A Compound Classification Algorithm for Opinion Mining on Twitter

Sandhya Rani Bairi
Computer Engineering
National Institute of Technology Karnataka
Email: www.14co205@nitk.edu.in

***Abstract*—Millions of users now use social networking sites to share their views and opinions on various trends and topics. Of those, Twitter being a valuable application conveys lot of ideas of individuals on several current issues. Analyzing such text and retrieving the sentiment through text mining can help the concerned in making better decisions in different domains. Several companies and organizations look at this as an intelligent way of determining public's opinion on current trends. The current techniques used for opinion mining are not upto the mark in classifying the sentiment. Most of the exisiting techniques classify sentiment into three types namely positive, negative and neutral. This paper suggests a new method of taking tweeets as the dataset and classifying the sentiments of the the tweets into more types giving a greater intuition on the public emotions. This paper also presents a comparison of the proposed method with an existing Sentiment Analysis tool.**

***Index Terms*—Opinion mining, Twitter, Sentiment Analysis**

I. Introduction

This is an era in which technology is advancing tremendously particularly in computer science. Social networking sites like Facebook, Twitter have become more popular than ever. People spend lot of their time on social media everyday. These platforms enable individuals to share their views, ideas and opinions on several products, events or issues. Such informal data can be mined in order to draw useful conclusions in several domains. Analyzing and modeling of activities occurring on several platforms is known to be Social Computing. The amount of digital data has been increasing at an alarming rate for the past few years. But only 3% of that data is actually used for analysis as of 2012. Such large amounts of data can be analyzed and processed to get valuable information for various companies, organizations and the governments through data mining. Part of the data mining called text mining is used to target the textual data which is mostly unstructured making the job challenging.

Twitter is highly beneficial when it comes to text mining owing to its 140 word-limit so that the people's opinions would be more explicit. The textual information on web is generally categorized into two types, namely, fact data and sentiment data. The fact data is the parlance which is impartial and unbiased. On the other hand, the sentiment data is emotional information that contribuetd to the individual's emotion on a specific entity. Sentiment Analysis is a process of identifying the sentiment of data shared by an individual resulting in user's perspective over that issue or product. This is a very effective and an efficient way to unveil the public opinions. For example, a business oriented corporation can utilize this technique to learn the users' feedback over their products and services. A political leader can find out his/her goodwill in the society.

Topic Modelling is a procedure used to determine the topics present in a text and to obtain unrevealed patterns displayed by a text corpus thereby helps in better decision making. This process can be distinguished from rule-based text mining methods in that these techniques make use of regular expressions or dictionary based classification techniques which comes under unsupervised methods. There are many common methods for deriving topics from a given text like Term Frequency and Inverse Document Frequency (tf-idf), NonNegative Matrix Factorization techniques etc. Latent Dirichlet Allocation is one such a techinque which is the most famous topic modeling technique. LDA considers that all textual information documents are generated from an aggregation of different topics. Those topics then produce words on the basis of their probability distribution. When a set of such text documents are provided, LDA retraces its steps and finds out what topics created those documents in the first place.

Research studies indicate that the real-life events certainly have a eloquent and informative effect on the public sentiment on Twitter. Therefore, it has now become important to find out the actual causes for such sentiment be it positive or negative or neutral. Currently research is going on in this area that is public sentiment variation for which opinion mining is the foundation which has to be accurate in order to find the reasons for such variations of the public sentiment. For instance, if a company receives a negative feedback on their product, they would want to find out the possible reasons for such expression from customers. Similarly, if a politician faces criticism in any way from the society, he/she may wish to know the causes for it and work on it in future.

In this paper, a method to recognize the emotion of a particular tweet is proposed such as happy, cheerful, disappointed etc. In contrast with the usual sentiment analysis techniques that classify the tweets as positive, negative or neutral, this method gives a deeper insight on public sentiment.

The rest of the paper is organized in the following way. In section II, some operating work in the domain of opinion mining is discussed. Section III relates the methodology to perform opinion mining. Section IV describes several supervised machine learning techniques used to perform opinion mining. Section V explains the proposed application scenario with pseudo code and project organization. Section VI elucidates the results and comparison with an existing tool. Finally Section VII gives the conclusion and future scope.

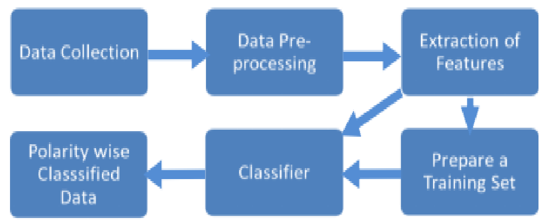
II. Related Work

In recent years, a lot of research has been going on in this area. [1] includes a method to categorize the data of the students on Twitter into different classes to confront several of their issues. In [2], another rational method was proposed to mine the opinions postedby the public on different social networking sites. The authors studied the emotions of the textual data through lexicon and semantic based approaches. [3] explains several techniques of classification and compares the robustness and correctness of classifiers like Support Vector Machine, Naive Bayes, Maximum Entropy etc. In [4], a hybrid method was introduced which uses lexicon approach and a machine learning approach to identify the polarity of textual data in consumer products field. In [5], a series of machine learning techniques were explained with semantic analysis to categorize Twitter data particularly the product reviews with WordNet to gain precision. [6] suggests an unconventional procedure to interpret the ups and downs of stocks with several fiscal communication panels and conducted prediction automatically for the stock market. In [7], several techniques of feature extraction and preferences such as information gain, mutual information, CHI statistics were analyzed and compared to scrutinize the capacity of opinion mining in e-learning sector. In [8], another classifier model was proposed to categorize the TV shows of Brazil based on the public reviews which resulted in having great robustness and accuracy. [9] depicts a system that retrieves tweets and categorizes them with the help of domain based improvement techniques to decrease the loss of information. [10] explains an upgraded version of opinion analysis method wherein the anomalies or abnormalities present in the datasets can be eliminated.

[11] is the first and foremost research done on understanding the causes for opinions shared by the public in microblogging services like Twitter. Emerging topics also known as foreground topics during the periods considered for sentiment analysis, are related to the actual reasons behind the variations to a great extent. In regard to this observational fact, an LDA based model was proposed, namely, Foreground and Background LDA (FB-LDA), to keep foreground topics and discard already fixed background topics. These emerging topics are proved to be accurate in finding genuine causes for sentiment variations. FB-LDA is followed by another generative model called Reason Candidate and Background LDA (RCB-LDA) for which the most indicative tweets for emerging topics are chosen as 'reason candidates' and are given ranks based on their support during the variation period considered. [12],[13] deals with opinion mining on movie and product reviews. In [14], sentiment analysis techniques on webpages and blogs were explained. Pang et al. perfomed an elaborate research on current opinion mining techniques [15]. Reports indeed suggests that real time events have an emphasis on the public opinions in Twitter. This association led to sentiment analysis on blogs and tweets in order to predict the box office revenues for movies [16] and results of elections [17]. These methods gave positive results contributing to display opinion mining as an excellent measure.

III. Methodology for Sentiment Analysis

The below figure depicts the levels to be followed for performing sentiment analysis on Twitter data. Among all the microblogging sites today, Twitter has been of utmost importance because of restriction to the content. Tweet is a message shared on the Twitter. Tweets can include text, hyperlinks, images or videos of upto six seconds. They are made up of three notations namely, account id (@), hashtag (#) and retweet(RT).



A. Data Collection Methods

The most prevalent tweet collection methods are listed below.

- APIs: Twitter facilitates for two kinds of APIs namely Search APIs and Streaming APIs. Search APIs can be used to extract the tweets upto past one week whereas streaming APIs give the tweets that are being posted at the moment.
- Repositories of data like Kdnuggets, Friendster, UCI etc.
- There are several automated and nonautomated tools like Lithium, Topsy etc.

B. Data Preprocessing

The task of mining information from twitter data is complicated since it is just raw data. In order to get accurate results, the raw data needs to be cleaned and preprocessed before the actual opinion mining process is performed. The preprocessing includes the following procedures.

- *Tokenization*: divides the tweet into separate words called as tokens.
- *Filtering*: includes the removal of Twitter notations like accountId (@), retweet (RT) and hashtag (#).
- *Stemming*: Since only textual data is dealt with, the non-textual data such as hyperlinks, URLs and emoticons are eliminated. Stop words like is, was etc that do not contribute at all to the sentiment of the tweet are also removed in order to reduce the huge size of the dataset.
- *Case folding*: Converts all lowercases into uppercases or viceversa to make sure that only the alphabets from a to z are comprised in the processed text.

C. Methods of Feature Extraction

The preprocessed dataset will have several distinct features. In this phase, different characteristics are identified in order to recognize the polarity of entire statement. Some common feature extraction methods are as follows.

- Term Frequency - Inverse Document Frequency: maintains the count of different words and their occurrences. Denotes the weightage of a word in a document.
- Singular Value Decomposition: a process of fragmenting a matrix of words and their occurences thus reducing its size.
- Parts of Speech: identifies nouns, adjectives etc. which are important measures of public opinion.

D. Techniques of Sentiment Classification

There are mainly two categories of techniques for sentiment analysis, namely knowledge based techniques, machine learning techniques.

Knowledge based techniques also known as lexicon based methods. Aggregation of already familiar and collected opinions terms are lexicons. These kind of techniques measure the orientation of the different token in a document from the semantic orientation of the tokens in the document. Semantic Orientation means the power and robustness and polarity of the words or tokens in a text. This methodology is again categorized into corpus based and dictionary based approaches. In the dictionary based approach, dictionaries are created either manually or automatically using base keywords to elongate the list of words. The corpus based approach, on the other hand, generates a list of sentiment terms on the basis of context specific orientations. Additional words related to the respective opinion are added to the expansive corpus. Many of the knowledge-based approaches concentrate on making use of the adjectives as gauge of the textual semantic orientation. A record of adjectives and their corresponding sentiment strength scores are assembled into a dictionary. Then for a specified text, all adjectives are retrieved and elucidated with their scores using the values in the dictionary. They are then summed up to give an overall score for the text.

Machine learning methods make an effective use of past observation of events and knowledge to generate algorithms that can optimize the system performance. All machine learning techniques follow the following steps for opinion mining.

1) Generate a classifier model that is then trained on the training set data. Training set refers to the data that is previously labeled.
2) Categorize the unclassified or unlabeled data on the basis of the trained model.

Techniques of machine learning are further categorized into two types, namely, supervised, unsupervised and reinforcement learning. Supervised learning techniques like classification, regression require training dataset in order to classify unknown test data. These are task driven. Whereas unsupervised learning techniques are data driven. Reinforcemrnt techniques unlike supervised and unsupervised learning, learn to react to the environment. Of these, supervised learning techniques are the best suitable for opinion mining and analysis. Many algorithms related to supervised machine learning like Nave Bayes, Random Forest, Support Vector Machine (SVM), Maximum Entropy etc. are being used for sentiment analysis. Choosing one among them based on the size of the dataset and other requirements is essential for proper results.

IV. Supervised Machine Learning Algorithms For Opinion Mining

For most of the research studies, supervised machine learning approaches are used for variable datasets, handling the outliers with a steady speed regardless of the size of the input dataset.

A. Naive Bayes

Nave Bayes classifier is a simple but robust probabilistic classifier that leverages the concept of mixture models to perform categorization. In this process, proportions of the class in a dataset denotes the probability of new input data being classified into certain class. The mixture model depends on the supposition that each of the previously defined classes is one of the constituents of the mixture alone. These components represent the probability of being related to the specific constituent. This method gives an accurate and efficient result, particularly when there are high number of variables. This classifier utilizes the concept of Bayes Theorem which is based on probability and finds maximum benefit of probability of any token fitting into a specific given or previously defined category. The probability P is deduced as below:

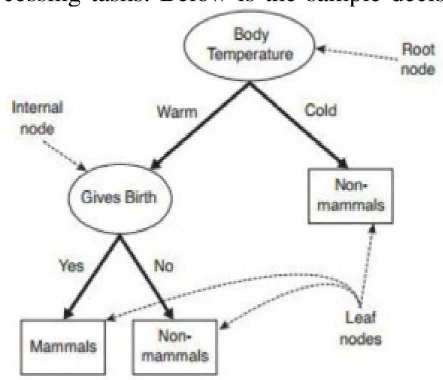
$$P\ (Xi \mid c) = \frac{Count\ of\ Xi\ in\ document\ of\ class\ c}{Total\ no\ of\ words\ in\ document\ of\ class\ c}$$

where X_i is a given term and c is a previously defined class label. During the training phase, the counts of the word occurrences are collected and stored in the hash tables. Naive Bayes approach suffers from a supposition that the features are independent in the feature space. Based on the definition of probability, the document d is categorized into class c using below equation.

$$c *= argmax\ P\ (c \mid d)$$

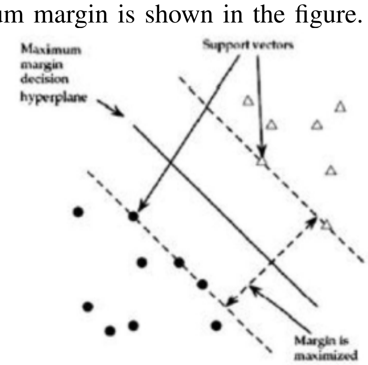
B. Random Forest

Random Forest classifier is a tree-based classifier similar to decision tree. It differs from decision tree in that it contains more number of classification trees that can be used to estimate the class label for a given data point on the basis of the explicit dependent variable. For each data point, each tree contributed for a specific class label and the one getting maximum contribution will be set to the datapoint. The association between any two trees that is the correlation decides the error rate of the classifier model, also the usability of each tree in the forest. The correlation should be very less in order to reduce the error rate. The tree structure comprises of root, internal, and terminal nodes. The internal nodes in the tree denote features whereas the edges that leave the node denote the tests on the weights of the features and the leaves denote different classes. It first does categorization from the root node and moves incrementally downward until we reach a leaf node. The document is then categorized in the class that labels the leaf node. This algorithm finds its use in several language processing tasks. Below is the sample decision tree.



C. Support Vector Machine

Support Vector Machine (SVM) algorithm is a fantastic technique for classification and regression. It is classified to be under supervised learning, that means it involves already classified dataset as for training the classifier model. Geometrically, SVM is considered a hyperplane that divides a dataset into two groups of data, i.e. positive and negative sentiments. We chose hyperplane such that there is maximal distance between the hyperplane and the support vectors of both groups on either side of the hyperplane considering it to be linear. This algorithm will explore the optimal function to divide the two sets of data. A sample linear hyperplane with the maximum margin is shown in the figure.



It generally outperforms Naive Bayes in text classification. If a tweet is denoted as t and hyperplane as h, two classes positive and negative using a set C_j 1, -1 into which the tweet needs to be classified, the resultant hyperplane is the required sentiment, calculated as follows.

$$\overrightarrow{h} = \sum i \; ai \; Ci \; \overrightarrow{tj}, \qquad ai \geq 0$$

D. Maximum Entropy

The maximum entropy depends on estimation technique using probability distribution to perform categorization. First, the classified feature sets are transformed into fixed feature vectors using any of the encoding schemes. Then, we use the encoded vector to calculate the weights for the retrieved characteristics which will encourage together in finding out the most suitable class for a given feature set. This technique finds its utilization in several natural language processing tasks like text classification. Maximum Entropy runs on a basic primary concept that whenever sufficient information is not available regarding the dataset, the distribution must be highly consistent. This fact removes the chance of non-consistent distribution. The probability of uniform distribution is inferred from the classified training dataset and represented as the expected values of extracted feature vectors as shown below.

$$P\,(c \mid d) = \frac{1}{Z\,(d)\{\exp(\sum \lambda i \; fi \; (c, d))\}}$$

V. Application Scenario

The traditional methods of analyzing sentiment can only classify the data based on positive, negative and neutral. Since this is not enough to know the sentiment variation causes correctly, the proposed method enables us to categorize the tweets into more accurate emotions so that we can analyze the reasons behind sentiment variations in a better way. A supervised machine learning algorithm namely, Support Vector Machine is used to categorize the dataset based on text classification. Below is a brief description of the implementation of the method.

1) *Data Collection:* Twitter Streaming API is used to collect data based on a keyword or hashtag inputted by the user. Once they keyword is given, it uses the tweepy API to fetch upto 200 tweets currently posted on Twitter and are added to a csv file and saved.
2) *Data Preprocessing:* Once the raw data is ready, all the unusable content such as stopwords, emoticons, URLs, hyperlinks, images, videos, hashtags, retweets, accountIds etc. are eliminated using respective functions.
3) *Training:* For training purpose, TextBlob package for Python is used. Feature Vectors for the tweets are extracted and are compared with an online corpus avaliable with English words and their impact values to measure the polarity of the tokens in each tweets to identify the sentiment of the tweet.This is then trained using Support Vector Machine.
4) *Labeling and Classification:* We label the classfied dataset based on the category of category to which they belong and on the basis of the calculated polarity and the cateory, we label the trained dataset.
5) *Testing:* Once labeling and classification is over, sample tweets can be used to test the SVM classifier and to categorize the new tweets.

A. Pseudo Code of the main algorithm

For the present method, Support Vector Machine is used to categorize the dataset because SVM is known to facilitate accuracy theoretically for a large set of data while trying to classify it into several classes. Though it takes little longer time to train, it is fast in testing the unknown data and prediction with correctness. Since we have the maximum marginal hyperplane determined in SVM, separation of two weighted input data points, here tweets with polarity, is maximum and prevents any overlap.

Input: A dataset A = a_1, a_2, a_3,..., a_x containing 'x' number of Tweets extracted from Twitter based on user inputted keywords.

Output: Categorization of the dataset A into 'y' categories b_1, b_2, b_3,..., b_y

Proposed Algorithm (T, n)

BEGIN

1) for i = 1 to x tweets
2) Transform a_i into even case either lower case or higher.
3) Eliminate URLs, hashtags, accountIds,emoticons hyperlinks and stop words from a_i
4) Compress the stretched words in a_i
5) Decompress the slang words in a_i
6) Create b_1, b_2,..., b_y categories into twitter corpus actively using lexicon method
7) Use SVM to classify 'x' Tweets into their corresponding classes to which they belong.

END

In the above pseudocode, x number of tweets based on a user inputted keyword are collected. The aggregated data is then preprocessed using the steps 2 to 5. During this phase, the tweets are transformed into uniform casing either lower case or upper case. After this, the Twitter representations like hashtags, retweets and account Id are eliminated followed by the removal of URLs, hyperlinks and emoticons. It is required to get rid of all non-alphabet data as we need only textual data. Stop words with no significance on emotions are discarded followed by decompression of adjectival slang words and compression of elongated words. In step 6, a corpus is produced that includes all tokens possible belonging to various distinguished classes. The classes creation process is made dynamic by developing the corpus using lexicon method. In step 7, supervised machine learning classifier called SVM is applied to categorize the x number of tweets into their corresponding classes. Suppost Vector Machine facilitates higher accuracy as it correctly identifies the class of the input data point. Also it is known for its speed in classification and robustness when huge datasets are taken.

The classified tweets can then be used for a better revelation of the reasons behind sentiment variation using Latent Dirichlet Allocation model-based algorithms like Foreground and background LDA and Reason Candidate and Background LDA[11].

B. Project Framework

The procedure framework of this methodology has various files coded in Python each performing individual functionalities. gettwitterinput.py has the main function. It conveys user to enter a keyword or a hashtag for which the corresponding tweets are retrieved from Twitter using APIs. It also includes verification and validation details like consumer key, consumer secret, access key and access token secret. There are calls to other functions/methods of the program. preprocessing.py has the methods that account for removal of unusable words, hashtags, emoticons etc. from each tweet and does the preprocessing. polarity.py measures the polarity of each tweet to allot it to a particular class of sentiment. svmdem.py includes training of the Support Vector Machine followed by testing. stopwords.txt consists of the stop words which must be eliminated. The extracted tweets from the Twitter API are saved into feeds.csv file. The adjectives and their impact values are stored in and used from AFINN-111.txt.

C. Code Structure

The code structure explains the order in which various functionalities are executed one by one with their nomenclature. It begins with the main method that elicits user to input the keyword and then makes function calls to clean the data, preprocess the cleaned data, extract feature vectors and lastly to calculate sentiment and polarity of each tweet in the tweets file. Then it generates call to SVM training function. get_tweets function gets the tweets and traverses them. Cleaning function eliminates hyperlinks, URLs, emoticons and other special characters from each tweet using regular expression libraries. get_tweet_sentiments function assigns the sentiment using textblob's opinion mining method. getStopWordsList function makes use of the stopwords.txt file to eliminate the words that do not contribute to any emotion and are considered to be insignificant for further analysis. RT that denotes retweet and url are appended to eliminate retweets and urls also.

ProcessTweet function transforms tweets to lower case here and exchanges links with URLs and user with AT_USER, eliminates extra blank or white spaces including hashtags. getFeatureVector function contributes to the generation of feature vector which consists of separating the tweet into words and eliminate punctuations if there are any and eliminate stop words from the tweet. replaceTwoOrMore function searches for two or more occurrences of a character or alphabet. If there are such repetitions of a word like happpyyyy, substitute it with happy. calculatescores function calculates the polarity of each tweet from the extracted feature vector. Then it measures any similarity betweeen the entities of the feature vector and those that are present in the corpus. If there are any comparisons, the imapct values for that element are summed up. This is repeated for all the elements in the feature vector. trainsvm function first assigns different emotions with the help of sentiment and polarity values of each tweet,then trains the Support Vector Machine. Finally, the classifier model is tested by asking the user to enter any tweet and view the prediction value.

VI. Interpretation of Results

For this application, result explanation is simple. When user inputs a keyword, then tweets undergo the process and based on that, it reveals the sentiment and polarity. Then the user is asked to enter a tweet whose sentiment and emotion is predicted using the model. The percentage of tweets based on their emotions is also presented.

```
Percentage of emotions detected in the training set :
--------------------------------------------------------
Disturbed tweets percentage: 44 %
cheerful tweets percentage: 16 %
anxious tweets percentage: 8 %
over the clouds tweets percentage: 5 %
Needs help tweets percentage: 4 %
Excited tweets percentage: 4 %
happy tweets percentage: 4 %
frown tweets percentage: 3 %
sad tweets percentage: 2 %
deeply depressed tweets percentage: 1 %
Glad tweets percentage: 2 %
Frown tweets percentage: 3 %
Chill tweets percentage: 1 %
Disappointed tweets percentage: 0 %
Neutral tweets percentage: 0 %
Training done.
```

A. Comparison with Sentiment140 tool

Sentiment140 is a web based tool that permits users to determine the sentiment of a topic, brand or product on Twitter. It is built based on Maximum Entropy machine learning algorithm. It uses around 160,000 tweets aggregated automatically along with emoticons to train the classifier model. Then on the basis of the classifiers outputs, it will decide the sentiment label either positive, neutral or negative with the maximum probability as the sentiment label of a tweet. Positive tweets are shown in green color, negative in red and neutral in white on this site. The results from the proposed method in this paper are compared with the results from Sentiment140 tool and since our method gives us a deeper understanding of the tweets with correctness, it gave positive results. Below is the result of a tweet ”mikelyon12: @SeymourTribune SMS students doing work! Get it y'all! I'm pretty sure I see some FCA peeps in there.” based on this tool.

> mikelyon12: @SeymourTribune SMS **students** doing work! Get it y'all! I'm pretty sure I see some FCA peeps in there.
> Posted: 37 seconds ago

Here it is classified as positive. Though it is correct, it doesn't give more information. For the same tweet, when our method was applied, the following result is extracted.

```
Enter a tweet message to find its sentiment polarity:
@SeymourTribune SMS students doing work! Get it y'all!
 I'm pretty sure I see some FCA peeps in there.
Calculating Polarity of your tweet....
Emotion Analysed:
['cheerful']
```

This shows that the tweet has been identified as ”cheerful” which is in fact, correct and gives the exact feeling of the person who tweeted about the students. Our method gives more descriptive results, classifies the result into more than three categories.

VII. Conclusion and Future Work

In this paper, an unconventional method of opinion mining that is very different from the conventional sentiment analysis techniques is proposed. To make opinion mining process more descriptive, qualitative analysis has also been consolidated along with text mining methods. This process categorizes Twitter data into their corresponding respective classes rather than just discovering positive or negative data. Also the proposed algorithm makes the classes creation process effective and active that removes the necessity to modify the algorithm whenever new data is added. This faciity enhances the accuracy and robustness of the classifier model. Future improvements to this algorithm can be performed by presenting more labels in the multiclass emotion categorization. The algorithm can also be further developed by taking emoticons into consideration as well since they add a lot of impact to any usual public opinion. Emoticons give prominence to the tweets much and can be useful to classify the public sentiment in a more accurate way.

References

[1] X. Chen, M. Vorvoreanu and K. Madhavan, *Mining Social Media Data to Understand Students Learning Experiences*, IEEE Transaction, 2014, vol. 7, no. 3, pp. 246-259

[2] N. Kasture and P. Bhilare, *An Approach for Sentiment analysis on social networking sites*, Computing Communication Control and Automation (ICCUBEA), 2015, pp. 390-395.

[3] M. S. Neethu and R. Rajasree, *Sentiment Analysis in Twitter using Machine Learning Techniques*, in 4th Int. Conf. of Computing, Communications and Networking Technologies (ICCCNT), 2013, pp. 1- 5.

[4] S. Bahrainian and A. Dangel, *Sentiment Analysis using Sentiment Features*, in Int. joint Conf. of Web Intelligence and Intelligent Agent Technologies, 2013, pp. 26-29.

[5] G. Gautam and D. Yadav, *Sentiment analysis of twitter data using machine learning approaches and semantic analysis*, in 7th Int. Conf. on Contemporary Computing, 2014, pp. 437-442.

[6] V. Sehgal and C. Song, *SOPS: Stock Prediction Using Web Sentiment*, in 7th IEEE Int. Conf. on Data Mining Workshop, 2007, pp. 21-26.

[7] Z. Kechaou, B. M. Ammar and A. M. Alimi, *Improving e-learning with sentiment analysis of users' opinions*, in Global Engineering Education Conference (EDUCON), 2011, pp. 1032-1038.

[8] A.C.E.S Lima. and L.N. de Castro, *Automatic sentiment analysis of Twitter messages*, in 4th Int. Conf. on Computational Aspects of Social Networks (CASoN), 2012, pp. 52-57.

[9] R. Batool, A. M. Khattak, J. Maqbool and S. Lee, *Precise tweet classification and sentiment analysis*, in 12th Int. Conf. on Computer and Information Science (ICIS), 2013, pp. 461-466.

[10] Z. WANG, V. J. Chuan TONG, X. XIN and H. C. CHIN, *Anomaly Detection through Enhanced Sentiment Analysis on Social Media Data*, in 6th International Conference on Cloud Computing Technology and Science, 2014, pp. 918-922.

[11] Shulong Tan, Yang Li, Huan Sun, Ziyu Guan, Xifeng Yan, Member, IEEE, Jiajun Bu, Member, IEEE, Chun Chen, Member, IEEE, and Xiaofei He, Member, IEEE, *Interpreting the Public Sentiment Variations on Twitter*, IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 5, May 2014

[12] M. Hu and B. Liu, *Mining and summarizing customer reviews*, in Proc. 10th ACM SIGKDD, Washington, DC, USA, 2004.

[13] L. Zhuang, F. Jing, X. Zhu, and L. Zhang, *Movie review mining and summarization*, in Proc. 15th ACM Int. Conf. Inform. Knowl. Manage., Arlington, TX, USA, 2006.

[14] W. Zhang, C. Yu, and W. Meng, *Opinion retrieval from blogs*, in Proc. 16th ACM CIKM, Lisbon, Portugal, 2007.

[15] B. Pang and L. Lee, *Opinion mining and sentiment analysis*, Found. Trends Inform. Retrieval, vol. 2, no. (12), pp. 1135, 2008.

[16] G. Mishne and N. Glance, *Predicting movie sales from blogger sentiment*, in Proc. AAAI-CAAW, Stanford, CA, USA, 2006

[17] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, *Predicting elections with twitter: What 140 characters reveal about political sentiment*, in Proc. 4th Int. AAAI Conf. Weblogs Social Media, Washington, DC, USA, 2010.

[18] Mitali Desai and Mayuri A. Mehta, *A Hybrid Classification Algorithm to Classify Engineering Students' problems and perks*, International Journal of Data Mining.