IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 11, 2020, accepted February 28, 2020, date of publication March 27, 2020, date of current version April 22, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2983859

A Cooperative Binary-Clustering Framework Based on Majority Voting for Twitter Sentiment Analysis

MARYUM BIBI[1], WAJID AZIZ[2], MAJID ALMARAASHI[2], IMTIAZ HUSSAIN KHAN[3], MALIK SAJJAD AHMED NADEEM[1], AND NAZNEEN HABIB[1]

[1]Department of Computer Science and IT, The University of Azad Jammu & Kashmir, City Campus, Muzaffarabad 13100, Pakistan
[2]College of Computer Sciences and Engineering, University of Jeddah, Jeddah 21959, Saudi Arabia
[3]Department of Computer Science, King Abdulaziz University, Jeddah 21589, Saudi Arabia

Corresponding author : Wajid Aziz (wloun@uj.edu.sa)

ABSTRACT Twitter sentiment analysis is a challenging problem in natural language processing. For this purpose, supervised learning techniques have mostly been employed, which require labeled data for training. However, it is very time consuming to label datasets of large size. To address this issue, unsupervised learning techniques such as clustering can be used. In this study, we explore the possibility of using hierarchical clustering for twitter sentiment analysis. Three hierarchical-clustering techniques, namely single linkage (SL), complete linkage (CL) and average linkage (AL), are examined. A cooperative framework of SL, CL and AL is built to select the optimal cluster for tweets wherein the notion of *optimal-cluster selection* is operationalized using majority voting. The hierarchical clustering techniques are also compared with k-means and two state-of-the-art classifiers (SVM and Naïve Bayes). The performance of clustering and classification is measured in terms of accuracy and time efficiency. The experimental results indicate that cooperative clustering based on majority voting approach is robust in terms of good quality clusters with tradeoff of poor time efficiency. The results also suggest that the accuracy of the proposed clustering framework is comparable to classifiers which is encouraging.

INDEX TERMS Cooperative clustering, majority voting, sentiment analysis, twitter sentiment analysis.

I. INTRODUCTION

Sentiment analysis has recently gained considerable popularity in different fields [1]–[6]. Companies perform sentiment analysis to examine feedback on products, government and other agencies use it for public-health monitoring and predicting political trends, and so on. Prior to the emergence of social networks, manual mechanisms were usually employed for this purpose. Companies used to manually analyze the popularity of their products by surveying customers. However, with the advent of social networks, e.g., *twitter*, manual analysis of data has become a challenging problem. Twitter is a popular microblogging platform that allows users to share their ideas, opinions and thoughts through real-time short messages (limited to 280 characters) called tweets. Researchers have explored twitter data for diverse issues including sentiment analysis [1], [7]–[15], public-health monitoring [16]–[19], election trends [20], [21], education [22] and sports [23]. People normally make spelling mistakes and use slang in tweets that pose significant challenges for twitter sentiment analysis [7]. Therefore, it is imperative to use intelligent techniques to extract useful knowledge from twitter data.

Machine learning techniques can be used to extract useful information from such noisy data generated on daily basis [24]. These techniques have largely been applied in diverse domains including banking [25], bio-informatics [26] and social media [7], [9]. Supervised learning uses labeled data to build a classification model, which is subsequently used to predict class labels for (unlabeled) test data. Supervised learning techniques have extensively been used for sentiment analysis [7], [10], [27]–[30]. The limitation of such techniques, however, is the requirement of labeled data. On the other hand, unsupervised learning techniques,

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

e.g., clustering, tend to group unlabeled data based on similarity. Clustering techniques are further divided into hierarchical and partitioned clustering. Hierarchical clustering recursively constructs clusters of given instances as dendrograms either in a top-down (a.k.a. divisive clustering) or bottom-up (a.k.a. agglomerative clustering) manner using some similarity or distance measure. The latter are further divided into single linkage (SL), complete linkage (CL) and average linkage (AL). Partitioned (a.k.a. flat) approach creates partitions of instances by relocating them from one cluster to another according to some criteria, e.g., minimizing the sum of square errors. One widely used partitioned clustering method for sentiment analysis is k-means clustering [31]. Other than individual clustering techniques, multiple clustering techniques can be combined to produce better quality results [32]. One such technique is *cooperative clustering* that combines different clustering approaches. Ensemble/cooperative methods provide more accurate and robust solutions in comparison with individual techniques [33]. Cooperative clustering has largely been explored in various domains including software modularization [34], [35] and pattern recognition [36], text classification [37].

Literature on sentiment analysis suggests that researchers have paid little attention to using unsupervised learning techniques in this area. Recently, few researchers have proposed unsupervised learning (or a combination of supervised and unsupervised learning) techniques for sentiment analysis [31], [38], [39]. Even though k-means clustering has previously been used for sentiment analysis [31], its performance is not thoroughly reported in terms of time efficiency. Therefore, it is unclear how scalable k-means is for sentiment analysis.

In this study, we use three hierarchical-clustering techniques (SL, CL and AL) to create a cooperative-clustering framework in a novel manner for twitter sentiment analysis. The cooperative framework selects the optimal cluster for a given tweet based on majority voting. Although we investigated traditional hierarchical methods to design the framework for Twitter sentiment analysis, yet these are popular among the research community. For example, these techniques have been investigated in recent times even during years 2018 and 2019 [40]–[42]. Novelty of the present study stems from the fact that a) hierarchical clustering is investigated first time thoroughly for (Twitter) sentiment analysis, and b) first time an ensemble of clustering techniques is created which achieves comparable performance to the widely studied classification techniques. The performance of hierarchical clustering techniques is compared with k-means and two widely studied classifiers, Naïve Bayes and support vector machines (SVM). All these algorithms are evaluated in terms of *authoritativeness (high quality clusters and classification results)* and time efficiency. Authoritativeness of a clustering algorithm is the notion of how closely its results match to the results of some authority (e.g. human experts). To show authoritativeness, a well known accuracy metric as reported in [24] has been used. The experiments are carried out on three publicly available datasets and one indigenous dataset NewTweets (collected for this study using Twitter4j API[1]). Unigram, TF-IDF and polarity based are used for feature representation. These features have previously been used in various studies for twitter sentiment analysis using supervised learning [43], [44]. The results of our empirical study suggest that the accuracy of the proposed clustering framework is comparable to classifiers. These results suggest that clustering techniques can be used for (twitter) sentiment analysis, without having a need of large size labeled data to train a classifier. Briefly, the work presented here has four main contributions: a) hierarchical clustering techniques are thoroughly explored for sentiment analysis, b) a novel cooperative-clustering framework based on majority voting approach is proposed for sentiment analysis, c) a sizeable indigenous twitter dataset (NewTweets) annotated by medical-domain specialists is created that can be used for further research, and d) it is empirically shown that unsupervised-learning techniques can achieve comparable performance to supervised learning.

II. RELATED WORK

In this section, the literature relevant to sentiment analysis using supervised learning techniques and unsupervised learning techniques is presented. In a recent study, deep neural network is experimented for targeted aspect-based sentiment analysis by integrating common sense knowledge in the network [45]. Experiments conducted on SentiHood, and SemEval 2015 dataset revealed encouraging results. In another recent study, [46] used a novel neural network design for formalizing sentiment information into market views. They built an ensemble of evolving clustering and long short-term memory. Experimental evaluation on opinion messages from StockTwits suggests that the proposed framework outperforms the existing forecasting techniques. In another study, [47] discovered sentiment polarity from short-video clips using deep convolutional neural network for feature extraction. They conducted experiments using SVM and reported better performance. In an earlier study [7], authors built models using Naïve Bayes, SVM and maximum entropy for sentiment classification of twitter data. They collected dataset using Twitter API; their experiments revealed that SVM outperforms other classifiers. In another study [27], tweets were assigned sentiment polarity using multinomial Naïve Bayes, conditional random fields and SVM, where Naïve Bayes offered better performance. In another work [28], a cooperative framework of Naïve Bayes, random forest, SVM and logistics regression is designed for classification of positive and negative tweets. Experiments were conducted on four twitter datasets and it was found that the proposed framework performs better as compared to the individual classifiers. [10] performed sentiment classification using ensemble classifiers. Naïve Bayes, SVM, k-nearest neighbor and C4.5 algorithm are used for

[1] http://twitter4j.org/en/

this purpose. Experimental results on three well-known twitter datasets showed better accuracy for ensemble classifiers. Recently, Naïve Bayes and SVM have been investigated for Twitter sentiment analysis [48] in which three publicly available Twitter datasets were considered for experimentation. Better performance for both techniques is reported on the proposed algorithm CAARIA. In a study, [29] combined lexical-based techniques and machine-learning techniques for sentiment analysis of Facebook data in an e-Learning domain. They implemented their method in SentBuk and reported promising results. Hate crimes targeting minorities have been addressed in [49] in which Donald Trump's tweets are considered and shown as highly correlated with anti-Muslim hate crimes. In [50], an algorithm is proposed based on sentiment diffusion patterns for Twitter sentiment analysis. Experimental evaluation on real-world datasets reveals better performance of proposed algorithm as compared to the state-of-the-art textual information methods.

Among unsupervised learning techniques, k-means clustering has widely been explored for sentiment analysis of twitter data [31]. k-means uses a moving centroid approach wherein cluster's center moves in each iteration to minimize error. [31] performed sentiment analysis on two widely explored twitter datasets using an unsupervised-learning framework. They used k-means clustering algorithm as a baseline and showed that the algorithm was not encouraging for sentiment analysis. Recently, hierarchical agglomerative clustering has been investigated in [40] on real time shopping data. Better performance of CL and Ward's method is reported. In [41], dependency between training methods and agglomerative hierarchical clustering has been investigated. For this purpose, a training algorithm has been designed that is well suited to agglomerative hierarchical clustering algorithms. Experimental evaluation showed improved results for the SL algorithm. Recently, cluster analysis is performed for classical portfolio selection. For this purpose, among other algorithms, CL clustering is also studied. Tweets from Taiwan during 2017 are analyzed to categorize into travel and non-travel classes. The authors integrated hierarchical clustering with deep neural network [51].

Cooperative methods aim to improve the performance of individual techniques by combining them using a particular ensemble method [32]. In literature, such techniques are proposed in various fields including software modularization. In a study [34], cooperative clustering has been used in order to perform software modularization, i.e., generating meaningful views of software systems. They performed experimental evaluation on five open source software systems and found that the proposed cooperative-clustering framework offers better performance. In [33], clustering ensemble is examined such that multiple clustering techniques are combined for a robust and stable solution. For this purpose an Iterative Combining Clusterings Method (ICCM) is proposed. It assigns the clusters to instances based on majority voting mechanism. Experiments are performed on two gene expression datasets and three real-life datasets. Detailed analysis revealed that proposed cooperative clustering algorithm outperformed. Another consensus cooperative-clustering technique based on software dependency graphs has been proposed in the area of software modularization [52]; promising results have been reported for this strategy after conducting experiments on six Java-based software systems. In another study [53], a distributed cooperative-clustering model has been developed for working on two-tier super-peer P2P. Authors observed better results by the cooperative framework. In another work [36], multiple clustering methods for pattern recognition are combined; impressive results on gene expression and text datasets were revealed. Recently, a novel selective-clustering ensemble is proposed [32] in which experiments are performed on 17 real datasets from UCI machine learning repository. In an interesting work, [54] used unlabeled texts and exploited lexical co-occurrence information to generate a direct acyclic graph of concepts. They used polarity score of initially known concepts in an incremental manner to compute polarity scores of new concepts.

In various research studies, a combination of supervised and unsupervised learning has also been investigated. In [38], sentiment analysis is performed on tweet data related to a product by using a hybrid approach based on k-means and supervised learning techniques. It is shown that this approach performs better in comparison to decision trees, SVM, random forests and logistic regression. Recently, some researchers proposed semi-supervised learning techniques based on statistical-learning theory for sentiment analysis [39]. Among these, [39] built an extreme learning machine model with a novel scalable approach and demonstrated its effectiveness for big (social) data analysis. The proposed framework is evaluated on a benchmark of 23244 common-sense concepts obtained through Sentic API.[2] In another study [55], sentiment analysis on movie reviews in Turkish language has been performed using different linguistic patterns.

III. MATERIALS AND METHODS

In the subsequent sections, the datasets, clustering and classification techniques (used for comparative analysis), feature selection methods and evaluation metrics used in the current study are discussed in turn.

A. TWITTER DATASETS

Four twitter datasets in English language are used for evaluating the selected algorithms. Three of these are existing datasets: Health Care Reform (HCR), Sentiment Strength Twitter Dataset (SS-Tweet), Stanford Twitter Sentiment Test Set (STS-Test). These datasets have widely been explored in previous studies [56], [57]. The fourth one is a newly collected indigenous dataset, which has been created as part of the current study. A brief description of these datasets is given below.

[2] http://sentic.net/api

TABLE 1. **Twitter datasets' statistics.**

Dataset	No. of tweets	Positive tweets	Negative tweets	No. of features
HCR	1922	541	1381	4600
SS-Tweet	2289	1252	1037	7921
STS-Test	359	182	177	1484
NewTweets	1500	703	797	4095

- HCR is a publicly available twitter dataset, which consists of 2156 manually-labeled (positive, negative and neutral) tweets [56]. In this study, we focussed only on positive and negative tweets, therefore, a subset of 1922 tweets is used (i.e., neutral tweets are excluded). The details of this dataset are given in Table 1.
- SS-Tweet was originally prepared for sentiment strength detection [58]. Tweets are labeled according to their positive and negative sentiment strength. Labels are then re-assigned to these tweets in [59]. This revised version of twitter dataset, which consists of 2289 tweets, is used in this study.
- STS-Test was collected for sentiment classification [7]. The original dataset consists of training and testing data; the former is labeled automatically while the latter is manually labeled. In this study we used only a subset of testing data, containing only positive and negative tweets. Although this test set is very small, it has been experimented in various studies [7], [56], [59].
- NewTweets, which is collected as part of the present study, comprises of 1500 tweets (Table 1). This dataset is collected based on keywords *flu* and *migraine* using Twitter4j API. It contains 703 positive and 797 negative tweets, which were manually labeled by medical-domain specialists. The novelty of NewTweets is that it is the first of its kind that has been labeled by medical-domain specialists.

B. K-MEANS CLUSTERING

k-means partitioned clustering has been explored for sentiment analysis of twitter data in numerous studies [31], [38]. This technique divides the given instances into k non-empty subsets. Depending on the value of k, it computes the initial centroid around which partitioning is performed. At the beginning, the centroids are selected randomly. Then, the distance between each instance and centroid is calculated. The instances are merged with centroids having the least distance (i.e. the nearest centroid). Distance can be calculated using different distance measures, e.g., Euclidean distance measure. After first iteration of k-means clustering, the mean value of each cluster is computed. The new mean values are now considered as centroids for each cluster. The process continues until instances do not change their clusters.

C. HIERARCHICAL CLUSTERING

Hierarchical clustering algorithms are broadly categorized into agglomerative and divisive clustering. Agglomerative clustering algorithms cluster data instances based on similarity. The process starts by considering every data point as singleton clusters and then merges similar data points in a bottom-up fashion until a single cluster is left containing all data points. However, clustering process can be stopped until a cutoff point (pre-defined number of clusters) is reached, e.g., the clustering process can stop when two clusters are left in the hierarchy. On the other hand, divisive clustering algorithms initially consider all data points as a big single cluster and then recursively split the cluster(s) in a top-down manner until some pre-specified termination condition is met as discussed above. In this study, three agglomerative hierarchical clustering algorithms are explored, which are described below.

- The SL (single linkage) algorithm starts by considering each instance as individual cluster and then merges the closest clusters together, i.e., the clusters having the least dissimilarity. The process continues until the specified criteria are met. Different distance measures can be used to find dissimilarity, e.g., Euclidean distance, Manhattan distance, etc. In this study, Euclidean distance measure is used as a starting point to explore the possibility of applying hierarchical clustering in (twitter) sentiment analysis. Euclidean distance is computed between two instances (i.e., tweets in our case) X and Y using Equation (1).

$$d(X, Y) = \sqrt{\sum_{i=1}^{n}(X_i - Y_i)^2} \tag{1}$$

  The minimum distance (d_{min}) between clusters C_i and C_j is computed using the relation: $d(C_i, C_j) = d_{min}(x_{ip}, x_{jq})$, where x_{ip} is the instance p in cluster C_i and x_{jq} is the instance q in cluster C_j.
- The CL (complete linkage) algorithm clusters those instances that are furthest away from each other. In CL, the distance between two clusters is considered as the longest distance from any data point of one cluster to any data point in the other cluster i.e. $d(Ci, Cj) = d_{max}(x_{ip}, x_{jq})$. The clustering process is repeated until some specified criteria are met.
- The AL (average linkage) algorithm computes the average distance among the elements in two clusters, i.e., $d(Ci, Cj) = d_{avg}(x_{ip}, x_{jq})$. Based on average values, closest clusters are merged by computing distance using Equation (2).

$$D = 1/kl \sum_{i=1}^{k}\sum_{j=1}^{l} d(Xi, Yj) \tag{2}$$

D. COOPERATIVE CLUSTERING

Cooperative clustering has gained much popularity in recent years. In this approach, the strengths of multiple techniques are combined together to improve the overall accuracy as compared to individual techniques. The notion of cooperative clustering refers to using multiple clustering techniques in many different ways. For example, clustering techniques can be combined in a cascading manner in which output of one clustering algorithm is given as input to the other algorithm. This kind of clustering is known as hybrid clustering [53]. On the other hand, voting merging method considers a consensus of multiple clustering techniques in order to assign an instance to a cluster based on some consensus function. An instance (tweet in our case) will be assigned to a cluster with maximum votes [60]. Suppose there are four tweets ($tweet_1$, $tweet_2$, $tweet_3$, $tweet_4$) in a dataset and there are three clustering algorithms A, B and C. Let us assume that both A and B have placed $tweet_1$ in a cluster C_1 and C has placed it in another cluster, say C_2. Then, based on the majority votes, C_1 would be selected as the optimal cluster for $tweet_1$. In this study, we built the cooperative clustering framework by combining SL, CL and AL using majority voting. That is, the process starts by taking each tweet in turn. Each algorithm places the tweet in one of the two clusters (i.e., the cluster of positive tweets or the cluster of negative tweets). Then, in the second step, the selection method is invoked which places the tweet in the optimal cluster based on the majority votes.

E. CLASSIFIERS

In this study, two well-known classifiers, Naïve Bayes and SVMs (support vector machines), are experimented for comparative analysis. These are widely adopted for sentiment analysis [48], [61]–[65]. Bayesian classification is a probability based supervised learning technique, which aims to predict the class label for unseen data. It infers the class label by computing the probabilities of unseen instances. Posterior probability is the conditional probability computed when relevant evidence is seen. The class with the maximum posterior probability is assigned to an instance. The posterior probability is calculated using the following mathematical relation as shown in Equation (3):

$$p(H|X) = \frac{p(X|H)p(H)}{p(X)} \tag{3}$$

where X and H represent a tweet and class (positive/negative), respectively, in our case. $p(H|X)$ represents posterior probability of H conditioned on X while $p(X|H)$ shows posterior probability for X conditioned on H. In addition, $p(H)$ shows prior probability for H and $p(X)$ shows prior probability for X. $p(X)$ is constant therefore, only $p(H|X)$ is to be maximized. To assign the sentiment label, the features of tweets are used to compute the posterior probabilities for positive and negative classes.

SVMs builds non-linear classification models from the training data to predict the class of unseen instances. SVMs find a separating hyperplane by transforming the original data into higher dimensionality. In this study, we used the relationship $w.x + b = 0$ for separating the hyperplane for a two-class classifier, where x, w and b show training instances (tweets in our case), weight vector and bias, respectively. The hyperparameters and the kernel used can impact the performance of SVM, in this research, we used Linear kernel with default parameters.

TABLE 2. Unigram representation of features.

	f_1	f_2	f_3	f_4	f_5
$tweet_1$	1	1	1	0	0
$tweet_2$	0	1	1	0	0
$tweet_3$	0	0	0	1	1
$tweet_4$	1	0	1	1	0

F. FEATURE SELECTION AND REPRESENTATION

Since twitter datasets consist of English language text, text-mining based features can be useful for representation of tweets in such a way that clustering techniques can be applied. In this study, three widely studied [66]–[71] text-mining feature representation techniques are considered: unigrams, TF-IDF and polarity based. Literature suggests that these techniques have predominantly been used so far for sentiment analysis using supervised learning. However, to the best of our knowledge, for sentiment analysis, these techniques have not been explored earlier using (hierarchical) clustering techniques. Therefore, in the current study, we examined empirically how effective these feature representation techniques are for clustering, especially for the sentiment analysis task. Let T be the set of tweets in a collection, $T = (t_1, t_2 \ldots\ldots t_m)$. From this collection, unique dictionary of terms will be generated that represent the features. Suppose F be the set of features then $F = (f_1, f_2, f_3 \ldots\ldots f_n)$. To represent these features, an m x n matrix is generated as shown in Table 2, where m is the number of tweets and n is the number of features. In Table 2, there are four tweets and five features. Each representation scheme (unigram, TF-IDF and polarity) weighs these features in different ways as described below.

- Unigram representation weighs features using the boolean approach. That is, if a feature (term) from dictionary of terms exists in a tweet, it will be assigned a boolean value 1, otherwise 0. Consider Table 2, the columns represent features and rows represent tweets. For example, three features exist in $tweet_1$ i.e. f_1, f_2 and f_3; where f_1, f_2 and f_3 represent $term_1$, $term_2$ and $term_3$, respectively.
- TF-IDF is another term weighting scheme which depicts importance of a term to different documents in a given corpus [72]. It can be computed as: TF-IDF $= tf * log(|m|/df)$, where, tf is the frequency of a term in a given tweet t_i, $|m|$ is the number of tweets, and df is the number of documents (tweets) containing a given term. Consider Table 2 where the frequency of a feature f_1 is assumed to be 2 in $tweet_1$. TF-IDF for f_1 in $tweet_1$ will be calculated as TF-IDF$(f_1) = 2 * log(4/3)$, which gives the result TF-IDF$(f_1) = 0.2498$.
- Polarity based representation of features examines positive and negative strength of words [57]. For example, a word *good* is a positive word and *bad* is a negative

TABLE 3. Polarity-based representation of features.

	f_1	f_2	f_3	f_4	f_5
$tweet_1$	1	0	0.5	0	0
$tweet_2$	0	0	0	1	0
$tweet_3$	0	0	0.5	0	0
$tweet_4$	1	0	0	0	0

TABLE 4. Confusion matrix.

	Cluster_1	Cluster_2
Actual positive tweets	a	b
Actual negative tweets	c	d

word. Different dictionaries, like WordNet, have been used by researchers in order to represent features based on their polarity. In the current study, the dictionary created by [73] is used which contains 4783 negative words and 2006 positive words. In this representation, if a feature in a tweet matches to a positive word in the dictionary, it is assigned a polarity strength of 2, and if a feature matches to a negative word (in the dictionary), it will be assigned a polarity strength of 1. Values of 1 and 2 are normalized by dividing each by 2. Suppose that $tweet_1$ contains a positive feature f_1 and a negative feature f_3, $tweet_2$ contains a positive feature f_4, $tweet_3$ contains a negative feature f_3, and $tweet_4$ contains a positive feature f_1. The resulting representation is shown in Table 3.

G. EVALUATION METRICS

The performance of clustering and classification algorithms can be measured along different dimensions, including quality and time efficiency. The quality of clustering (and classification) results is a complex phenomenon that can be measured in different ways. One approach is *authoritativeness* whereby the clustering results of an algorithm are compared against the results of some authority (e.g., human experts). In this study, to show *authoritativeness*, we adopted the accuracy measure as reported in [74] and area under the curve (AUC) metric. In accuracy measure, a confusion matrix is built as shown in Table 4, where a, b, c and d are the number of tweets in each cluster. The cluster labeling is achieved as follows: if (a+d)>(b+c), cluster_1 will be considered as the positive cluster, otherwise cluster_2 will be regarded as the positive cluster. Then, the accuracy is calculated as (a+d)/n or (b+c)/n, where n is the total number of tweets in this case. In this study, we used this accuracy measure, AUC and time efficiency to measure the performance of both clustering and classification techniques.

IV. EMPIRICAL STUDY

Weka,[3] an open source machine learning library, is used for the experimental setup. The schematic diagram of this setup is shown in Figure 1 and its essential elements are outlined below.

- The process starts by necessary preprocessing wherein all tweets are first converted to lowercase letters and then tokenized. The stopwords, downloaded from the WordNet, are removed. All punctuations (e.g. !, =, ; etc.) and numbers are eliminated. Emoticons are not considered in this study, hence they are also removed. All the repeated words and URLs are removed as well in order to create unique tokens.
- Next, the preprocessed tweets are transformed into feature vectors. The features are represented using unigrams, TF-IDF and words' polarity.
- Finally, the feature representation is submitted to each competing algorithm, in turn, for clustering. At this stage, the cooperative framework is also envoked which selects the optimal cluster for the given tweet based on majority voting. Each algorithm terminates when the number of clusters is equal to 2 (i.e., the threshold point) because we need to create two groups i.e. one for positive tweets and the other for negative tweets.
- Accuracy and total time elapsed for each algorithm are recorded for later analysis.

All experiments were run on Core TM(i3) machine with 1.70 GHZ CPU speed and 4GB RAM.

A. WORKING OF THE PROPOSED CLUSTERING FRAMEWORK

It is instructive to use an example to demonstrate how the clustering framework works. We consider the tweets as shown in Table 2 for this purpose. In this example, the clustering process will terminate at a threshold of 2, i.e., when only two clusters remain in the hierarchy.

- SL initially considers all tweets as individual clusters, resulting into four clusters at the beginning. Using the threshold value 2, the algorithm will cluster the tweets as a Euclidean-distance measure: $d(tweet_1, tweet_1) = 0$
  $d(tweet_1, tweet_2)$
  $= \sqrt{(0-1)^2+(1-1)^2+(1-1)^2+(0-0)^2+(0-0)^2}$
  $= 1$
  Similarly, mutatis mutandis, the distance between the other tweet combinations is computed as shown in Table 5. In this matrix, it can be observed that the minimum distance is 1, which results in grouping $tweet_1$ and $tweet_2$ into a single cluster: $[tweet_1, tweet_2]$. The matrix in Table 5 will be updated in the next iteration. The distance between cluster $[tweet_1, tweet_2]$ and other tweets will be computed again (see second iteration). According to this updated matrix, the minimum distance is found between cluster $[tweet_1, tweet_2]$ and $tweet_4$ thereby placing $tweet_4$ in this cluster: $[tweet_1, tweet_2, tweet_4]$. Since, only two clusters remain in the hierarchy, the clustering process is terminated, yielding two clusters: $[tweet_1, tweet_2, tweet_4]$ and $[tweet_3]$ (see third iteration).

[3] https://www.cs.waikato.ac.nz/ml/weka/

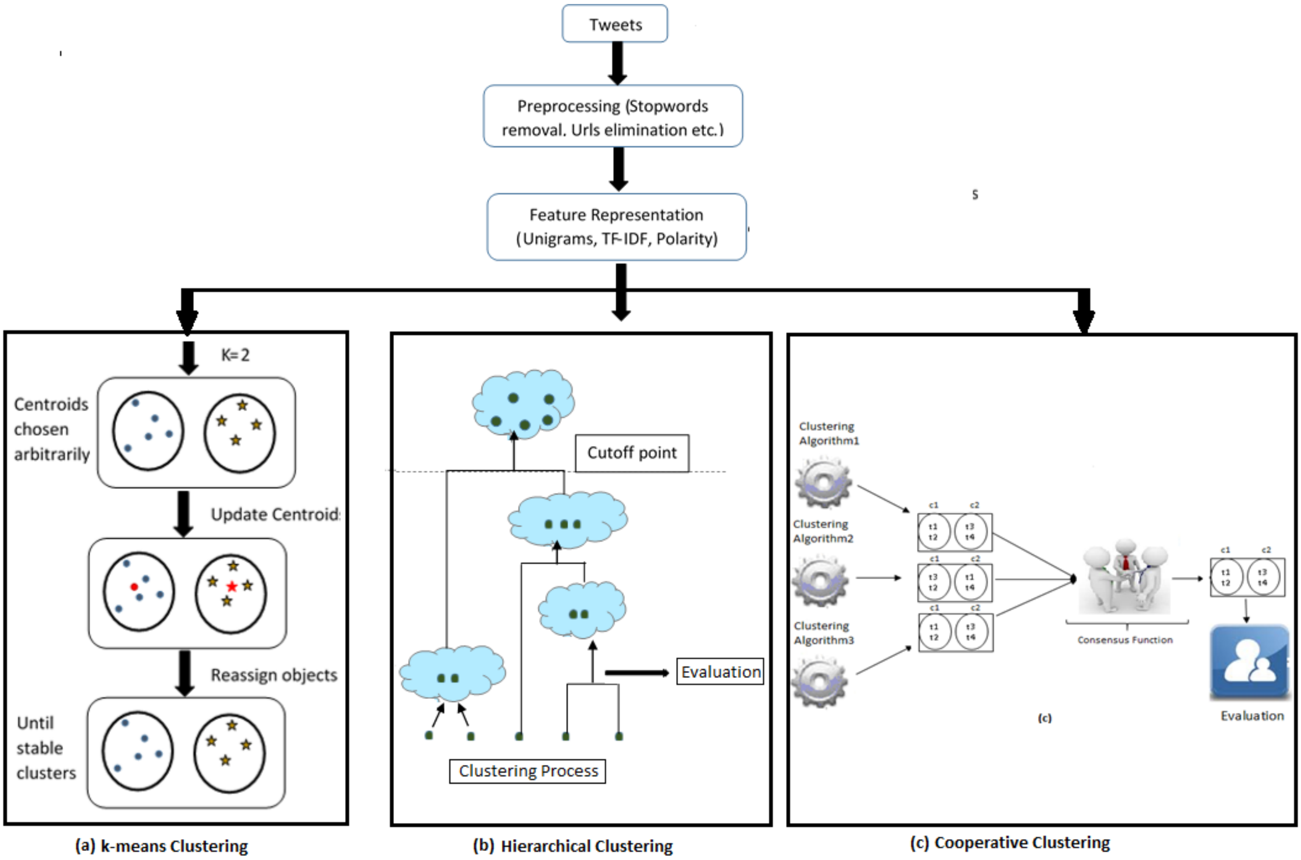

FIGURE 1. **Schematic diagram of clustering process.**

- CL starts with the same initial matrix in the first iteration (like SL). In the second iteration, $tweet_1$ and $tweet_3$ will be grouped into one cluster $[tweet_1, tweet_3]$ and the matrix values will be updated as shown in Table 5. Subsequently, in the third iteration, $tweet_2$ is grouped with cluster $[tweet_1, tweet_3]$. At this point, the clustering process terminates with two clusters: $[tweet_1, tweet_2, tweet_3]$ and $[tweet_4]$.
- AL clusters the tweets in Table 2 as follows:

$$d(tweet_1, tweet_2) = \sqrt{(-1)^2 + (0)^2 + (0)^2 + (0)^2 + (0)^2/1 * 1} = 1$$

  The distance matrix in the first iteration is the same like SL and CL. The algorithm then combines $tweet_1$ and $tweet_2$ into one cluster $[tweet_1, tweet_2]$ and the distance matrix is updated. Finally, in the third iteration, $tweet_4$ will be merged together with this cluster $[tweet_1, tweet_2, tweet_4]$ and the clustering process terminates, forming two clusters: $[tweet_1, tweet_2, tweet_4]$ and $[tweet_3]$.
- The cooperative framework assigns a tweet to its appropriate cluster based on majority vote. For example, both SL and AL place $tweet_4$ in cluster $[tweet_1, tweet_2, tweet_4]$, whereas CL places it in cluster $[tweet_4]$. Therefore, based on the majority vote, $tweet_4$ is assigned to cluster $[tweet_1, tweet_2, tweet_4]$.

V. RESULTS AND DISCUSSION

In this section, results of clustering and classification algorithms for twitter sentiment analysis are discussed. In order to perform tweet classification, 10-fold cross validation [24] is used. Performance of each algorithm is measured along two dimensions, clustering/classification quality and time efficiency. To measure clustering quality, a well-known accuracy metric [74], and area under the curve (AUC) are used. AUC is a measure of indication for the degree of separability between classes. AUC value generally lies between 0.5 and 1, closer the value to 1, better the performance. On the other hand, time efficiency is computed in terms of CPU time elapsed.

A. CLUSTERING AND CLASSIFICATION QUALITY

In Table 6, results of accuracy, AUC and time efficiency (secs) are presented for classification techniques. Table 7 and Table 8 depict accuracy, AUC and time efficiency, respectively, for clustering techniques.

1) HIERARCHICAL CLUSTERING

The results of three hierarchical-clustering algorithms in Table 7 indicate that overall CL (mean accuracy value on all datasets and features $\simeq$ 68 percent, AUC $\simeq$ 0.65) outperformed both SL (mean accuracy value $\simeq$ 62 percent, AUC $\simeq$ 0.59) and AL (mean accuracy value $\simeq$ 59 percent,

TABLE 5. Working of the clustering framework.

SL					
First iteration		$[tweet_1]$	$[tweet_2]$	$[tweet_3]$	$[tweet_4]$
	$[tweet_1]$	0	1	2.23	1.414
	$[tweet_2]$	1	0	2	1.732
	$[tweet_3]$	2.23	2	0	1.732
	$[tweet_4]$	1.414	1.732	1.732	0
Second iteration		$[tweet_1, tweet_2]$	$[tweet_3]$	$[tweet_4]$	
	$[tweet_1, tweet_2]$	0	2	1.414	
	$[tweet_3]$	2	0	1.732	
	$[tweet_4]$	1.414	1.732	0	
Third iteration		$[tweet_1, tweet_2, tweet_4]$	$[tweet_3]$		
	$[tweet_1, tweet_2, tweet_4]$	0	1.732		
	$[tweet_3]$	1.732	0		
CL					
Second iteration		$[tweet_1, tweet_3]$	$[tweet_2]$	$[tweet_4]$	
	$[tweet_1, tweet_3]$	0	2	1.732	
	$[tweet_2]$	2	0	1.732	
	$[tweet_4]$	1.732	1.732	0	
Third iteration		[tweet$_1$, $tweet_2, tweet_3$]	[tweet$_4$]		
	$[tweet_1$,tweet$_2$, $tweet_3]$	0	1.732		
	[tweet$_4$]	1.732	0		
AL					
Second iteration		$[tweet_1, tweet_2]$	$[tweet_3]$	$[tweet_4]$	
	$[tweet_1, tweet_2]$	0	2.1	1.573	
	$[tweet_3]$	2.1	0	1.732	
	$tweet_4]$	1.573	1.732	0	
Cooperative Clustering					
		$Cluster1$	$Cluster2$		
		$[tweet_1, tweet_2, tweet_4]$	$[tweet_3]$		

TABLE 6. Accuracy (Acc in %), AUC and time efficiency (TE in secs) values for classification techniques using 10-fold cross validation.

	Unigrams		TF-IDF		Polarity	
	NB	SVM	NB	SVM	NB	SVM
	Acc(AUC)(TE)	Acc(AUC)(TE)	Acc(AUC)(TE)	Acc(AUC)(TE)	Acc(AUC)(TE)	Acc(AUC)(TE)
HCR	71(0.71)(10)	69(0.66)(7)	72(0.70)(21)	72(0.73)(8)	62(0.63)(25)	65(0.65)(10)
SS-Tweet	69(0.75)(57)	70(0.71)(23)	70(0.75)(64)	74(0.73)(26)	50(0.62)(60)	55(0.65)(21)
STS-Test	70(0.77)(0.22)	75(0.73)(0.06)	71(0.70)(1)	75(0.70)(0.09)	64(0.53)(1)	67(0.69)(0.06)
NewTweets	70(0.67)(11)	72(0.71)(3)	77(0.77)(14)	79(0.79)(2)	53(0.5)(13)	60(0.60)(1)
Average	70(0.7)(26)	72(0.7)(8.2)	73(0.73)(25)	75(0.74)(9)	57(0.57)(24)	62(0.64)(11)

AUC $\simeq$ 0.52). The best performance for CL was observed for NewTweets dataset while using unigram features (accuracy $\simeq$ 75 percent, AUC $\simeq$ 0.75). The performance of both SL and AL was comparable to each other on all combinations of datasets and features.

2) HIERARCHICAL VS. PARTITIONED CLUSTERING

It is evident from Table 7 that both k-means and CL offer comparable performance. Even though the mean accuracy and AUC values for CL are slightly lower than that of k-means, the difference is negligible.

3) COOPERATIVE CLUSTERING VS. INDIVIDUAL CLUSTERING TECHNIQUES

The results in Table 7 show that individually on all datasets and features the cooperative clustering outperforms k-means and individual hierarchical-clustering algorithms. The mean accuracy and AUC values of cooperative clustering (accuracy $\simeq$ 75 percent, AUC $\simeq$ 0.68) is higher than that of CL (accuracy $\simeq$ 68 percent, AUC $\simeq$ 0.65) and k-means (accuracy $\simeq$ 70 percent, AUC $\simeq$ 0.66). These results suggest that improved cluster quality can be obtained by combining different techniques in a systematic manner.

TABLE 7. Time efficiency (secs) of the competing techniques.

	Unigram					TF-IDF					Polarity				
	SL	CL	AL	k-means	Cooperative clustering	SL	CL	AL	k-means	Cooperative clustering	SL	CL	AL	k-means	Cooperative clustering
HCR	65(0.6)	70(0.69)	62(0.5)	69(0.7)	69(0.7)	62(0.5)	70(0.68)	62(0.5)	71(0.7)	70(0.69)	62(0.5)	65(0.6)	60(0.5)	67(0.66)	65(0.66)
SS-Tweet	61(0.67)	63(0.63)	55(0.5)	70(0.7)	69(0.69)	63(0.6)	72(0.72)	61(0.67)	74(0.7)	70(0.73)	60(0.62)	62(0.7)	60(0.6)	68(0.69)	69(0.70)
STS-Test	64(0.65)	70(0.63)	56(0.5)	67(0.66)	70(0.67)	64(0.6)	70(0.62)	64(0.6)	70(0.7)	71(0.74)	58(0.5)	60(0.5)	55(0.5)	60(0.60)	64(0.63)
NewTweets	60(0.6)	75(0.75)	53(0.5)	76(0.72)	77(0.77)	66(0.66)	70(0.68)	60(0.5)	75(0.62)	70(0.61)	60(0.67)	64(0.65)	54(0.5)	67(0.6)	69(0.67)
Average	63(0.63)	70(0.67)	57(0.5)	71(0.69)	71(0.69)	64(0.59)	71(0.67)	62(0.56)	73(0.68)	70(0.69)	60(0.57)	63(0.61)	57(0.5)	66(0.63)	67(0.66)

TABLE 8. Time efficiency (secs) of the competing techniques.

	Processing time(secs)														
	Unigram					TF-IDF					Polarity				
	SL	CL	AL	k-means	Cooperative clustering	SL	CL	AL	k-means	Cooperative clustering	SL	CL	AL	k-means	Cooperative clustering
HCR	385.3	280	356.5	3	1021.8	491.4	307.9	425	2.34	1224.4	385.8	303.2	344.8	5.64	1033.9
SS-Tweet	755.8	552	645.5	6.25	1953.6	994.8	566	824.4	6.77	2385.5	1037.6	555	769	14	2362
STS-Test	2.0	1.9	1.9	0.46	5.86	3.5	2.1	2.9	0.45	8.7	1.9	2.1	2.0	0.47	6.0
NewTweets	180.3	95.9	145.1	1.1	421.42	108.3	87.1	104.4	1.3	299.8	184.1	94.4	147.2	1.19	426.06
Average	330.9	232.5	287.2	2.71	850.7	399.5	240.7	339.175	2.71	979.6	402.6	238.8	315.6	5.34	957.1

4) PERFORMANCE OF CLASSIFIERS

In classification techniques, SVM performs better than Naïve Bayes as shown in Table 6. Individually, highest accuracy can be seen for SVM (accuracy $\simeq$ 79 percent, AUC $\simeq$ 0.79) on NewTweets. However, on average, performance of Naïve Bayes and SVM is close for all datasets.

5) CLASSIFIERS VS. CLUSTERING

It is evident from Table 6 and Table 7 that, on an average performance of k-means, CL and cooperative clustering is close to Naïve Bayes and SVM in case of unigrams and TF-IDF based features. Performance of CL, k-means and cooperative clustering is better than Naïve Bayes and SVM for polarity based features.

6) FEATURES ANALYSIS

We also examined the impact of different feature representation schemes on cluster quality. The findings elucidate that all the representations performed equally good in determining the cluster quality. Among clustering techniques, unigrams scheme showed the best performance on NewTweets dataset by using cooperative clustering (accuracy $\simeq$ 77 percent, AUC $\simeq$ 0.77). In addition, mean accuracy values show that k-means and cooperative clustering perform better than SL, CL and AL for all types of features. While among classifiers, SVM gives best performance for TF-IDF on NewTweets. For comparative analysis between clustering and classification based on features, consider Table 6 and Table 7. It is interesting to note that both clustering and classification achieve the best performance on TF-IDF.

B. TIME EFFICIENCY

We measured time efficiency of each algorithm in terms of CPU time elapsed (secs) as shown in Table 6 and Table 8 for classification and clustering, respectively.

1) HIERARCHICAL CLUSTERING

Among the three hierarchical clustering algorithms, it is interesting to note that the processing time taken by CL (mean time across all features $\simeq$ 235 secs) is considerably lower as compared to SL (mean time $\simeq$ 355 secs) and AL (mean time $\simeq$ 330 secs). The SL seems fairly expensive compared to CL and AL.

2) HIERARCHICAL VS. PARTITIONED CLUSTERING

The k-means algorithm outperforms all other techniques in terms of time efficiency. On average, the total elapsed time for each dataset and feature representation is below 5 secs, which is far better than CL, which offers the best time efficiency ($\simeq$ 235 secs) among hierarchical clustering algorithms. As expected, cooperative clustering is very expensive (mean time across all features $\simeq$ 930 secs), because it combines three individual hierarchical clustering techniques. Hence the total time elapsed by each technique adds up in cooperative clustering.

3) FEATURES ANALYSIS

Both unigrams and TF-IDF offered competitive time efficiency ($\simeq$ 2 secs). However, words' polarity feature is relatively slower ($\simeq$ 5 secs) as compared to other two representations.

4) CLASSIFICATION ALGORITHMS

Table 6 shows that on an average Naïve takes more time as compared to SVM for all datasets. Both classifiers took the maximum time for SS-Tweet dataset. It is due to the fact that the number of features are greater in number for SS-Tweet in comparison with other datasets as shown in Table 1. The minimum time is consumed for STS-Test dataset by both techniques.

5) CLASSIFICATION VS. CLUSTERING

Comparative analysis from Table 6 and Table 8 reveals that among all competing clustering and classification techniques, k-means took the least time for all datasets and all types of features. SVM and Naïve took modest time while SL, CL, AL and cooperative clustering are expensive in terms of time consumption.

C. DISCUSSION

Some interesting observations relevant to the strengths and limitations of clustering approaches arose during this study which are worth discussing. Our results suggest that hierarchical clustering techniques offer better-quality clusters as compared to the k-means algorithm. However, the latter offers better time efficiency. If output quality and time efficiency are addressed together, this raises the question of how these two factors should be traded off against each other. If one algorithm produces better quality clusters than that of the other but is beaten in time efficiency, which algorithm should be preferred? Perhaps output quality be ranked higher as compared to time efficiency because ultimately it is the solution quality that matters and hierarchical clustering techniques give better quality results. Cooperative clustering generates most authoritative clusters but it is computationally very expensive as compared to the other techniques. However, it might be interesting to see how the performance of this approach could change both in terms of solution quality and time efficiency if multiple clustering techniques can be integrated using other methods, for example, in a cascading manner. Our notion of cooperative clustering also makes an interesting case for applying this on a high-performance computing (HPC) platform for better results (both in terms of solution quality and speed).

Computational cost of hierarchical clustering algorithms increases with the increase in the size of the dataset. However, low computational cost of k-means still supports the argument that clustering can be useful for sentiment analysis as compared to supervised learning techniques, which require manual labeling of data.

A detailed comparative analysis suggests that in terms of accuracy, performance of CL, k-means and cooperative clustering is comparable to classification. Furthermore, in terms of time efficiency, k-means is least expensive algorithm. An important motivation to use unsupervised learning instead of supervised learning is the labeled-data bottleneck in the latter. Because twitter datasets can be huge in size, if supervised learning techniques are used there is a requirement to manually label the data which is a time consuming and tedious task. The results of our empirical study suggest that the accuracy of the proposed clustering framework is comparable to classifiers. We also compared the performance of hierarchical clustering techniques with k-means, whose performance has already been shown similar to supervised learning techniques [74], and showed that the performance of the former, especially complete linkage, is comparable to k-means. The findings of our study depict that clustering techniques can be used for reliable (twitter) sentiment analysis. Therefore, it is reasonable to explore clustering techniques for twitter sentiment analysis if clustering accuracy is comparable to that of supervised learning.

Recently, a paradigm shift from word-level to concept-level sentiment analysis encourages to consider this latest methodology for sentiment analysis in which deep learning is taken into account [45]–[47], [54]. In the concept based approach, word embedding vectors [75] are generally used instead of conventional bag-of-words models [76]–[78]. In [54], authors used unlabeled texts and exploited lexical co-occurrence information to generate a direct acyclic graph of concepts. The polarity score of initially known concepts is used in an incremental manner to compute polarity scores of new concepts. Building on earlier work [39], findings encourage us to extend the existing work by taking deep-learning approaches into account.

VI. CONCLUSION AND FUTURE WORK

Twitter sentiment analysis is an important yet challenging problem. In this work, an empirical study aimed at investigating to what extent the individual hierarchical clustering techniques (SL, AL and CL) and their combination (cooperative clustering) improve the quality of clustering for sentiment analysis of unlabeled data. The results of these techniques were also compared with k-means and two state-of-the-art classifiers (SVM and Naïve Bayes). The accuracy metric and AUC measure were used to measure the quality of clustering/classification and the CPU elapsed time was computed for time efficiency. Experimental results revealed that, on average, CL provided better quality clusters as compared to SL, AL and k-means. Cooperative clustering seems to be the most suitable in terms of creating high quality clusters than all other techniques. However, time efficiency of k-means clustering is the best as compared to other techniques. The results also suggest that, especially, accuracy of the proposed cooperative-clustering framework is comparable to classifiers which is encouraging. In summary, our results suggest that cooperative clustering based on majority voting provides better cluster quality with tradeoff of poor time efficiency. The findings of our study depict that clustering techniques can be used for reliable (twitter) sentiment analysis. One future direction can be to combine hierarchical clustering and k-means clustering to balance the tradeoff of the clustering quality and time efficiency. We also intend to apply our notion of cooperative clustering on a HPC platform to account for time efficiency. Another useful future work is to explore the possibility of using more recent Sentic-computing methods, including deep recurrent neural network [78] to aid the process of sentiment analysis.

REFERENCES

[1] E. Martinez-Camara, M. T. Martin-Valdivia, L. A. Urena-Lopez, and A. R. Montejo-Raez, "Sentiment analysis in Twitter," *Natural Lang. Eng.*, vol. 20, no. 1, pp. 1–28, 2014.

[2] E. D'Avanzo and G. Pilato, "Mining social network users opinions' to aid buyers' shopping decisions," *Comput. Hum. Behav.*, vol. 51, pp. 1284–1294, Oct. 2015.

[3] S. Das and H. K. Kalita, "Sentiment analysis for Web-based big data: A survey.," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, pp. 1996–1999, 2017.

[4] P. Adinolfi, E. D'Avanzo, M. D. Lytras, I. Novo-Corti, and J. Picatoste, "Sentiment analysis to evaluate teaching performance," *Int. J. Knowl. Soc. Res.*, vol. 7, no. 4, pp. 86–107, Oct. 2016.

[5] E. D'Avanzo, G. Pilato, and M. Lytras, "Using Twitter sentiment and emotions analysis of Google trends for decisions making," *Program*, vol. 51, no. 3, pp. 322–350, Sep. 2017.

[6] H. Peng, E. Cambria, and A. Hussain, "A review of sentiment analysis research in Chinese language," *Cognit. Comput.*, vol. 9, no. 4, pp. 423–435, Aug. 2017.

[7] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Project Rep. Stanford*, vol. 1, no. 12, p. 2009, 2009.

[8] M. S. Neethu and R. Rajasree, "Sentiment analysis in Twitter using machine learning techniques," in *Proc. 4th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2013, pp. 1–5.

[9] M. Hagen, M. Potthast, M. Büchner, and B. Stein, "Twitter sentiment detection via ensemble classification using averaged confidence scores," in *Proc. Eur. Conf. Inf. Retr.*, 2015, pp. 741–754.

[10] C. Troussas, A. Krouska, and M. Virvou, "Evaluation of ensemble-based sentiment classifiers for Twitter data," in *Proc. 7th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2016, pp. 1–6.

[11] K. Dashtipour, S. Poria, A. Hussain, E. Cambria, A. Y. A. Hawalah, A. Gelbukh, and Q. Zhou, "Multilingual sentiment analysis: State of the art and independent comparison of techniques," *Cognit. Comput.*, vol. 8, no. 4, pp. 757–771, Aug. 2016.

[12] M. Z. Asghar, A. Khan, A. Bibi, F. M. Kundi, and H. Ahmad, "Sentence-level emotion detection framework using rule-based classification," *Cognit. Comput.*, vol. 9, no. 6, pp. 868–894, Dec. 2017.

[13] Y. Xia, E. Cambria, A. Hussain, and H. Zhao, "Word polarity disambiguation using Bayesian model and opinion-level features," *Cognit. Comput.*, vol. 7, no. 3, pp. 369–380, Jun. 2015.

[14] R. Pandarachalil, S. Sendhilkumar, and G. S. Mahalakshmi, "Twitter sentiment analysis for large-scale data: An unsupervised approach," *Cognit. Comput.*, vol. 7, no. 2, pp. 254–262, Apr. 2015.

[15] I. Himelboim, X. Xiao, D. K. L. Lee, M. Y. Wang, and P. Borah, "A social networks approach to understanding vaccine conversations on Twitter: Network clusters, sentiment, and certainty in HPV social networks," *Health Commun.*, vol. 35, no. 5, pp. 607–615, Apr. 2020.

[16] H. Achrekar, A. Gandhe, R. Lazarus, S.-H. Yu, and B. Liu, "Predicting flu trends using Twitter data," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2011, pp. 702–707.

[17] X. Ji, S. A. Chun, Z. Wei, and J. Geller, "Twitter sentiment classification for measuring public health concerns," *Social Netw. Anal. Mining*, vol. 5, no. 1, p. 13, Dec. 2015.

[18] M. J. Paul, A. Sarker, J. S. Brownstein, A. Nikfarjam, M. Scotch, K. L. Smith, and G. Gonzalez, "Social media mining for public health monitoring and surveillance," in *Proc. Biocomput.*, Jan. 2016, pp. 468–479.

[19] D. Mowery, H. A. Smith, T. Cheney, C. Bryan, and M. Conway, "Identifying depression-related tweets from Twitter for public health monitoring," *Online J. Public Health Informat.*, vol. 8, no. 1, 2016.

[20] A. Jungherr, "Twitter use in election campaigns: A systematic literature review," *J. Inf. Technol. Politics*, vol. 13, no. 1, pp. 72–91, Jan. 2016.

[21] A. Bovet, F. Morone, and H. A. Makse, "Validation of Twitter opinion trends with national polling aggregates: Hillary clinton vs donald trump," 2016, *arXiv:1610.01587*. [Online]. Available: http://arxiv.org/abs/1610.01587

[22] T. Menkhoff, Y. W. Chay, M. L. Bengtsson, C. J. Woodard, and B. Gan, "Incorporating microblogging ('tweeting') in higher education: Lessons learnt in a knowledge management course," *Comput. Hum. Behav.*, vol. 51, pp. 1295–1302, Oct. 2015.

[23] Y. Yu and X. Wang, "World cup 2014 in the Twitter world: A big data analysis of sentiments in U.S. Sports fans' tweets," *Comput. Hum. Behav.*, vol. 48, pp. 392–400, Jul. 2015.

[24] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*. San Mateo, CA, USA: Morgan Kaufmann, 2006.

[25] S. Maes, K. Tuyls, B. Vanschoenwinkel, and B. Manderick, "Credit card fraud detection using Bayesian and neural networks," *Proc. 1st Int. NAISO Congr. Neuro Fuzzy Technol.*, 2002, pp. 261–270.

[26] A. C. Tan and D. Gilbert, "Ensemble machine learning on gene expression data for cancer classification," in *Proc. New Zealand Bioinf. Conf.*, 2003, pp. 1–15.

[27] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining.," in *Proc. Eur. Lang. Resour. Assoc. (ELRA)*, Valletta, Malta, 2010, vol. 10, no. 2010, pp. 1320–1326.

[28] N. F. F. da Silva, E. R. Hruschka, and E. R. Hruschka, "Tweet sentiment analysis with classifier ensembles," *Decis. Support Syst.*, vol. 66, pp. 170–179, Oct. 2014.

[29] A. Ortigosa, J. M. Martín, and R. M. Carro, "Sentiment analysis in Facebook and its application to e-learning," *Comput. Hum. Behav.*, vol. 31, pp. 527–541, Feb. 2014.

[30] S. Kübler, C. Liu, and Z. A. Sayyed, "To use or not to use: Feature selection for sentiment analysis of highly imbalanced data," *Natural Lang. Eng.*, vol. 24, no. 1, pp. 3–37, Jan. 2018.

[31] X. Hu, J. Tang, H. Gao, and H. Liu, "Unsupervised sentiment analysis with emotional signals," in *Proc. 22nd Int. Conf. World Wide Web (WWW)*, 2013, pp. 607–618.

[32] F. Li, Y. Qian, J. Wang, C. Dang, and B. Liu, "Cluster's quality evaluation and selective clustering ensemble," *ACM Trans. Knowl. Discovery Data*, vol. 12, no. 5, pp. 1–27, 2018.

[33] S. Khedairia and M. T. Khadir, "A multiple clustering combination approach based on iterative voting process," *J. King Saud Univ.-Comput. Inf. Sci.*, Sep. 2019.

[34] R. Naseem, O. Maqbool, and S. Muhammad, "Cooperative clustering for software modularization," *J. Syst. Softw.*, vol. 86, no. 8, pp. 2045–2062, Aug. 2013.

[35] Z. Shah, R. Naseem, M. A. Orgun, A. Mahmood, and S. Shahzad, "Software clustering using automated feature subset selection," in *Proc. Int. Conf. Adv. Data Mining Appl.*, 2013, pp. 47–58.

[36] R. Kashef and M. S. Kamel, "Cooperative clustering," *Pattern Recognit.*, vol. 43, no. 6, pp. 2315–2329, Jun. 2010.

[37] Y. Li, Y. Lv, S. Wang, J. Liang, J. Li, and X. Li, "Cooperative hybrid semi-supervised learning for text sentiment classification," *Symmetry*, vol. 11, no. 2, p. 133, 2019.

[38] R. Soni and K. James Mathai, "Improved Twitter sentiment prediction through cluster-then-predict model," 2015, *arXiv:1509.02437*. [Online]. Available: http://arxiv.org/abs/1509.02437

[39] L. Oneto, F. Bisio, E. Cambria, and D. Anguita, "Semi-supervised learning for affective common-sense reasoning," *Cognit. Comput.*, vol. 9, no. 1, pp. 18–42, Feb. 2017.

[40] Vijaya, S. Sharma, and N. Batra, "Comparative study of single linkage, complete linkage, and ward method of agglomerative clustering," in *Proc. Int. Conf. Mach. Learn., Big Data, Cloud Parallel Comput. (COMITCon)*, Feb. 2019, pp. 568–573.

[41] N. Yadav, A. Kobren, N. Monath, and A. McCallum, "Supervised hierarchical clustering with exponential linkage," 2019, *arXiv:1906.07859*. [Online]. Available: http://arxiv.org/abs/1906.07859

[42] L. Gubu, D. Rosadi, and Abdurakhman, "Classical portfolio selection with cluster analysis: Comparison between hierarchical complete linkage and ward algorithm," in *Proc. 8TH SEAMS-UGM Int. Conf. Math. ITS Appl., Deepening Math. Concepts Wider Appl. Through Multidisciplinary Res. Industries Collaborations*, vol. 2192. Melville, NY, USA: AIP Publishing LLC, 2019, Art. no. 090004.

[43] A. Aue and M. Gamon, "Customizing sentiment classifiers to new domains: A case study," *Proc. Recent Adv. Natural Lang. Process. (RANLP)*, vol. 1, no. 3, pp. 207–218, 2005.

[44] M. Abdul-Mageed, A. Buffone, H. Peng, J. Eichstaedt, S. Giorgi, and L. Ungar, "Recognizing pathogenic empathy in social media.," in *Proc. Int. Conf. Web Social Media*, 2017, pp. 448–451.

[45] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain, "Sentic LSTM: A hybrid network for targeted aspect-based sentiment analysis," *Cognit. Comput.*, vol. 10, no. 4, pp. 639–650, Aug. 2018.

[46] F. Z. Xing, E. Cambria, and R. E. Welsch, "Intelligent asset allocation via market sentiment views," *IEEE Comput. Intell. Mag.*, vol. 13, no. 4, pp. 25–34, Nov. 2018.

[47] S. Poria, E. Cambria, and A. Gelbukh, "Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 2539–2544.

[48] M. Bibi, M. S. A. Nadeem, I. H. Khan, S.-O. Shim, I. R. Khan, U. Naqvi, and W. Aziz, "Class association and attribute relevancy based imputation algorithm to reduce Twitter data for optimal sentiment analysis," *IEEE Access*, vol. 7, pp. 136535–136544, 2019.

[49] K. Müller and C. Schwarz. (2019). *From Hashtag to Hate Crime: Twitter and Anti-Minority Sentiment*. [Online]. Available: https://ssrn.com/abstract.

[50] L. Wang, J. Niu, and S. Yu, "SentiDiff: Combining textual information and sentiment diffusion patterns for Twitter sentiment analysis," *IEEE Trans. Knowl. Data Eng.*, early access, Apr. 26, 2019, doi: 10.1109/TKDE.2019.2913641.

[51] W.-H. Liao, Y.-T. Huang, T.-H. Yang, and Y.-C. Wu, "Analyzing social network data using deep neural networks: A case study using Twitter posts," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2019, pp. 237–2371.

[52] A. Ibrahim, D. Rayside, and R. Kashef, "Cooperative based software clustering on dependency graphs," in *Proc. IEEE 27th Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2014, pp. 1–6.

[53] R. Kashef, "Cooperative clustering model and its applications," Ph.D. dissertation, Dept. Elect. Comput. Eng., Univ. Waterloo, Waterloo, ON, Canada, 2008.

[54] N. Ofek, S. Poria, L. Rokach, E. Cambria, A. Hussain, and A. Shabtai, "Unsupervised commonsense knowledge enrichment for domain-specific sentiment analysis," *Cognit. Comput.*, vol. 8, no. 3, pp. 467–477, Jun. 2016.

[55] R. Dehkharghani, B. Yanikoglu, Y. Saygin, and K. Oflazer, "Sentiment analysis in turkish at different granularity levels," *Natural Lang. Eng.*, vol. 23, no. 4, pp. 535–559, Jul. 2017.

[56] M. Speriosu, N. Sudan, S. Upadhyay, and J. Baldridge, "Twitter polarity classification with label propagation over lexical links and the follower graph," in *Proc. 1st Workshop Unsupervised Learn. NLP*, 2011, pp. 53–63.

[57] L. F. S. Coletta, N. F. F. D. Silva, E. R. Hruschka, and E. R. Hruschka, "Combining classification and clustering for tweet sentiment analysis," in *Proc. Brazilian Conf. Intell. Syst.*, Oct. 2014, pp. 210–215.

[58] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social Web," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 63, no. 1, pp. 163–173, Jan. 2012.

[59] H. Saif, M. Fernandez, Y. He, and H. Alani, "Evaluation datasets for Twitter sentiment analysis: A survey and a new dataset, the STS-gold," in *Proc. 1st Int. Workshop Emotion Sentiment Social Expressive Media, Approaches Perspect. AI (ESSEM)*, 2013, pp. 9–21.

[60] S. Vega-Pons and J. Ruiz-Shulcloper, "A survey of clustering ensemble algorithms," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 25, no. 3, pp. 337–372, May 2011.

[61] M. Rezwanul, A. Ali, and A. Rahman, "Sentiment analysis on Twitter data using KNN and SVM," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 6, pp. 19–25, 2017.

[62] M. Ahmad, S. Aftab, M. Salman, N. Hameed, I. Ali, and Z. Nawaz, "SVM optimization for sentiment analysis," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 4, pp. 393–398, 2018.

[63] K. Lu and J. Wu, "Sentiment analysis of film review texts based on sentiment dictionary and SVM," in *Proc. 3rd Int. Conf. Innov. Artif. Intell. (ICIAI)*, 2019, pp. 73–77.

[64] K. Korovkinas, P. Danėnas, and G. Garšva, "SVM and k-means hybrid method for textual data sentiment analysis," *Baltic J. Modern Comput.*, vol. 7, no. 1, pp. 47–60, 2019.

[65] R. Singh and V. Goel, "Various machine learning algorithms for Twitter sentiment analysis," in *Information and Communication Technology for Competitive Strategies*. Singapore: Springer, 2019, pp. 763–772.

[66] L. Barbosa and J. Feng, "Robust sentiment detection on Twitter from biased and noisy data," in *Proc. 23rd Int. Conf. Comput. Linguistics, Posters*, 2010, pp. 36–44.

[67] E. Kouloumpis, T. Wilson, and J. D. Moore, "Twitter sentiment analysis: The good the bad and the OMG!" in *Proc. Int. Conf. Web Social Media*, 2011, pp. 538–541.

[68] L. Hong, O. Dan, and B. D. Davison, "Predicting popular messages in Twitter," in *Proc. 20th Int. Conf. Companion World Wide Web (WWW)*, 2011, pp. 57–58.

[69] O. Phelan, K. McCarthy, and B. Smyth, "Using Twitter to recommend real-time topical news," in *Proc. 3rd ACM Conf. Recommender Syst. (RecSys)*, 2009, pp. 385–388.

[70] S. Phuvipadawat and T. Murata, "Breaking news detection and tracking in Twitter," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol.*, Aug. 2010, pp. 120–123.

[71] K. Lee, D. Palsetia, R. Narayanan, M. M. A. Patwary, A. Agrawal, and A. Choudhary, "Twitter trending topic classification," in *Proc. IEEE 11th Int. Conf. Data Mining Workshops*, Dec. 2011, pp. 251–258.

[72] G. Paltoglou and M. Thelwall, "A study of information retrieval weighting schemes for sentiment analysis," *Proc. 48th Annu. Meeting Assoc. Comput. Linguistics*, 2010, pp. 1386–1395.

[73] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2004, pp. 168–177.

[74] G. Li and F. Liu, "A clustering-based approach on sentiment analysis," in *Proc. IEEE Int. Conf. Intell. Syst. Knowl. Eng.*, Nov. 2010, pp. 331–337.

[75] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

[76] S. Poria, E. Cambria, R. Bajpai, and A. Hussain, "A review of affective computing: From unimodal analysis to multimodal fusion," *Inf. Fusion*, vol. 37, pp. 98–125, Sep. 2017.

[77] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis : A survey," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 10, no. 12, pp. 701–719, 2018.

[78] I. Chaturvedi, E. Cambria, R. E. Welsch, and F. Herrera, "Distinguishing between facts and opinions for sentiment analysis: Survey and challenges," *Inf. Fusion*, vol. 44, pp. 65–77, Nov. 2018.

MARYUM BIBI received the M.Phil. degree in computer science from Quaid-i-Azam University Islamabad, Pakistan, in 2012. She is currently pursuing the Ph.D. degree with the Department of Computer Sciences and Information Technology, The University of Azad Jammu & Kashmir, Muzaffarabad. She has publications in area of machine learning and reverse engineering including impact factor publication. She has various achievements including a Gold Medal on securing First Position in HSSC.

WAJID AZIZ received the B.Sc. and M.Sc. degrees from The University of Azad Jammu & Kashmir University (UAJ&K), Muzaffarabad, Pakistan, and the Ph.D. degree from the Pakistan Institute of Engineering & Applied Sciences (PIEAS), Islamabad, Pakistan. He was worked with UAJ&K, in 1998, as a Lecturer. He is currently working as a Professor with the College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia. He has published three books and more than 50 research articles in the reputed national and international journals and conference proceedings. His core research expertise is in biomedical information systems and his focused research areas are biomedical signal processing, time series analysis, and biomedical data analytics. Based on his academic and research contributions, he was the recipient of the HEC University Best Teacher Award for the year 2012–2013 awarded by HEC Pakistan, in 2014, and the University Best Teacher Award by the University of AJ&K, in 2013.

MAJID ALMARAASHI is currently an Associate Professor in artificial intelligence with the University of Jeddah, Saudi Arabia. In his previous works, he applied AI algorithms to different problem domains. In addition, he has worked on national funded projects in optimization, renewable energy, diseases modeling, and tomography applications. Regarding consultancy, he served as a technology consultant in some government agencies as well as the private sector during the last six years. Alongside publishing in some prominent journals in AI, he was granted two novel patents in using artificial intelligence with satellite-based monitoring and tomography applications. His main interest in computational intelligence is the theory and applications of optimization and soft computing.

IMTIAZ HUSSAIN KHAN received the M.S. degree in computer science from the University of Essex, U.K., in 2005, and the Ph.D. degree in artificial intelligence from the University of Aberdeen, U.K., in 2010. He is currently an Associate Professor with the Department of Computer Science, King Abdulaziz University, Jeddah, Saudi Arabia. He is an author of more than 20 research articles. His research interests are natural language processing, cognitive computing, and evolutionary computation.

MALIK SAJJAD AHMED NADEEM received the Ph.D. degree from the University of Paris, in 2011. He is currently an Assistant Professor with the Department of Computer Sciences and Information Technology, The University of Azad Jammu & Kashmir, Muzaffarabad. He has published various journal articles in the area of machine learning.

NAZNEEN HABIB was worked with The University of Azad Jammu & Kashmir (UAJ&K), in 2002, as a Lecturer, where she is currently an Assistant Professor in sociology. Her core research expertise is in sociology of health and illness and focused area of research is socio-cultural factors contributing towards anemia in women. Other research interests include social and network informatics, women health, and public health, and the main research interest is in computational intelligence.