Sentiment Analysis of tweets Using Support Vector Machine and Naïve Bayes Classifier

CS 5824

Aman Sarawgi
Virginia Polytechnic Institute and State University
Department of Computer Science and Application

1. Abstract

Twitter is a social media platform where users from all over the world share their thoughts in the form of tweets, it could be text messages or photos. In this project, only text messages will be considered. Two approaches will be used to automatically classify the sentiment of tweets on Twitter. These messages are classified as either positive or negative with respect to certain key words. This is useful in various fields where one can classify sentiment with respect to movie reviews or products[10,12]. There has been some previous research on classifying sentiment of messages on microblogging services like Twitter [1,6,11]. The results of machine learning algorithms for classifying the sentiment of Twitter messages will be obtained using Support Vector Machines[2,3] and Naïve Bayes Classifier[4]. The training data consists of Twitter messages, from which certain key words will be used to label the messages as positive or negative. This type of training data is abundantly available and can be obtained through automated means. The machine learning algorithms which will be used will be compared and see what techniques can be used to maximize the accuracy. This paper also describes the preprocessing steps needed to achieve high accuracy. The main contribution of this paper will be to compare Support Vector Machines and Naïve Bayes Classifier and how its accuracy gets affected by adjusting the attributes.

2. Specific Aims

The aim of this project is to classify the tweets into two categories i.e., positive and negative. We will implement the two machine learning algorithms as mentioned in [2,3,4]. The main aim for this project is to compare the performance of these two models and study various factors that help increase the accuracy. The techniques used to preprocess the data were stemming and lemmatization. Models will be built based on each technique independently to see which factors can help increase the performance.

This project can further be used to categorize the tweets into more sentiments like anger, depression, threat etc.

3. Background

Twitter is an online social media platform to share news, messages and micro blogging where people communicate in short messages called tweets. In 2019 there was 330 million

active users. In today's date, Twitter has become an important medium for political conversation, protest and for giving personal views so the possibilities for harm increased exponentially. Twitter popularity is giving rise to new spam marketplace.
Any user can send tweets which can be viewed by everyone and it's a global platform for people to discuss about important topics. Tweets have ranking, which is based on many individual attributes, the most important attribute is the number of followers the user has. These tweets appear at the top and get the most views, thus it is a very influential tweet. Tweets about a particular topic can be retrieved using the twitter's real time search engine. Tweets are different from long texts such as movie reviews as it has a 140-character limit. Movie reviews are usually longer, summarizing the viewers opinions. Another key difference is reviews are a tad bit formal, well formatted and structured as opposed to the casual and on the go status update nature of tweets. Tweets are however a good source of gathering feedback for products and services by corporations and could offer exciting new avenue in contrast to corporate feedback surveys.
In this project we are using machine learning algorithms to classify if the tweet is positive or negative which becomes very important to protect users from negative tweets which may influence them in a wrong way.

4. Research Design and Method

This paper is based on analyzing sentiment of tweets using Support Vector Machine and Naïve Bayes Classifier. I have worked on implementing Support Vector Machines and my partner, Dvijen Trivedi has worked on implementing Naive Bayes Classifier. Accuracy for both the models have been compared to find a which classifier performs better for the problem statement. The underlying approach (feature reduction and pre-processing etc.) is the same as the dataset used for training and testing classifiers should be uniform. This is done to ensure that model performance and accuracy for Naive Bayes Classifier and Support Vector Machines is directly comparable to one another. Supervised learning is used as the class labels for the positive or negative sentiments are known in the dataset.

4.1 Datasets

The dataset consists of 1.6 million tweets in the record database.
It contains the following 6 fields:

1. **target**: the polarity of the tweet (*0* = negative, *4* = positive)
2. **ids**: The id of the tweet (*1467811592*)
3. **date**: the date of the tweet (*Mon Apr 06 22:20:03 PDT 2009*)
4. **flag**: The query (*lyx*). If there is no query, then this value is NO_QUERY.
5. **user**: the user that tweeted (*mybirch*)
6. **text**: the text of the tweet (Need a hug)

```
[ ]  1 df_twt_data.head()
```

	target	ids	date	flag	user	text
0	0	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, that's a bummer. You shoulda got David Carr of Third Day to do it. ;D
1	0	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by texting it... and might cry as a result School today also. Blah!
2	0	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Managed to save 50% The rest go out of bounds
3	0	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
4	0	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all. i'm mad. why am i here? because I can't see you all over there.

Figure 1: Snapshot of Dataset

```
[ ]   1 df_twt_data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1600000 entries, 0 to 1599999
Data columns (total 6 columns):
 #   Column  Non-Null Count    Dtype
---  ------  --------------    -----
 0   target  1600000 non-null  int64
 1   ids     1600000 non-null  int64
 2   date    1600000 non-null  object
 3   flag    1600000 non-null  object
 4   user    1600000 non-null  object
 5   text    1600000 non-null  object
dtypes: int64(2), object(4)
memory usage: 73.2+ MB
```

Figure 2: Snapshot of Attribute Data Type

4.2 Features

Feature Extraction is a very important step in building any classification model and hence for SVM [5,9]. A good set of features i.e., highly informative features lead to a classifier with high accuracy. Following describes the features used to train the SVM for sentiment analysis of tweets.

4.2.1 Stemming and Lemmatization

Stemming and lemmatization play an important role in order to increase the efficiency of an information retrieval system[7,8]. The basic principle of both techniques is to group similar words which have the same root. Stemming algorithms remove suffixes as well as inflections, so that word variants can be reduced to their respective stems. If we consider the words amusing and amusement, the stem will be amus. Lemmatization uses vocabularies and morphological analyses to remove the inflectional endings of a word and to convert it in its dictionary form. Considering the example taken earlier, the lemma for

amusing and amusment will be amuse. Stemmers and lemmatizers differ in the way they are built and trained. Statistical stemmers are important components for text search over languages and can be trained even with few linguistic resources. Lemmatizers can be generic or optimized for a specific domain.

	target	ids	date	flag	user	text	lemma_text	stem_text
0	NEGATIVE	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, that's a bummer. You shoulda got David Carr of Third Day to do it. ;D	awww bummer shoulda got david carr day d	awww bummer shoulda got david carr third day
1	NEGATIVE	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by texting it... and might cry as a result School today also. Blah!	upset update facebook texting cry result school today blah	upset updat facebook text might cri result school today also blah
2	NEGATIVE	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Managed to save 50% The rest go out of bounds	dived times ball managed save 50 rest bounds	dive mani time ball manag save 50 rest go bound
3	NEGATIVE	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire	body feels itchy like fire	whole bodi feel itchi like fire
4	NEGATIVE	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all. i'm mad. why am i here? because I can't see you all over there.	behaving mad	behav mad see

Figure 3: Conversation of tweets to Stemming Texts

	target	ids	date	flag	user	text	lemma_text
0	NEGATIVE	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, that's a bummer. You shoulda got David Carr of Third Day to do it. ;D	awww bummer shoulda got david carr day d
1	NEGATIVE	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by texting it... and might cry as a result School today also. Blah!	upset update facebook texting cry result school today blah
2	NEGATIVE	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Managed to save 50% The rest go out of bounds	dived times ball managed save 50 rest bounds
3	NEGATIVE	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire	body feels itchy like fire
4	NEGATIVE	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all. i'm mad. why am i here? because I can't see you all over there.	behaving mad

Figure 4: Conversation of tweets to Lemmatization Texts

4.3 Support Vector Machine

SVMs are set of related supervised learning methods used for classification and regression. They belong to a family of generalized linear classification. A special property of SVM is , SVM simultaneously minimize the empirical classification error and maximize the geometric margin.

SVM maps the input vector to a higher dimensional space where a maximal separating hyperplane is constructed. We draw two parallel hyperplanes on each side of the separating hyperplane. The separating hyperplane is the hyperplane that maximize the distance between the two parallel hyperplanes.

Parallel hyperplanes can be described by equation -

$w.x + b = 1$

$w.x + b = -1$

To excite data points, we need to ensure that for all I either -

$w.\ xi - b \geq 1$ or $w.\ xi - b \leq -1$

This can be written as -

$yi\ (w.\ xi - b) \geq 1\ ,\ 1 \leq i \leq n$

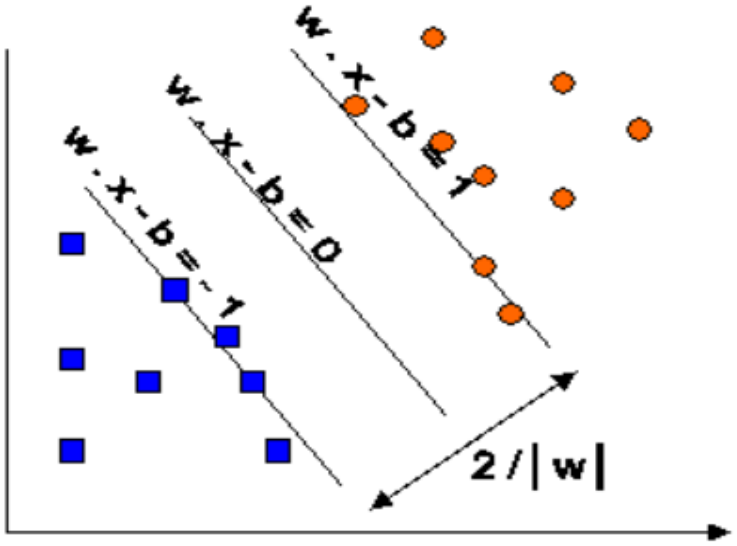

Figure 5: Maximum margin hyperplanes for a SVM trained with samples from two classes

The classification will be done as the data is split by the hyperplane, thus prediction of the sentiment of tweets should be done accurately. This will be checked and compared with the performance of Naïve Bayes Classifier.

5. Experiments Design

The dataset was reduced to 200,000 data values to compensate for the large computational time SVM takes to build a model. The dataset consisted of 100,000 positive sentiments and 100,000 negative sentiments. The dataset for partitioned into 160,000 training instances and 40,000 testing instances. The SVM and Naïve bayes classifier models were built on this dataset. The accuracy was calculated for the two different preprocessing techniques i.e., stemming and lemmatization. The accuracy of each model of each method was compared to see which techniques suitable and which model performs better under these varying factors.

target	ids	date	flag	user	text
NEGATIVE	100000	100000	100000	100000	100000
POSITIVE	100000	100000	100000	100000	100000

Figure 6 : Distribution of the Data

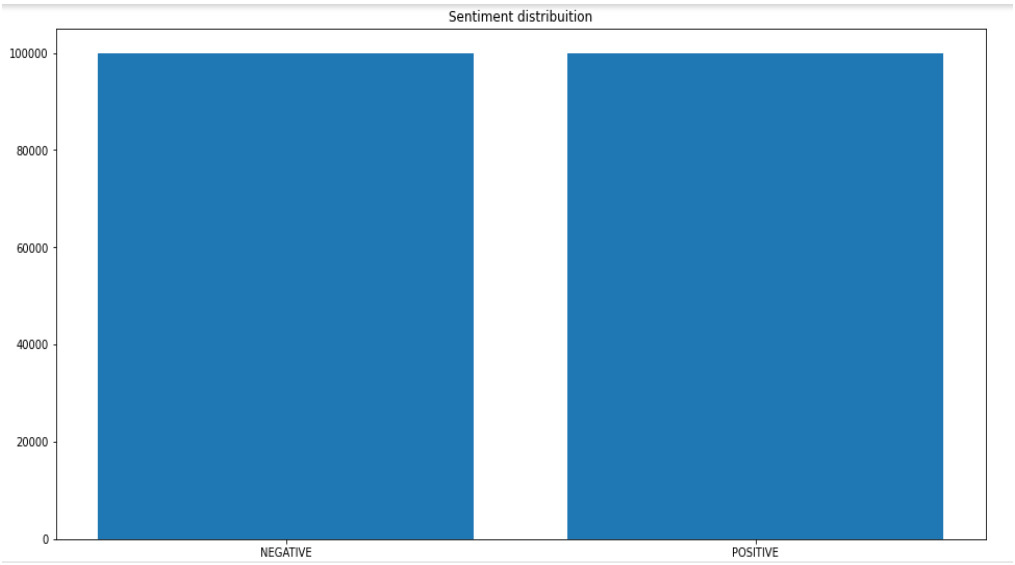

Figure 7: Graphical Representation of the Distribution of the Data

6. Results and Discussions

The model that has being built by Support Vector Machines help classify the dataset into positive and negative sentiments. Two techniques have been applied i.e., Stemming and lemmatization and the performance of the model for each technique is reported below.

Performance of model for Stemming texts –

Accuracy Score – 0.760975

Precision Score – 0.751128585990713

Recall Score – 0.7772754671488848

F1 Score – 0.7639783751758868

From the below figures we can see the performance of the model –

	precision	recall	f1-score	support
NEGATIVE	0.77	0.74	0.76	20092
POSITIVE	0.75	0.78	0.76	19908
accuracy			0.76	40000
macro avg	0.76	0.76	0.76	40000
weighted avg	0.76	0.76	0.76	40000

Figure 8: Classification Report of model for Stemming Texts

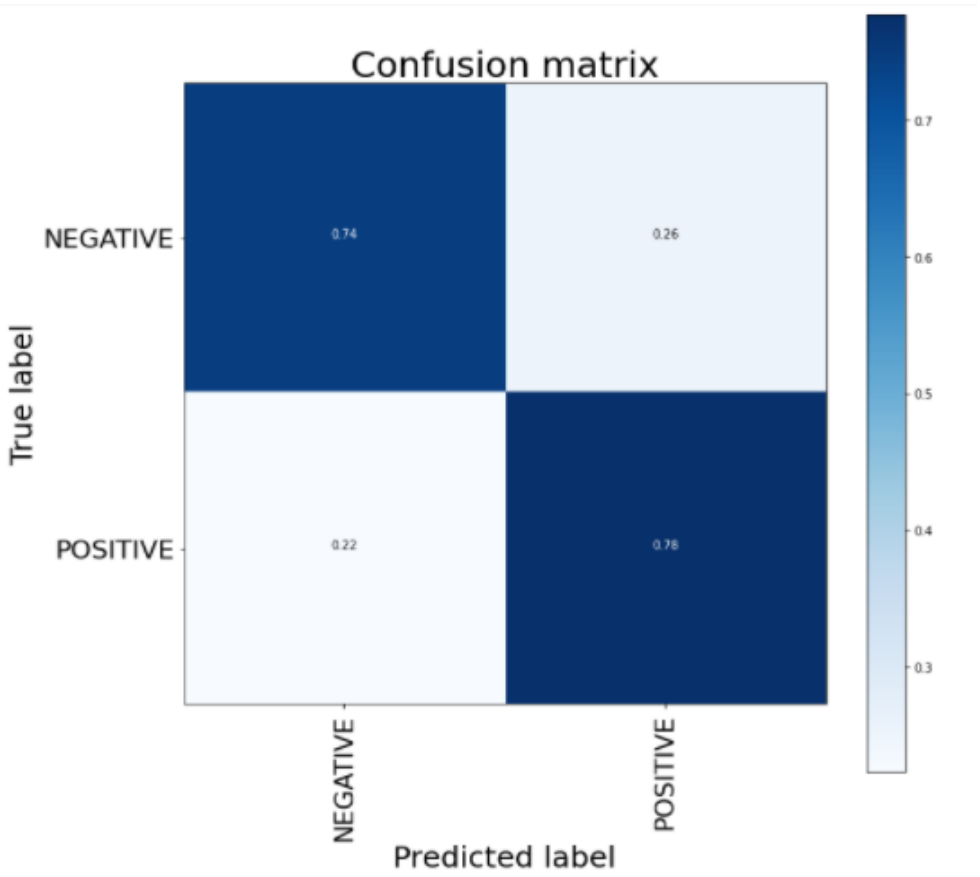

Figure 9: Confusion Matrix of model for Stemming Texts

Performance of model for Lemmatization texts –
Accuracy Score – 0.75975
Precision Score – 0.7484558965450685
Recall Score – 0.7791340164757886
F1 Score – 0.7634869068714315
From the below figures we can see the performance of the model -

	precision	recall	f1-score	support
NEGATIVE	0.77	0.74	0.76	20092
POSITIVE	0.75	0.78	0.76	19908
accuracy			0.76	40000
macro avg	0.76	0.76	0.76	40000
weighted avg	0.76	0.76	0.76	40000

Figure 10: Classification Report of model for Lemmatization Texts

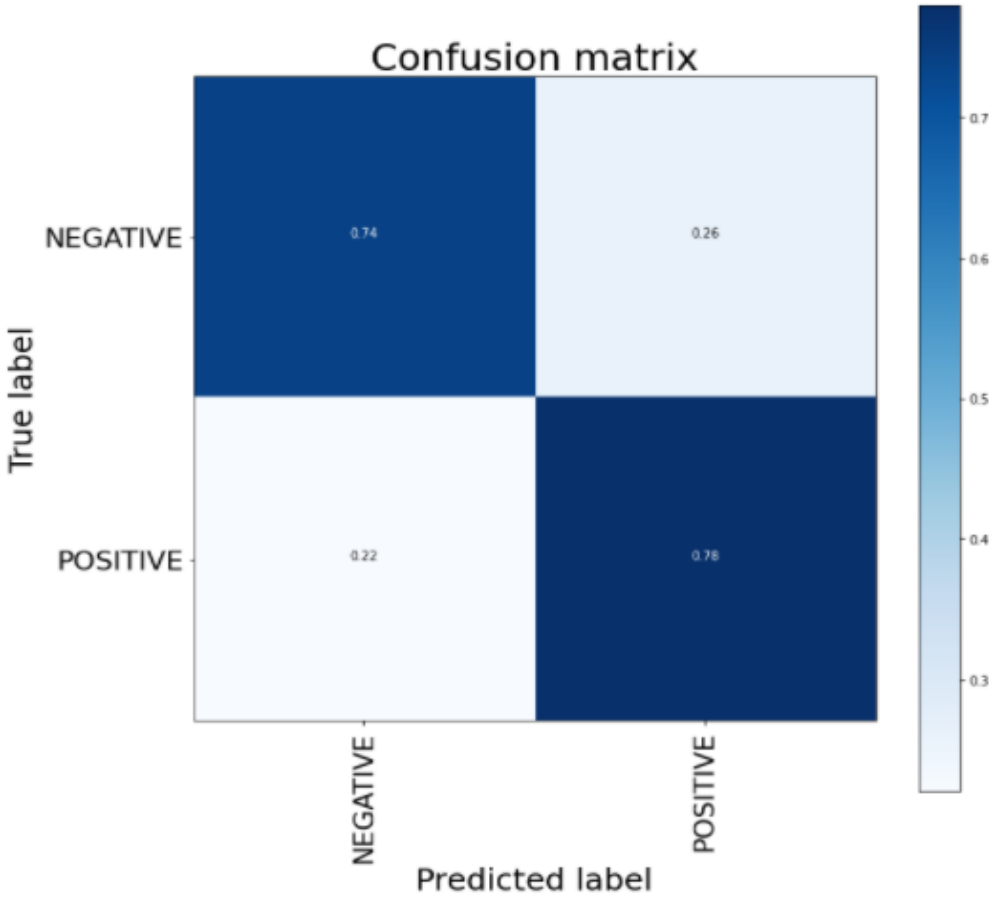

Figure 11: Confusion Matrix of model for Lemmatization Texts

The model performs relatively well while using both approaches. But the computational time taken the build the model is huge and thus it is not the most optimal way to analyze the sentiments of tweets. With the complexity of using SVM known, it is expected to generate a better accuracy rate while classifying this dataset.

7. Evaluation of the two Models

Performance of model built by Naïve Bayes Classifier for Stemming texts –
Accuracy Score – 0.7585
Precision Score – 0.7771574413782102
Recall Score – 0.7277072442120985
F1 Score – 0.7516198704103672

Performance of model built by Naïve Bayes Classifier for Lemmatization texts –
Accuracy Score – 0.76075
Precision Score – 0.7786020878596789
Recall Score – 0.7315409509584266
F1 Score – 0.7543382277441215

Here we can see the results of the performance of the two models for both the pre-processing techniques. The result suggests that both the models have a similar accuracy for this dataset, but the complexity of the models vary drastically. The classifications report received and quite good and can be further used for sentimental analysis on other datasets.

8. Conclusion

In this project, the main aim was to build a model for the dataset available to maximize the accuracy for which Support Vector Machines and Naïve Bayes Classifier was used. Two techniques were used to pre-process the data and models were built independently for stemming texts and lemmatization texts. The accuracy generated by the models ranged between 75-76%, which is quite good, but when we compare the two models, SVM took a long time to build a model and generate results as it is a complex algorithm and on the other hand Naïve Bayes classifier generated results very quickly with almost the same accuracy. Based on the Principle of Parsimony, we can conclude that Naïve Bayes classifier is the most suitable classification model for this dataset as it is the simpler model and generates good results. The result of this project could vary for different datasets, but overall Naïve Bayes Classifier should be the preferred model for document-based datasets.

9. References

[1] Alec Go, Richa Bhayani and Lei Huang, "Twitter Sentiment Classification using Distant Supervision", Stanford, 2009.

[2] N. Cristianini and J. Shawe-Taylor, "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods", Cambridge University Press, March 2000.

[3] D. O. Computer, C. wei Hsu, C. chung Chang, and C. jen Lin, "A practical guide to support vector classification", Technical report, 2003.

[4] Pouria Kaviani and Mrs. Sunita Dhotre, "Short Survey on Naive Bayes Algorithm", International Journal of Advance Engineering and Research Development *Volume 4, Issue 11, November -2017*

[5] Symeon Symeonidis, Dimitrios Effrosynidis and Avi Arampatzis, "A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis", Expert Systems with Applications Volume 110, 15 November 2018.

[6] Bhumika Gupta, Monika Negi, Kanika Vishwakarma, Goldi Rawat and Priyanka Badhani, "Study of Twitter Sentiment Analysis using Machine Learning Algorithms on Python", International Journal of Computer Applications (0975 – 8887) Volume 165 – No.9, May 2017

[7] Fajri Koto and Mirna Adriani, "A Comparative Study on Twitter Sentiment Analysis: Which Features are Good", International Conference on Application of Natural Language to Information Systems. NLDB 2015: Natural Processing and Information Systems pp 453-457. 04 June 2015

[8] Swati Sharma and Mamta Bansal, "Stemming and Lemmatization of Tweets for Sentiment Analysis using R", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-2, July 2019.

[9] Tajinder Singh and Madhu Kumari, “Role of Text Pre-Processing in Twitter Sentiment Analysis”, Procedia Computer Science, 2016.
[10] Nehal Mamgain, Ekta Mehta, Ankush Mittal and Gaurav Bhatt, “Sentiment Analysis of Top Colleges in India Using Twitter Data”, (IEEE) ISBN -978-1-5090-0082-1, 2016.
[11] Neethu M S and Rajasree R, “Sentiment Analysis in Twitter using Machine Learning Techniques”, IEEE – 31661, 4th ICCCNT 2013.
[12] A. Tripathy, A. Agrawal and S. K. Rath, "Classication of Sentimental Reviews Using Machine Learning Techniques", *Procedia Computer Science*, vol. 57, pp. 821-829, 2015.