258	0.1194	0.0170	0.0920
10.	Neutral Words	0.1102	0.2105	0.0125	0.1113	0.0050	0.1194	0.0015	0.0387
11.	Intense Words	0.0222	0.0932	0.0626	0.2426	0.0298	0.1227	0.0298	0.0899

Table 3
Parameter settings for all the considered datasets.

Sr.No.	Parameter	CS	ICS	GCS	PSO	DE	CSK	n-grams
1.	Probability (P_a)	0.25	[0.05, 0.5]	0.25	–	–	0.25	–
2.	Step scaling factor (α)	0.01	[0.01, 0.5]	0.01	–	–	0.01	–
3.	Number of iterations	600	600	600	600	600	600	–
4.	Cognitive constant (c_1)	–	–	–	2	–	–	–
4.	Social constant (c_2)	–	–	–	2	–	–	–
5.	Inertia weight (w)	–	–	–	0.8	–	–	–
5.	Crossover rate (CR)	–	–	–	–	0.5	–	–
6.	Mutation rate (F)	–	–	–	–	0.8	–	–
7.	gram sequence (n)	–	–	–	–	–	–	3

5. Replace all the short forms in the respective full forms using acronym dictionary (Acronym dictionary, 2015).

3.2. Feature extraction method

After applying the preprocessing (Haddi, Liu, & Shi, 2013; Uysal & Gunal, 2014), tweets are converted into the feature vector by calculating the following 11 features from the Twitter dataset.

1. **Total Characteristics:** It represents the total number words available in the tweets.
2. **Positive Emoji:** Positive emoji, such as :), ;), : D, etc., are the symbols used to express happy moments. This feature uses a positive emoticon dictionary (Emo, 2015) to count the total number of positive emojis in the tweets.
3. **Negative Emoji:** The special symbols used to express sad/ negative feelings, such as : (, : '(, > : (, etc., are known as negative emoji. To get the total counts of negative emoji in tweets a negative emoticon dictionary (Emo, 2015) is used.
4. **Neutral Emoji:** Neutral emoji (straight-faced emoji) do not provide any particular emotion. Total neutral emoji is counted by comparing tweets with neutral emoticon dictionary (Emo, 2015).
5. **Positive Exclamation:** Exclamatory words, such as hurrah! wow! etc., can be used to convey a very strong feeling/ opinion about the topic. For the same, positive exclamation dictionary (int, 2015) is used to count the positive exclamation.
6. **Negative Exclamation:** Negative exclamations are counted by comparing the tweet with negative exclamation dictionary (int, 2015).
7. **Negation:** To express the negative opinion, negation words like no, not, etc., are generally used. Therefore, this feature counts the negation words in the tweet by comparing it with negation words.

Table 4

Comparison of proposed method with the existing methods in terms of mean accuracy, mean computational time, and mean fitness function value.

Sr.No.	Dataset	Method	Mean Accuracy	Mean Computational Time	Mean Fitness Function Value
1.	Testdata.manual.2009.06.14	CS	59.54%	328.23	0.2506
2.		ICS	61.48%	319.30	0.2508
3.		GCS	60.41%	349.63	0.2493
4.		PSO	59.28%	306.93	0.2434
5.		DE	59.36%	**291.81**	0.2467
6.		SVM-tri	44.47%	324.15	–
7.		NB-tri	41.23%	319.34	–
8.		CSK	**78.17**%	296.95	**0.2627**
1.	Twitter-sanders-apple2	CS	58.28%	293.25	0.2606
2.		ICS	58.29%	291.56	0.2602
3.		GCS	56.81%	280.95	0.2603
4.		PSO	57.24%	277.46	0.2595
5.		DE	57.45%	**236.29**	0.2608
6.		SVM-tri	60.51%	264.98	–
7.		NB-tri	52.50%	272.16	–
8.		CSK	**84.16**%	241.23	**0.2629**
1.	Twitter-sanders-apple3	CS	63.62%	472.75	0.2447
2.		ICS	64.85%	462.29	0.2434
3.		GCS	63.07%	467.24	0.2413
4.		PSO	62.17%	478.48	0.2423
5.		DE	63.01%	**432.77**	0.2455
6.		SVM-tri	52.27%	513.95	–
7.		NB-tri	50.53%	509.07	–
8.		CSK	**82.21**%	443.42	**0.2519**
1.	Twitter dataset	CS	50.58%	782.28	0.2389
2.		ICS	54.63%	675.50	0.2473
3.		GCS	52.60%	575.51	0.2401
4.		PSO	50.55%	644.43	0.2429
5.		DE	51.60%	**340.11**	0.2371
6.		SVM-tri	56.15%	952.54	–
7.		NB-tri	55.25%	959.14	–
8.		CSK	**67.45**%	482.54	**0.2606**

8. **Positive Words:** This feature counts the number of positive words like achieve, confidence, etc., using positive word dictionary (jeffreybreen, 2015; Liu, Hu, & Cheng, 2005). If there are two negative words (double negation) then these words are counted as single positive word.
9. **Negative Words:** This feature represents the total counts of negative words such as bad, lost, etc., in tweets (jeffreybreen, 2015; Liu et al., 2005).
10. **Neutral Words:** Neutral words (okay, rarely) do not provide any particular emotion/feeling. Total counts of neutral words are obtained by comparing the tweets with neutral word dictionary (psy, 2015).
11. **Intense Words:** Intense words, like very, much etc. are used in a sentence to make it more effective/intense. Total counts of intense words are determined by using intense word dictionary (psy, 2015).

Moreover, the value of above mentioned features may be affected due to the presence of sarcasm or irony in tweets. To deal with the problem of sarcasm or irony in Twitter dataset, the proposed method uses explicit incongruity (Joshi et al., 2015), implicit congruity (Joshi et al., 2015), pragmatic features (smilies, emoticons, etc.) (Bharti, Vachha, Pradhan, Babu, & Jena, 2016), and hyperbole features (interjection, quotes, punctuation, etc.) (Bharti et al., 2016) . In a tweet, explicit incongruity (Joshi et al., 2015) is evident through the presence of both the polarity words (positive and negative), especially if the tweet has a prior positive polarity. For example "I love being annoyed", where love is positive word and annoyed is negative word. To detect this type of tweet, positive and negative words are counted along with their order and corresponding feature value of negative words increases. Further, some tweets contain negative word prior to positive word such as , "I hate Usain Bolt, because he always win". These tweets seems to be negative but actually these are positive. In this case, total count of positive and negative words are obtained and if the counts of positive and negative word are equal and positive word follows negative word, the value of positive feature word is incremented.

Implicit incongruity (Joshi et al., 2015) in tweets are identified through the presence of an implied phrase opposing a positive polarity word. For example; "I love maths so much that I gain least marks in it". Here, only polar word is " love" and clause, "I gain least marks", has incongruous implied with polar word "love". In this case, negative feature value is updated.

Hyperbole (Bharti et al., 2016; Bharti, Babu, & Jena, 2015) in tweets is the combination of features such as interjection, intensifier (adjective, adverbs), quotes, and punctuation marks. In the proposed method, count of negative word feature is

Table 5
Comparative results of the paired student's *t*-test for mean accuracy for Twitter datasets where the proposed method (CSK) is paired, compared to existing methods.

Sr.No.	Dataset	Method	Standard Error	t	95% of Confidence Interval	Two- tailed P	Significance
1.	Testdata.manual.2009.06.14	CS	0.0123	1312.27	59.03–59.78	4.73E−71	Extremely significant
2.		ICS	0.0212	678.34	60.78–61.64	1.34E−61	Extremely significant
3.		GCS	0.0344	460.47	60.01–60.74	2.10E−57	Extremely significant
4.		PSO	0.0410	321.94	58.90–59.93	1.72E−53	Extremely significant
5.		DE	0.0400	328.60	58.65–59.41	1.34E−53	Extremely significant
6.		SVM-tri	0.0401	290.21	44.38–44.54	1.10E−55	Extremely significant
7.		NB-tri	0.0411	284.45	40.98–41.54	2.34E−54	Extremely significant
1.	Twitter-sanders-apple2	CS	0.0012	73621.09	57.81–58.74	2.32E−116	Extremely significant
2.		ICS	5.12E−12	7,414,525	57.85–58.37	2.31E−170	Extremely significant
3.		GCS	1.11E−3	812514.80	56.20–56.95	7.22E−142	Extremely significant
4.		PSO	0.0004	76658.80	56.80–57.37	2.33E−108	Extremely significant
5.		DE	0.0004	81287.75	57.04–57.97	1.41E−121	Extremely significant
6.		SVM-tri	0.0301	26180.21	60.45–66.72	1.32E−119	Extremely significant
7.		NB-tri	0.0312	25190.21	51.77–52.89	1.05E−117	Extremely significant
1.	Twitter-sanders-apple3	CS	0.0007	24455.37	62.68–63.75	2.15E−106	Extremely significant
2.		ICS	0.0010	16480.51	64.18–64.94	2.36E−101	Extremely significant
3.		GCS	0.0003	56,451	62.93–63.41	1.32E−115	Extremely significant
4.		PSO	0.0005	42543.40	61.96–62.40	2.11E−111	Extremely significant
5.		DE	0.0003	93050.18	62.84–63.88	1.20E−114	Extremely significant
6.		SVM-tri	0.0201	40480.21	52.11–52.35	1.35E−113	Extremely significant
7.		NB-tri	0.0104	35290.74	50.08–50.84	2.10E−111	Extremely significant
1.	Twitter dataset	CS	0.0020	621.71	50.23–50.88	2.01E−60	Extremely significant
2.		ICS	0.0003	334.72	54.21–54.87	2.30E−53	Extremely significant
3.		GCS	0.0003	362.50	52.40–52.97	2.12E−53	Extremely significant
4.		PSO	0.0002	659.96	50.14–50.73	3.20E−62	Extremely significant
5.		DE	0.0002	676.60	51.05–51.90	2.87E−62	Extremely significant
6.		SVM-tri	0.0104	644.21	56.02–56.30	1.15E−47	Extremely significant
7.		NB-tri	0.0102	640.11	55.44–55.82	3.06E−50	Extremely significant

Table 6
Comparative results of the paired student's *t*-test for mean computational time for Twitter datasets where the proposed method (CSK) is paired, compared to existing methods.

Sr.No.	Dataset	Method	Standard Error	t	95% of Confidence Interval	Two- tailed P	Significance
1.	Testdata.manual.2009.06.14	CS	0.0409	−636.46	327.77–328.94	2.21E−61	Extremely significant
2.		ICS	0.0444	−452.37	317.77–319.96	6.42E−37	Extremely significant
3.		GCS	0.0448	−882.19	348.62–350.12	3.13E−55	Extremely significant
4.		PSO	0.0462	−189.10	304.15–306.95	1.56E−44	Extremely significant
5.		DE	0.0415	150.17	290.45–292.25	2.23E−45	Extremely significant
6.		SVM-tri	0.0220	−643.12	326.81–333.85	3.41E−60	Extremely significant
7.		NB-tri	0.0209	−681.36	310.76–323.42	1.18E−58	Extremely significant
1.	Twitter-sanders-apple2	CS	0.1021	−482.12	290.46–294.89	4.01E−54	Extremely significant
2.		ICS	0.1043	−430.28	290.70–294.13	6.53E−55	Extremely significant
3.		GCS	0.1017	−343.60	278.73–282.15	4.91E−54	Extremely significant
4.		PSO	0.1102	−256.57	275.01–278.46	3.61E−51	Extremely significant
5.		DE	0.0932	90.07	234.77–237.17	2.50E−88	Extremely significant
6.		SVM-tri	0.0125	−313.23	259.18–274.25	3.40E−48	Extremely significant
7.		NB-tri	0.0101	−344.16	268.23–287.36	2.23E−52	Extremely significant
1.	Twitter-sanders-apple3	CS	0.1037	−151.37	469.69–473.12	1.39E−43	Extremely significant
2.		ICS	0.0940	−140.84	460.63–466.03	4.46E−42	Extremely significant
3.		GCS	0.0962	−165.47	464.77–469.23	6.46E−42	Extremely significant
4.		PSO	0.0976	−201.48	476.71–480.18	8.53E−46	Extremely significant
5.		DE	0.0917	96.40	430.71–434.08	5.22E−37	Extremely significant
6.		SVM-tri	0.0225	−213.23	515.05–520.12	7.42E−40	Extremely significant
7.		NB-tri	0.0225	−213.23	504.24–516.41	4.62E−42	Extremely significant
1.	Twitter dataset	CS	0.0503	5483.10	777.15–786.25	3.86E−89	Extremely significant
2.		ICS	0.0656	2129.69	674.16–677.34	2.46E−56	Extremely significant
3.		GCS	0.0562	1194.63	572.22–577.51	1.30E−67	Extremely significant
4.		PSO	0.0343	3493.94	643.45–645.57	3.74E−80	Extremely significant
5.		DE	0.0420	3402.04	338.59–342.72	2.35E−80	Extremely significant
6.		SVM-tri	0.0302	2134.13	930.91–948.93	1.16E−82	Extremely significant
7.		NB-tri	0.0217	2047.21	942.74–967.47	2.52E−80	Extremely significant

Table 7
Comparative results of the paired student's *t*-test for mean fitness function value for Twitter datasets where the proposed method (CSK) is paired, compared to existing methods.

Sr.No.	Dataset	Method	Standard Error	t	95% of Confidence Interval	Two- tailed P	Significance
1.	Testdata.manual.2009.06.14	CS	0.0001	69.10	0.2413–0.2531	1.14E−51	Extremely significant
2.		ICS	0.0001	86.58	0.2403–0.2528	1.45E−36	Extremely significant
3.		GCS	0.0020	6.71	0.2474–0.2499	2.34E−7	Extremely significant
4.		PSO	0.0017	10.97	0.2430–0.2443	7.8E−12	Extremely significant
5.		DE	0.0023	6.88	0.2450–0.2468	147E−7	Extremely significant
1.	Twitter-sanders-apple2	CS	0.0001	14.82	0.2596–0.2610	4.55E−15	Extremely significant
2.		ICS	0.0001	23.52	0.2593–0.2605	4.36E−81	Extremely significant
3.		GCS	9.84E−5	26.81	0.2593–0.2607	5.12E−22	Extremely significant
4.		PSO	4.35E−5	77.44	0.2583–0.2600	3.65E−35	Extremely significant
5.		DE	0.0001	19.53	0.2596–0.2611	3.11E−18	Extremely significant
1.	Twitter-sanders-apple3	CS	0.0002	29.75	0.2393–0.2457	2.77E−23	Extremely significant
2.		ICS	0.0002	37.18	0.2403–0.2440	5.07E−26	Extremely significant
3.		GCS	0.0002	43.93	0.2393–0.2417	4.40E−28	Extremely significant
4.		PSO	0.0002	33.27	0.2397–0.2427	1.19E−29	Extremely significant
5.		DE	0.0002	24.24	0.2419–0.2457	8.48E−28	Extremely significant
1.	Twitter dataset	CS	0.0023	9.17	0.2373–0.2397	4.43E−10	Extremely significant
2.		ICS	0.0016	7.98	0.2423–0.2487	8.36E−09	Extremely significant
3.		GCS	0.0012	16.95	0.2393–0.2407	1.37E−16	Extremely significant
4.		PSO	0.0062	2.92	0.2410–0.2437	0.0065	Extremely significant
5.		DE	0.0063	3.80	0.2363–0.2387	0.0006	Extremely significant

increased if interjection appears at the starting point and intensifier appears other than the starting point in any tweet. For example; "Wow, thats a huge discount, Im not buying anything!!", contains "wow" at the beginning and "not" at other than beginning. Hence, this is sarcasm. Further, due to message limit of tweets, pragmatic features involve symbolic and figurative text such as smiles, emoticons, etc. in tweets. These features play an important role to discover sarcasm in tweets. For example "I work 40 hours a week to be this poor :)" consist of positive and negative words along with positive emoji. In these type of situations, negative feature value is increased.

3.3. Hybrid clustering using K-means & cuckoo search (CSK)

The normalized feature vector is given input to the proposed clustering method (CSK) which uses K-means and cuckoo search method to cluster the data. As K-means is very popular cluster method, but it generally stuck to initial clusters which is a major drawback of K-means method. However, the generated clusters can be used for further analysis. Therefore, in this method, the generated clusters from K-means have been used in the cuckoo search method for further optimizing the cluster-heads. Since, in the cuckoo search, a random initialization of the population is required and this may increase the number of iterations to converge and also stuck to some local solution. Therefore, this method modifies the initialization process of cuckoo search which results in faster convergence and better optimum solution. In the CSK, the solutions obtained from K-means are used to initialize the population of cuckoo search, which resolve the problem of random initialization in CS. Thereafter cuckoo search is executed for obtaining the optimum result and faster convergence.

Let there be *n* number of tweets which are to be clustered into N classes. Each tweet is represented by a feature vector having *S* number of features and each feature has been scaled in [0, *T*]. The probability distribution of each feature can be defined as follows (Mendenhall, Beaver, & Beaver, 2012; Saraswat, Arya, & Sharma, 2013):

$$p_i = \frac{O_i}{n}. \tag{2}$$

where *i* represents the ith feature value, i.e., $0 \le i \le T$, and O_i denotes the total number of tweets having i^{th} feature value. Moreover, the total mean of each feature is calculated using Eq. (3).
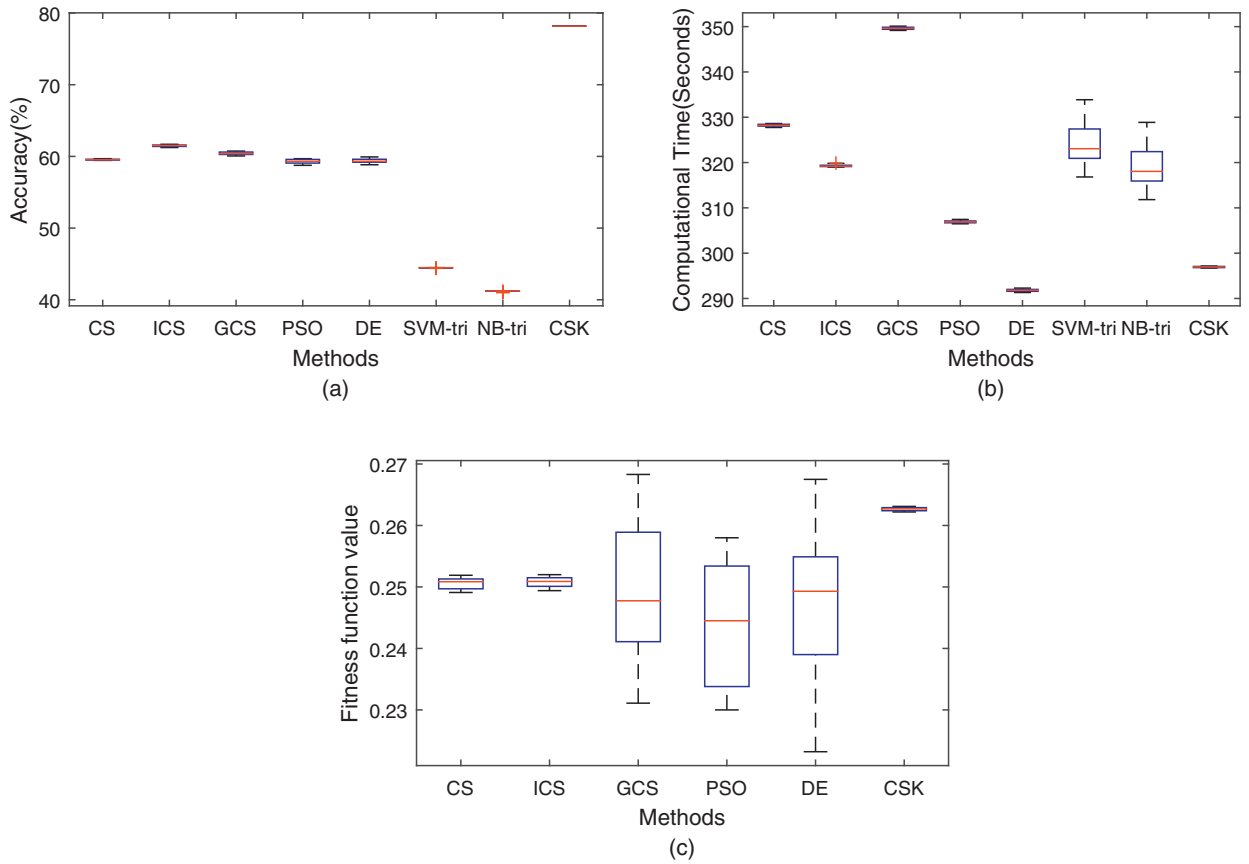
$$\mu = \sum_{i=1}^{T} ip_i. \tag{3}$$

Any tweet is classified into class D_j for which it has minimum Euclidean distance. Therefore, the probability (w_j) of occurrence of class D_j ($j = 1, 2.., N$) is given by Eq. (4).

$$w_j = \sum_{i \epsilon D_j} p_i \tag{4}$$

The mean of class D_j can be calculated by Eq. (5).

$$\mu_j = \sum_{i \epsilon D_j} ip_i / w_j. \tag{5}$$

Fig. 3. Box plots for all the considered methods and proposed method (CSK) for the performance parameter (a) accuracy, (b) computational time, (c) fitness function value of Testdata.manual.2009.06.14 dataset.

The inter-class variance can be generally defined as:

$$\sigma^2 = \sum_{j=1}^{N} w_j(\mu_j - \mu)^2. \tag{6}$$

To cluster the different tweets into their respective class, the inter-class variance shown in Eq. (6) should be maximized. Therefore, the objective function for the proposed hybrid cuckoo search method is to maximize the functions as defined in Eq. (6). The detailed steps of the proposed method is given in Algorithm 3.

Algorithm 3 Proposed method.

Set the size of population as *N*.
for $i = 1$ *to N* **do**
    Generate *k* clusters using the K-means algorithm.
    Use *k* cluster-heads to initialize the population of cuckoo search
end for
Calculate the fitness of these N solutions by using objective function
while $t < MaxGeneration$ **do**
    Generate *N* new solutions using Cuckoo Search
    Calculate the fitness of new solutions
    Remove the old solutions with better new solutions
    Replace the fraction (P_a) of worse solutions by random new solutions
end while
Print the best solution and its fitness
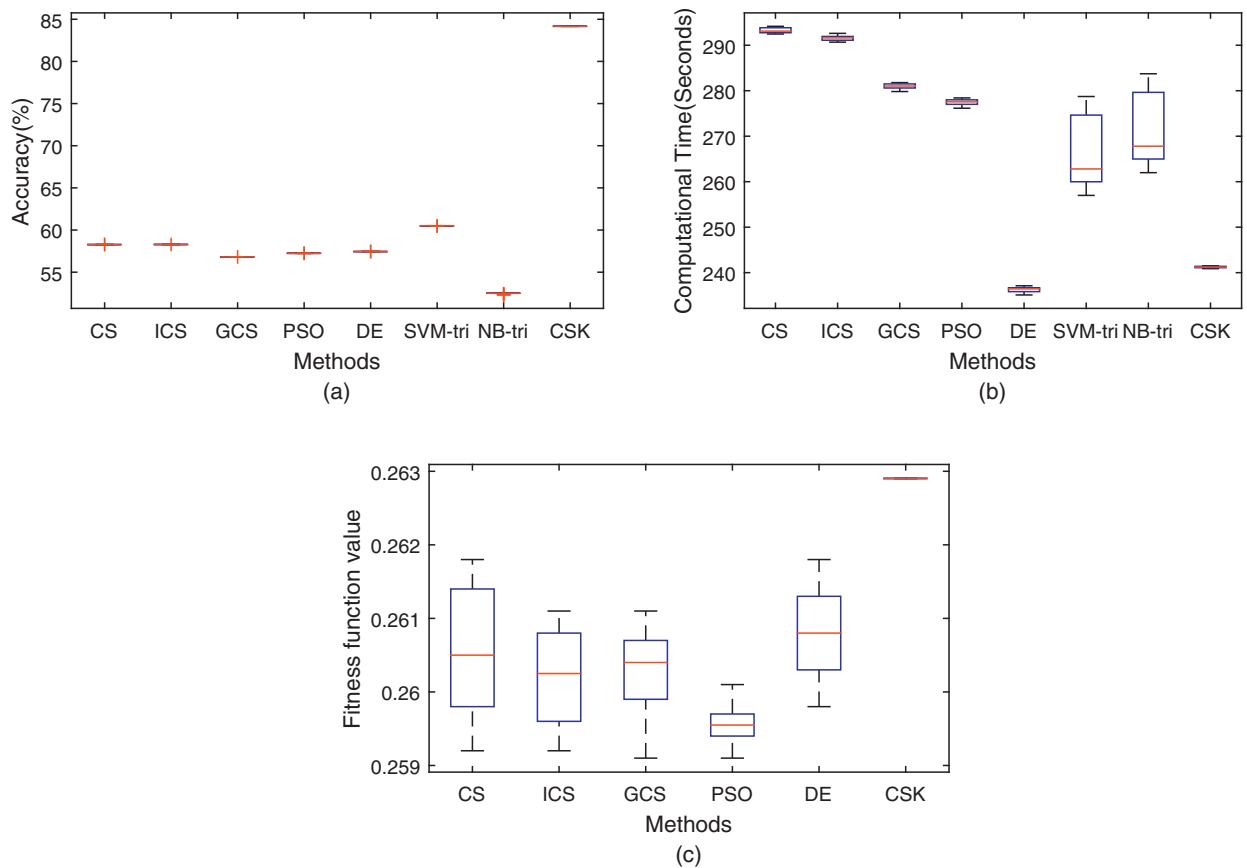
Fig. 4. Box plots for all the considered methods and proposed method (CSK) for the performance parameter (a) accuracy, (b) computational time, (c) fitness function value of Twitter-sanders-apple2 dataset.

4. Experimental results

The accuracy and efficiency of the proposed data clustering method (CSK) has been tested on the following four Twitter datasets containing tweets on different topics. The brief description of all the considered datasets has also been depicted in Table 1.

4.1. Testdata.manual.2009.06.14

This dataset has been taken from Stanford Twitter corpus (Testdata, 2015) and contains 1.6 million automatically annotated tweets which are sub-divided into training and testing datasets. In this paper, testing dataset has been used which contains 498 tweets having 182, 177, 139 positive, negative, and neutral tweets respectively. All the tweets in dataset are collected from May 11, 2009 to Jul 14, 2009 and are based on different topics such as Google, Obama, Kindle, China, north Korea, Iran, San Francisco, dentist, insects, and Nike. In this dataset 0 has been used for negative, 2 for neutral and 4 for positive polarity.

4.2. Twitter-sanders-apple

Sanders Analytics (Twitter-sanders-apple, 2015) have collected the following two datasets from Oct 15, 2011 to Oct 20, 2011 for Apple Corporation on four different topics namely; Apple, Google, Microsoft, and Twitter. Each tweet was manually annotated to either positive, negative, or neutral by Niek Sanders. Any tweet which contains positive indicator or topic is considered as positive tweets. Those tweets which neither have positive nor negative indicators, or have mixed positive and negative indicators, or have simple factual statements, or have questions with no strong emotions are considered as neutral tweets. Tweets with negative indicator or topic are classified as negative tweets (Carstens, 2016). In dataset, positive tweets are represented by ′*pos*′, negative tweets by ′*neg*′, and neutral by ′*neut*′.
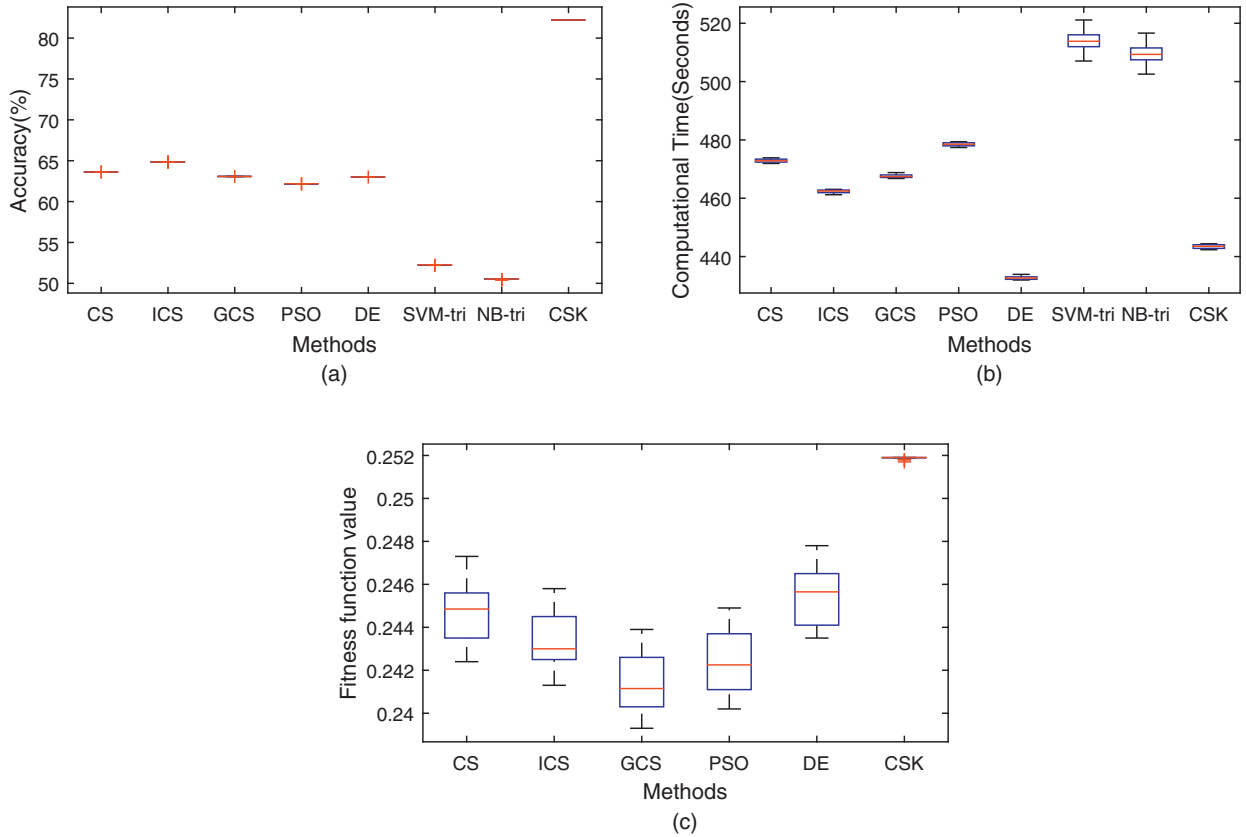
Fig. 5. Box plots for all the considered methods and proposed method (CSK) for the performance parameter (a) accuracy, (b) computational time, (c) fitness function value of Twitter-sanders-apple3 dataset.

4.2.1. Twitter-sanders-apple2

This dataset (Twitter-sanders-apple, 2015) is a subset of Twitter-sanders-apple and consists of 479 tweets. There are 163 positive and 316 negative tweets in the given dataset.

4.2.2. Twitter-sanders-apple3

This dataset is also a subset of Twitter-sanders-apple (Twitter-sanders-apple, 2015) and contains 988 tweets. It has three classes having 163 positive, 316 negative, and 509 neutral tweets.
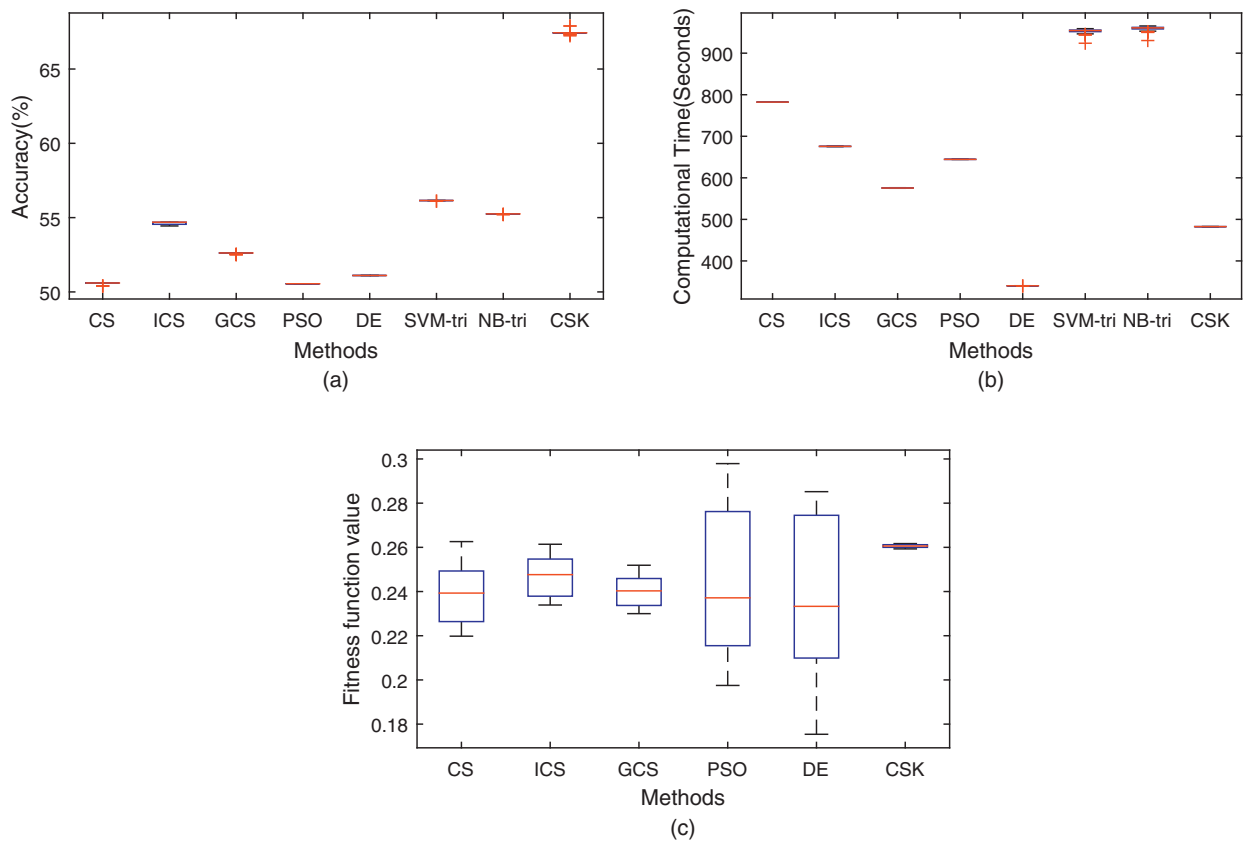
4.3. Twitter dataset

The Twitter dataset (twi, 2014) has been taken from Twitter which is based on the topics of sports, saints, funny images, jokes, and college students. This dataset has 2000 tweets posted from No. 17, 2014 to Dec 10, 2014. The considered dataset is manually labeled in two classes namely; positive and negative, each containing 1000 tweets. In dataset, positive tweets are represented by 1 and negative tweets by 0.

The Twitter dataset has been preprocessed to remove the undesired words and characters as discussed in Section 3.1. From the preprocessed dataset, 11 features have been extracted as shown in Table 2 along with their mean and standard deviation values for each dataset. The statistical mean shows the central tendency of each dataset. From the table, it is observed that each dataset is unbiased and contains different types of words which may affect the clustering accuracy. Further, standard deviation shows that each feature has sufficient variation in tweets.

Moreover, the proposed method has been compared with seven existing methods namely; two word-level n-grams (support vector machine-trigram (SVM-tri) and naive Bayes-trigram (NB-tri)), cuckoo search (CS), improved cuckoo search (ICS), gauss distribution-based cuckoo search (GCS), particle swarm optimization (PSO), and differential evolution (DE). The considered n-grams are weighted using term frequency (tf) and the value of n has been selected using cross-validation (rotation estimation) (Kohavi et al., 1995). The parameter settings for all the considered methods have been presented in Table 3.

To measure the performance of the proposed method, three parameters have been considered namely; accuracy, computational time, and fitness function value. Table 4 shows the comparative results of the proposed method and existing considered methods in terms of all the above three parameters. For fair comparison, each method has been executed 30

Fig. 6. Box plots for all the considered methods and proposed method (CSK) for the performance parameter (a) accuracy, (b) computational time, (c) fitness function value of Twitter dataset.

times and Table 4 represents the mean values of accuracy, computational time and fitness function values. From the table, it is visualized that the proposed method gives the best accuracy among all the considered methods. Moreover, the proposed method also outperforms in the mean fitness function value. Further, the proposed method is computationally efficient as compared to other existing methods except DE method. However, the main concern is the accuracy of the system, where proposed method outperformed.
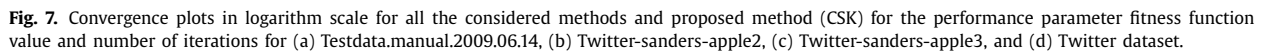
To test the significant difference between the proposed method and considered methods, a statistical comparison is performed for accuracy, computational time, and fitness function value using student's *t*-test (Owen, 1965) with a confidence level of 95%. In this experiment, student's *t*-test is applied for the null hypothesis that there is no significant difference in the parameter values for 30 runs with respect to proposed method and existing methods. The results of *t*-test are demonstrated in Tables 5–7. The results indicate that there are better significant differences between the proposed method and considered existing methods for all the considered parameters i.e., the null hypothesis is rejected except DE method which shows better computational efficiency.

Moreover, to compare the performance of all the considered methods and proposed method, boxplot analysis (McGill, Tukey, & Larsen, 1978) is carried out. The boxplot graphically represents the empirical distribution of the data. The boxplot for existing and proposed methods are shown in Figs. 3–6. In the boxplot, the x-axis represents the name of the methods and the corresponding parameters under consideration on the y-axis. From the boxplots, it is observed that proposed method gives the better and consistent results for all the considered performance parameters except computational time where DE outperforms.

To show the convergence behavior of all the considered methods and proposed method convergence plot have also been plotted in Fig. 7. In the convergence plot, the x and y-axis represent the number of iterations and fitness function values respectively. From the convergence plots, it is observed that proposed method converges quickly as compared to all the considered methods and gives the better results.

5. Conclusion

In this paper, a novel hybrid clustering method (CSK) has been introduced to analyze the sentiments of tweets using K-means and CS method. The proposed method modifies the random initialization process of CS by the solutions obtained

Fig. 7. Convergence plots in logarithm scale for all the considered methods and proposed method (CSK) for the performance parameter fitness function value and number of iterations for (a) Testdata.manual.2009.06.14, (b) Twitter-sanders-apple2, (c) Twitter-sanders-apple3, and (d) Twitter dataset.

from K-means which enhances its performance. The method has been tested on four Twitter datasets. Further, the method has also been compared with CS, ICS, GCS, PSO, DE, SVM-tri, and NB-tri. For the better comparison, Student's *t*-test, box plot, and convergence plot analysis have also been performed for all the considered datasets. From the experimental and statistical results, effectiveness of the proposed method has been observed.

However, the proposed method shows better accuracy as compared to existing methods, improvement in accuracy is still desired. Therefore, further work will include to explore the possibilities of accuracy improvement by introducing some feature selection method and applying different variants of optimization methods. Moreover, there is a scope of improvement for dealing with sarcasm and irony tweets. Further, word level and post level contextual information along with domain specific ontology can also be considered for the classification of the tweets.

Acknowledgments

Appendix A. Regular expression

1. Regular expression to replace URLs from a tweet with string url

```
tweet = re.sub('((www\.[^\s]+)|(https?://[^\s]+))','url',tweet)
```

2. Regular expression to remove @username from tweet

```
tweet = re.sub('(?<=^|(?<=[^a-zA-Z0-9-_.]))@([A-Za-z]+[A-Za-z0-9]+)','', tweet)
```

3. Regular expression to remove additional white spaces from tweet

```
tweet = re.sub('[\s]+', ' ', tweet)
```

4. Regular expression to replace #word with word in tweet

```
tweet = re.sub(r'#([^\s]+)', r'\1', tweet)
```

5. Regular expression to strip punctuations from a word

```
tweet = tweet.strip('-' '()' '\'  '/')
```

References

Acronym dictionary, (2015). www.netlingo.com/acronyms.php.

Agarwal, B., Mittal, N., Bansal, P., & Garg, S. (2015). Sentiment analysis using common-sense and context information. *Computational Intelligence and Neuroscience, 2015*, 30.

Altınel, B., & Ganiz, M. C. (2016). A new hybrid semi-supervised algorithm for text classification with class-based semantics. *Knowledge-Based Systems, 108*, 50–64. 2016

Appel, O., Chiclana, F., Carter, J., & Fujita, H. (2016). A hybrid approach to the sentiment analysis problem at the sentence level. *Knowledge-Based Systems*.

Asur, S., Huberman, B., et al. (2010). Predicting the future with social media. In *International conference on web intelligence and intelligent agent technology (WI-IAT): 1* (pp. 492–499). IEEE.

Basari, A. S. H., Hussin, B., Ananta, I. G. P., & Zeniarja, J. (2013). Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization. *Procedia Engineering, 53*, 453–462.

Bello-Orgaz, G., Menéndez, H. D., & Camacho, D. (2012). Adaptive k-means algorithm for overlapped graph clustering. *International Journal of Neural Systems, 22*.

Bharti, S., Vachha, B., Pradhan, R., Babu, K., & Jena, S. (2016). Sarcastic sentiment detection in tweets streamed in real time: A big data approach. *Digital Communications and Networks, 2*(3), 108–121.

Bharti, S. K., Babu, K. S., & Jena, S. K. (2015). Parsing-based sarcasm sentiment recognition in twitter data. In *2015 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)* (pp. 1373–1380). IEEE.

Boiy, E., Hens, P., Deschacht, K., & Moens, M.-F. (2007). Automatic sentiment analysis in on-line text. In *ELPUB* (pp. 349–360).

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science, 2*, 1–8.

Bravo-Marquez, F., Frank, E., & Pfahringer, B. (2016). Building a twitter opinion lexicon from automatically-annotated tweets. *Knowledge-Based Systems*.

Bravo-Marquez, F., Mendoza, M., & Poblete, B. (2013). Combining strengths, emotions and polarities for boosting twitter sentiment analysis. In *Proceedings of the second international workshop on issues of sentiment discovery and opinion mining* (p. 2). ACM.

Brown, C. T., Liebovitch, L. S., & Glendon, R. (2007). Lévy flights in dobe ju/hoansi foraging patterns. *Human Ecology, 35*, 129–138.

Cambria, E. (2016). Affective computing and sentiment analysis. *IEEE Intelligent Systems, Vol. 31*, 102–107.

Canuto, S., Gonçalves, M. A., & Benevenuto, F. (2016). Exploiting new sentiment-based meta-level features for effective sentiment analysis. In *Proceedings of the ninth ACM international conference on web search and data mining* (pp. 53–62). ACM.

Carstens, L. (2016). Using argumentation to improve classification in natural language problems.

Carvalho, P., Sarmento, L., Silva, M. J., & De Oliveira, E. (2009). Clues for detecting irony in user-generated contents: oh... !! it's so easy;-). In *Proceedings of the 1st international CIKM workshop on topic-sentiment analysis for mass opinion* (pp. 53–56). ACM.

Chen, T., Xu, R., He, Y., Xia, Y., & Wang, X. (2016). Learning user and product distributed representations using a sequence model for sentiment analysis. *IEEE Computational Intelligence Magazine, 11*(3), 34–44.

Chiang, M. M.-T., & Mirkin, B. (2010). Intelligent choice of the number of clusters in k-means clustering: An experimental study with different cluster spreads. *Journal of Classification, 27*(1), 3–40.

Coletta, L. F. S., da Silva, N. F. F., Hruschka, E. R., & Hruschka, E. R. (2014). Combining classification and clustering for tweet sentiment analysis. In *Intelligent systems (BRACIS), 2014 Brazilian conference on* (pp. 210–215). IEEE.

Danielsson, P.-E. (1980). Euclidean distance mapping. *Computer Graphics and Image Processing, 14*, 227–248.

Emoticon dictionary, (2015). http://www.netlingo.com/smileys.php.

Exclamation word dictionary, (2015). http://www.vidarholen.net/contents/interjections/.

Fernández-Gavilanes, M., Álvarez-López, T., Juncal-Martínez, J., Costa-Montenegro, E., & González-Castaño, F. J. (2016). Unsupervised method for sentiment analysis in online texts. *Expert Systems with Applications, 58*, 57–75.

González-Ibáñez, R., Muresan, S., & Wacholder, N. (2011). Identifying sarcasm in twitter: A closer look. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies: short papers-volume 2* (pp. 581–586). Association for Computational Linguistics.

Gupta, D. K., Reddy, K. S., Ekbal, A., et al. (2015). Pso-asent: Feature selection using particle swarm optimization for aspect based sentiment analysis. In *Natural language processing and information systems* (pp. 220–233). Springer.

Haddi, E., Liu, X., & Shi, Y. (2013). The role of text pre-processing in sentiment analysis. *Procedia Computer Science, 17*, 26–32.

Howard, P. N. (2011). The Arab springs cascading effects. *Pacific Standard, 23*.

Hu, X., Tang, J., Gao, H., & Liu, H. (2013a). Unsupervised sentiment analysis with emotional signals. In *Proceedings of the 22nd international conference on world wide web* (pp. 607–618). ACM.

Hu, X., Tang, L., Tang, J., & Liu, H. (2013b). Exploiting social relations for sentiment analysis in microblogging. In *Proceedings of the sixth ACM international conference on web search and data mining* (pp. 537–546).

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recognition Letters, 31*(8), 651–666.

jeffreybreen (2015). Positive word dictionary. twitter-sentiment-analysis-tutorial-201107/data/opinion-lexicon-English/positive-words.txt. 2011 (accessed December 15.

jeffreybreen (2015). Negative word dictionary. twitter-sentiment-analysis-tutorial-201107/data/opinion-lexicon-English/Negative-words.txt. accessed December 15.

Joshi, A., Bhattacharyya, P., & Carman, M. J. (2016). Automatic sarcasm detection: A survey. arXiv preprint arXiv:1602.03426,.

Joshi, A., Sharma, V., & Bhattacharyya, P. (2015). Harnessing context incongruity for sarcasm detection. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing: 2* (pp. 757–762).

Kanakaraj, M., & Guddeti, R. M. R. (2015). Nlp based sentiment analysis on twitter data using ensemble classifiers. In *Signal processing, communication and networking (ICSCN), 2015 3rd international conference on* (pp. 1–5). IEEE.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Neural Networks, 4*, 1942–1948.

Khan, A. Z., Atique, M., & Thakare, V. (2015). Combining lexicon-based and learning-based methods for twitter sentiment analysis. *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)*, 89.

Kogan, J., Teboulle, M., & Nicholas, C. (2005). Data driven similarity measures for k-means like clustering algorithms. *Information Retrieval, 8.*, 331–349.

Kohavi, R., et al. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai: 14* (pp. 1137–1145).

Kontopoulos, E., Berberidis, C., Dergiades, T., & Bassiliades, N. (2013). Ontology-based sentiment analysis of twitter posts. *Expert Systems with Applications, 40*, 4065–4074.

Kranjc, J., Smailović, J., Podpečan, V., Grčar, M., Žnidaršič, M., & Lavrač, N. (2015). Active learning for sentiment analysis on data streams: Methodology and workflow implementation in the clowdflows platform. *Information Processing & Management, 51*, 187–203.

Liu, B., Hu, M., & Cheng, J. (2005). Opinion observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th international conference on world wide web* (pp. 342–351).

McGill, R., Tukey, J. W., & Larsen, W. A. (1978). Variations of box plots. *The American Statistician, 32*, 12–16.

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal, 5*(4), 1093–1113.

Mendenhall, W., Beaver, R., & Beaver, B. (2012). *Introduction to probability and statistics*. Cengage Learning.

Mohammad, S. M., Zhu, X., Kiritchenko, S., & Martin, J. (2015). Sentiment, emotion, purpose, and style in electoral tweets. *Information Processing & Management, 51*(4), 480–499.

Muhammad, A., Wiratunga, N., & Lothian, R. (2016). Contextual sentiment analysis for social media genres. *Knowledge-Based Systems*.

Nugent, R., Dean, N., & Ayers, E. (2010). Skill set profile clustering: The empty k-means algorithm with automatic specification of starting cluster centers. *Educational data mining 2010*.

Owen, D. (1965). The power of student's *t*-test. *Journal of the American Statistical Association, 60*, 320–333.

Pandarachalil, R., Sendhilkumar, S., & Mahalakshmi, G. (2015). Twitter sentiment analysis for large-scale data: An unsupervised approach. *Cognitive Computation, 7*, 254–262.

Pavlyukevich, I. (2007). Lévy flights, non-local search and simulated annealing. *Journal of Computational Physics, 226*, 1830–1844.

Psychological feelings, (2015). http://www.psychpage.com/learning/library/assess/feelings.html.

Qiu, G., Liu, B., Bu, J., & Chen, C. (2009). Expanding domain sentiment lexicon through double propagation. In *IJCAI: 9* (pp. 1199–1204).

Reyes, A., Rosso, P., & Buscaldi, D. (2012). From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering, 74*, 1–12.

Ritter, A., Clark, S., Etzioni, O., et al. (2011). Named entity recognition in tweets: an experimental study. In *Proceedings of international conference on empirical methods in natural language processing* (pp. 1524–1534).

Saif, H., He, Y., Fernandez, M., & Alani, H. (2016a). Contextual semantics for sentiment analysis of twitter. *Information Processing & Management, 52*(1), 5–19. 2016

Saif, H., Ortega, F. J., Fernández, M., & Cantador, I. (2016b). Sentiment analysis in social streams, 2016,.

Sarcasm, (2016). http://examples.yourdictionary.com/examples-of-sarcasm.html.

Saraswat, M., Arya, K., & Sharma, H. (2013). Leukocyte segmentation in tissue images using differential evolution algorithm. *Swarm and Evolutionary Computation, 11*, 46–54.

Shah, R. R., Yu, Y., Verma, A., Tang, S., Shaikh, A. D., & Zimmermann, R. (2016). Leveraging multimodal information for event summarization and concept-level sentiment analysis. *Knowledge-Based Systems*.

Stopwords dictionary, (2015). http://www.ranks.nl/stopwords.

Storn, R., & Price, K. (1997). Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11*, 341–359.

Sulis, E., Farías, D. I. H., Rosso, P., Patti, V., & Ruffo, G. (2016). Figurative messages and affect in twitter: Differences between# irony,# sarcasm and# not. *Knowledge-Based Systems*.

Tang, H., Tan, S., & Cheng, X. (2009). A survey on sentiment detection of reviews. *Expert Systems with Applications, 36*, 10760–10773.

Thet, T. T., Na, J.-C., & Khoo, C. S. (2010). Aspect-based sentiment analysis of movie reviews on discussion boards. *Journal of Information Science*.

Testdata.manual.2009.06.14, (2015). http://help.sentiment140.com/for-students/.

Twitter dataset, (2014). https://drive.google.com/file/d/0BwPSGZHAP_yoN2pZcVl1Qmp1OEU/view?usp=sharing.

Twitter-sanders-apple, (2015). http://boston.lti.cs.cmu.edu/classes/95-865-K/HW/HW3/.

Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management, 50*, 104–112.

Valian, E., Mohanna, S., & Tavakoli, S. (2011). Improved cuckoo search algorithm for feedforward neural network training. *International Journal of Artificial Intelligence & Applications, 2*, 36–43.

Wilkinson, R., & Hingston, P. (1991). Using the cosine measure in a neural network for document retrieval. In *Proceedings of the 14th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 202–210). ACM.

Xia, R., Xu, F., Yu, J., Qi, Y., & Cambria, E. (2016). Polarity shift detection, elimination and ensemble: A three-stage model for document-level sentiment analysis. *Information Processing & Management, 52*, 36–45.

Yang, X.-S., & Deb, S. (2009). Cuckoo search via lévy flights. In *World congress on nature & biologically inspired computing* (pp. 210–214). IEEE.

Yokoyama, S., Nakayama, A., & Okada, A. (2009). One-mode three-way overlapping cluster analysis. *Computational Statistics, 24*, 165–179.

Yusof, N. N., Mohamed, A., & Abdul-Rahman, S. (2015). Reviewing classification approaches in sentiment analysis. In *International conference on soft computing in data science* (pp. 43–53). Springer.

Žalik, K. R. (2008). An efficient k'-means clustering algorithm. *Pattern Recognition Letters, Vol. 29*, 1385–1391.

Zheng, H., & Zhou, Y. (2012). A novel cuckoo search optimization algorithm based on Gauss distribution. *Journal of Computational Information Systems, 8*, 4193–4200.

Zhu, J., Wang, H., & Mao, J. (2010). Sentiment classification using genetic algorithm and conditional random fields. In *2nd IEEE international conference on information management and engineering (ICIME)* (pp. 193–196). IEEE.