REAL TIME SENTIMENT CLASSIFICATION OF TWITTER DATA USING SVM, NAIVE BAYES & CART ALGORITHM

MD Saddam Hussain
Department of Computer Science
School of Research & Technology
People's University, Bhopal,India

Shital Gupta
Associate Professor & Head
Department of Computer Science
School of Research & Technology
People's University, Bhopal,India

***Abstract*— The data which comes from e-commerce site is unstructured text data, Text mining is becoming an important field in research for finding valuable information from unstructured texts. Data which contains an unstructured text which stores large amount of valuable information cannot simply be used for further processing by computers. Therefore, we need an exact processing methods, techniques and algorithms in order to extract this meaningful information which is done by using text mining. Classification of these user opinions is an Information Extraction and Natural Language Processing task that classify the user opinions into various categories like in the form of positive or negative. In this paper we can build a classifier's based on Support Vector machine (SVM), Naive Bbayes (NB) and decision tree (CART) classification algorithm to identify the opinions and classify them onto categories and we can compute the performance measure of these classifiers and also compare the classification algorithms performance based on their accuracy and we can say that CART algorithm performed better as compared to SVM and NB. In this we can also classify the real time tweets review into various emotions and polarity through decision tree classification model.**

***Keywords*—web data, text mining, data mining, R, text mining techniques, SVM, Decision tree, classification, Naive Bayes, CART.**

I.INTRODUCTION

In the past years, social platforms have gathered huge attention in day to day life where users publish their views about any concept. This user generated information that shows customer views on a particular topic is very helpful for analysing the interest of customer and it demands efficient techniques to obtain general opinion results. Twitter is now being one of the most common microblogs. Twitter Sentiment Analysis (TSA) handles the issue of analysis of the messages posted on Twitter in context of the sentiment expressed by tweets. The initial step of TSA process incorporates collection of tweets and marking them by their expressed sentiment. The next step involves extraction of feature sets needed for classifier training. The feature selection & combination of features shows a great impact on the performance of the classifier. Both the labelled data and the chosen features are sent to the machine-learning algorithm and utilized to build the classifier model. In the last step, the classifier allots labels to testing data (unlabelled tweets).

Opinion mining and sentiment analysis is one of the most important research topics at our present time. It refers to the study of people's emotions, sentiment or opinions that can be expressed in a written text.

Different companies and organizations trying to find the users opinion about their products, services and so on to them for making better decisions. Moreover, Microblogging platforms such as: Twitter, Facebook, weibo…etc. can help to extract and analyzed user's opinions and reviews. The amount of these information is too large to be analyzed by normal users.

So, to avoid this, sentiment analysis techniques could be used. Twitter is one of the most popular social networks and microblogging platform where it is a convenient way for senders to write and share their opinion about several aspects of life within 140-character length. Twitter has enormous number of text and posts that has grown rapidly. It is a quite difficult to analyze these tweets based on misspellings, emoji, and slang words where it should have a preprocessing step before dealing with it per the polarity detection of positivity or negativity of the tweets.

II. LITERATURE REVIEW

Text mining, also known as text data processing which is a method of extracting attention-grabbing and non-trivial patterns or data from text documents. It uses algorithms to remodel free flow text (unstructured) into information which will be analysed (structured) by applying applied mathematics, Machine Learning and natural language processing (NLP) techniques. Text

mining is an evolving technology that permits enterprises to know their customers well, and facilitate them in redefining client wants. The amount of client reviews and feedback that a product receives has grown up quickly over the time. For a well-liked quality, the amount of review comments is in thousands or more. This makes it trouble for the manufacturer to scan all of them to form a decision in rising product quality and support. Once more trouble for the manufacturer to stay track and to manage all client opinions. This text makes an attempt to derive some significant information from quality reviews which can be utilized in enhancing quality options from engineering purpose and helps in rising the support quality and client expertise.

There is a massive increase in number of people who access various social networking and micro-blogging websites that gives new shape to the impression of today's generation [4]. Several reviews for a specific product, brand, individual personality, forum and movies etc. are very helpful in directing the perception of people. Hence the analysts are commenced to create algorithms to automated classification of distinctive reviews on the basis of their polarities particularly: Positive, Negative and Neutral. This automated classification mechanism is referred as Sentiment Analysis. The ultimate aim of this paper is to apply Support Vector Machine (SVM) classification technique to classify the sentiment and texts for smart phone product review that analyses different datasets used for classification of sentiments and texts. Furthermore, various data sets have been utilized for training as well as testing and implemented using Support Vector Machine (SVM) to investigate polarity of the ambiguous tweets. The experimental work includes three performance features such as Precision, Recall and F-measure. On the basis of these features, the accuracy of the different products has been computed. The obtained result approves high accuracy as predicted on the basis of smart phone reviews.

In [2], Analytics companies develop the facility to support their selections through analytic reasoning using a mathematical techniques. Thomas Devonport in his book titled, “Competing on analytics: The new science of winning”, claims that an enormous proportion of high performed companies have high analytical skills among their personnel. On the alternative hand, a recent study has in addition survey that over 59% of the organizations haven't got data required for decision-making. Learning “Data Analysis with R” not only adds to existing analytics information and methodology, but in addition equips with exposure into latest analytics techniques furthermore as prediction, social media analytics, text mining on. It provides an opportunity to work on real time info from Twitter, Facebook and different social networking sites.

In [3], One in every of the common discussions around an organization is that the preference of one tool over another and thus the various factors like current ability sets procurable at intervals the organization, user capability and capacity of the tool to handle visual capabilities that leads to the selection and utilization of these tools. So as to answer variety of the queries around performance and easy tool usage and image, a comparison between SAS Text mining, Python and R Programming tools was conducted. SAS Text mining could also be a data process tool used for locating patterns across text information through modelling. Python and R programming tools (both open offer tools) unit of measurement used for mathematics analysis and knowledge interpretation.

III PROBLEM DEFINITION

Text mining [7] that help an organization derive potentially valuable business insights from text-based content such as word documents, email and postings on social media streams like Facebook, Twitter and LinkedIn. Data mining or Text mining plays an important role in decision making because through these mining techniques we can analyse the data and on the basis of result we can take a decision.

IV PROPOSED WORK

In this we proposed SVM, Naive Bayes and decision tree classification algorithm to classify the reviews into various categories like positive or negative. We build classifiers based on these classification algorithms such as Decision tree (CART), Naive Bayes, SVM (Support Vector Machine), and trained this model on training data and then apply this model on test data and compute the performance measure of these classification algorithms.

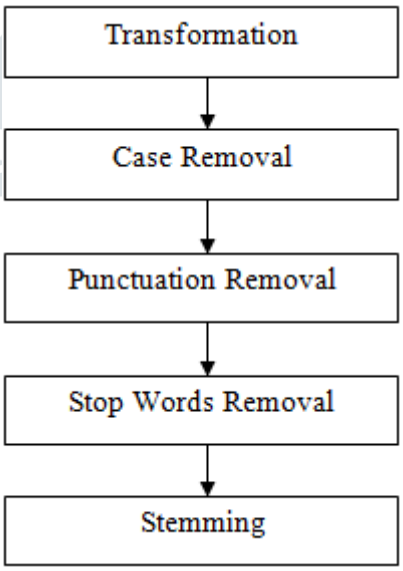


Figure 1. Flow Diagram

Step-1. First, we can collect the text dataset for classification.

Step-2. After collection of data we can pre-processed the data before apply classification algorithm. In this we can label the dataset attributes and transform the data, and also perform various text mining techniques like in this we find the term frequency.

Step-3. After that we can build classifier based on various classification algorithm and apply the train dataset to the models.

Step-4. Then we can compute the performance of these classification algorithm on the test dataset.

Step-5. After computing the performance, we can compare these algorithms based on their performance, And choose the best algorithm for real time classification.

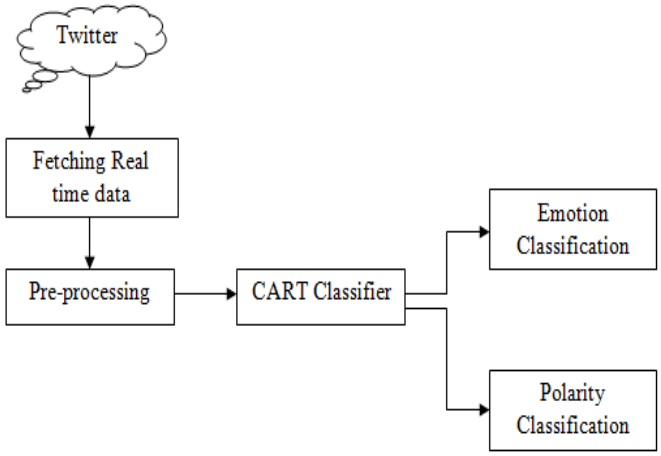


Figure 2. Flow Diagram of Real time classification

Preprocessing:

Raw tweets scraped from twitter generally result in a noisy dataset. This is due to the casual nature of people's usage of social media. Tweets have certain special characteristics such as retweets, emoticons, user mentions, etc. which have to be suitably extracted. Therefore, raw twitter data has to be normalized to create a dataset which can be easily learned by various classifiers. We have applied an extensive number of pre-processing steps to standardize the dataset and reduce its size. We first do some general pre-processing on tweets which is as follows.

• Convert the tweet to lower case.
• Replace 2 or more dots (.) with space.
• Strip spaces and quotes (" and ') from the ends of tweet.
• Replace 2 or more spaces with a single space.

Classifier:

Decision Tree:

Decision trees are popular methods for inductive inference. They are robust to noisy data and learn disjunctive expressions. A decision tree is a k-array tree in which each internal node specifies a test on some attributes from input feature set representing data. Each branch from a node corresponds to possible feature values specified at that node. And every test results in branches, representing varied test outcomes. The decision tree induction basic algorithm is a greedy algorithm constructing decision trees in a top-down recursive divide-and-conquer manner.

The algorithm begins with tuples in the training set, selecting best attribute yielding maximum information for classification. It generates a test node for this and then a top down decision trees induction divides current tuples set according to current test attribute values. Classifier generation stops when all subset tuples belong to the same class or if it is not worthy to proceed with additional separation to further subsets, i.e. if more attribute tests yield information for classification alone below a pre-specified threshold. In this paper, it is proposed to base the threshold measure based on information gain.

Input:

Data partition, D, which is a set of training tuples and their related class labels;
Attribute_list;
Attribute_selection_method, to determine the splitting criterion that "best" partitions.

Output: A decision tree.

Algorithm

Step 1- CREATING A ROOT NODE
1. Create a root node N
2. If tuples in D are all of the similar class, C then
3. Return N as a leaf node label with the class C;
4. If attribute list is empty then
5. Return N as a leaf node label with the majority class in D

Step 2- ATTRIBUTE SELECTION
6. Apply attribute_selection_method(D, attribute_list) to discover the "best " splitting _criterion attribute;
7. Label node N with splitting _criterion;
8. Update the attribute_list

Step 3- SPLIT THE TREE
9. for each outcome j of splitting_criterion
//partition the tuples and produce subtrees for each partition
10.Based on splitting_criterion attribute
Split the tree into two part
12. attach a leaf labeled with the majority class D in node N:
13. else attach the node returned by Generate_decision_tree(Dj:attribute_list)to node N:
end for
14. return N,

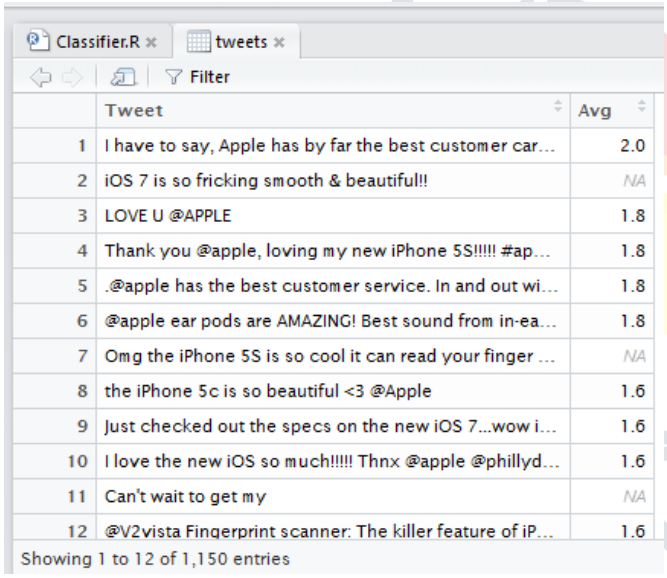
In the proposed feature selection, a Decision tree induction selects relevant features. Decision tree induction is the learning of decision tree classifiers constructing tree structure where each internal node (no

leaf node) denotes attribute test. Each branch represents test outcome and each external node (leaf node) denotes class prediction. At every node, the algorithm selects best partition data attribute to individual classes. The best attribute to partitioning is selected by attribute selection with Information gain. Attribute with highest information gain splits the attribute. Information gain of the attribute is found by

$$\inf o(D) = -\sum_{i=1}^{m} p_i \log_2(p)$$

V EXPERIMENTAL & RESULT ANALYSIS

The experimental and result analysis is done by using intel i5-2410M CPU with 2.30 GHz processor along with 4 GB of RAM and the windows operating system is running. For result analysis we use R and R studio for processing the data and then we load the tweets and performing Sentiment analysis on that collected tweets. So, we can load the data and figure 4 shows he data is loaded.


Classifier.R × tweets ×

Filter

	Tweet	Avg
1	I have to say, Apple has by far the best customer car...	2.0
2	iOS 7 is so fricking smooth & beautiful!!	NA
3	LOVE U @APPLE	1.8
4	Thank you @apple, loving my new iPhone 5S!!!!! #ap...	1.8
5	.@apple has the best customer service. In and out wi...	1.8
6	@apple ear pods are AMAZING! Best sound from in-ea...	1.8
7	Omg the iPhone 5S is so cool it can read your finger ...	NA
8	the iPhone 5c is so beautiful <3 @Apple	1.6
9	Just checked out the specs on the new iOS 7...wow i...	1.6
10	I love the new iOS so much!!!!! Thnx @apple @phillyd...	1.6
11	Can't wait to get my	NA
12	@V2vista Fingerprint scanner: The killer feature of iP...	1.6

Showing 1 to 12 of 1,150 entries

Figure -3. Load the data

We can see that the dataset has 1150 observations and 2 variables. One of the variables is the tweet itself and the other variable is the average sentiment about the tweet. After loading the dataset, we perform pre-processing the text so we can first create a corpus of tweets so we can see that the corpus has 1150 documents in it. Then we create a function that will clean the corpus to convert all characters to lowercase, remove punctuation, remove any stop words and stem the document. After cleaning the tweets, we can look at the tweet, we can see that the tweets have been cleaned, figure 4 shows the pre-processing the tweets and text.

```
> # View tweets
> tweet_corpus[[1]][1]
$content
[1] "I have to say, Apple has by far the best customer care service I have ever received
! @Apple @AppStore"

> clean_corpus <- function(corp){
+   corp <- tm_map(corp, content_transformer(tolower))
+   corp <- tm_map(corp, removePunctuation)
+   corp <- tm_map(corp, removeWords, c('apple', stopwords('en')))
+   corp <- tm_map(corp, stemDocument)
+ }
>
> # Apply function on corpus
> clean_corp <- clean_corpus(tweet_corpus)
>
> # Check a tweet from cleaned corpus
> clean_corp[[1]][1]
$content
[1] "say far best custom care servic ever receiv appstor"

>
```

Figure-4. Pre-processing the data

After pre-processing, dataset we build the model so we can split the data into train data or test data, and then we can take SVM , NB and CART (Decision tree) classification models to learn on train data and then we can compute the performance of these models. Let's evaluate these models and pick the best model for real time classification. Figure 5 shows the training of these models and Figure 6 shows the evaluation measures of the models.

```
# bayes
set.seed(101)
tweet.bayes <- train(Negative~., data=trainSparse, method='nb',
                     trControl=control, metric=metric)

# CART
set.seed(101)
tweet.cart <- train(Negative~., data=trainSparse, method='rpart',
                    trControl=control, metric=metric)

# SVM
set.seed(101)
tweet.svm <- train(Negative~., data=trainSparse, method='svmRadial',
                   trControl=control, metric=metric)
```

Figure 5. Training of these models

```
Models: Naive_Bayes, CART, SVM
Number of resamples: 10

Accuracy
             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
Naive_Bayes 0.8433735 0.8433735 0.8433735 0.8462783 0.8497323 0.8536585    0
CART        0.8292683 0.8704819 0.8957392 0.8897855 0.9126506 0.9397590    0
SVM         0.8433735 0.8662577 0.8727593 0.8705080 0.8795181 0.8915663    0

Kappa
             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
Naive_Bayes 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000    0
CART        0.2347024 0.3424612 0.4403611 0.4650376 0.5890784 0.7296417    0
SVM         0.0000000 0.2244787 0.2779996 0.2559332 0.3360000 0.5133550    0
```

Figure 6. Performance measure summary of the models

After computing the performance of these model, we can compare the accuracy of these models and based on this comparison we pick the best models for classification. figure 7 shows the comparison of the models.

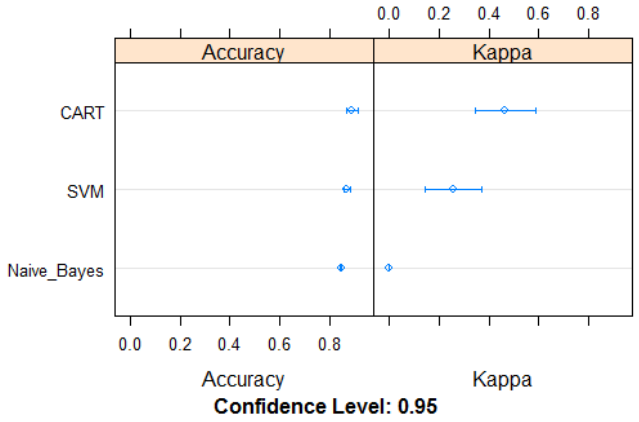


Figure-7 Performance comparison

Looking at the summary and graphs, we can see that CART (Decision Tree) model still has the highest accuracy of 0.8845 on test dataset and figure 8 shows the comparison of these models on the basis of accuracy. And after picking the best model we can real time classification on tweets data.

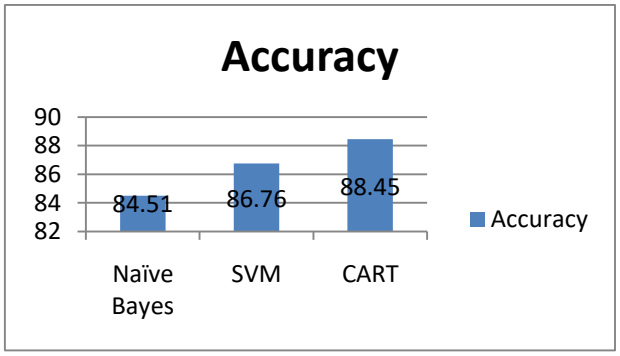


Figure 8. Accuracy comparison

Based on comparison be can say that Decision Tree perform better than SVM and NB, so we can pick the Decision tree for real time classification of twitter data. So, for collect twitter tweets data we need a r package called twitteR and ROAuth package to authenticate user consumer and token keys.

```
> library(twitteR)
> library(ROAuth)
>
>
> consumer_key<-'vE12EJOT2niBwUONTt7Q5sA4u'
> consumer_secret<- 'J5fOrVLmldAafwS4Pxm3eRyB537MMrrsXPdayANnpLyalfpCsY'
> access_token<- '772457331795726337-6IHz8xax4nKXgYfGy7qnZvYmayoOv41'
> access_secret<- 'xRcOOuVc7nRp2Wmq55Z2lzTCzYiDu2daRJNbX1ScOCcHn'
>
> consumer_key<-'vE12EJOT2niBwUONTt7Q5sA4u'
> consumer_secret<- 'J5fOrVLmldAafwS4Pxm3eRyB537MMrrsXPdayANnpLyalfpCsY'
> access_token<- '772457331795726337-6IHz8xax4nKXgYfGy7qnZvYmayoOv41'
> access_secret<- 'xRcOOuVc7nRp2Wmq55Z2lzTCzYiDu2daRJNbX1ScOCcHn'
>
> setup_twitter_oauth(consumer_key, consumer_secret, access_token,access_secret
> setup_twitter_oauth(consumer_key, consumer_secret, access_token,access_secret
[1] "Using direct authentication"
[1] "Using direct authentication"
>
>
> some_tweets = searchTwitter("oneplus6", n=1500, lang="en")
> some_tweets = searchTwitter("oneplus6", n=1500, lang="en")
```

For consumer key and access tokens we need to create a twitter app through which we can generate our twitter keys and put into twitter_oauth and than we are storing to collects tweets on modi with frame size of 1000 and stored into some_tweets variable. So, we can get the raw data so before classification we need to pre-process the data so with the help of tm and NLP package's we can remove numbers, punctual, and special characters coming from text. And first we are converting all the text into lower case to remove noise and we can also eliminate URLs coming from text.

After pre-processing the data, we are applying tree classifier to classify the text into various sentiment emotions and sentiment polarities.

```
#Sentiment Analysis

# using sentiment package to classify emotions
emotions <- classify_emotion(clean_tweets, algorithm='tree')

# using sentiment package to classify polarities
polarities = classify_polarity(clean_tweets, algorithm='tree')
```

The Decision Tree algorithm classify the text into various emotions shown in figure 9.

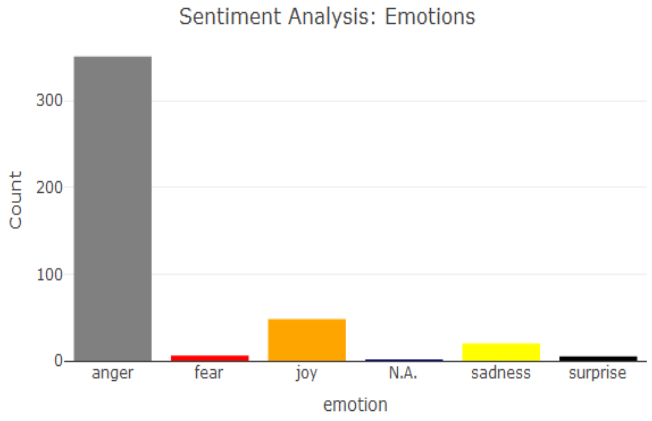


Figure 9. Classification by emotions

We can also classify the sentiment polarity of tweets text and which is shown in figure 10.

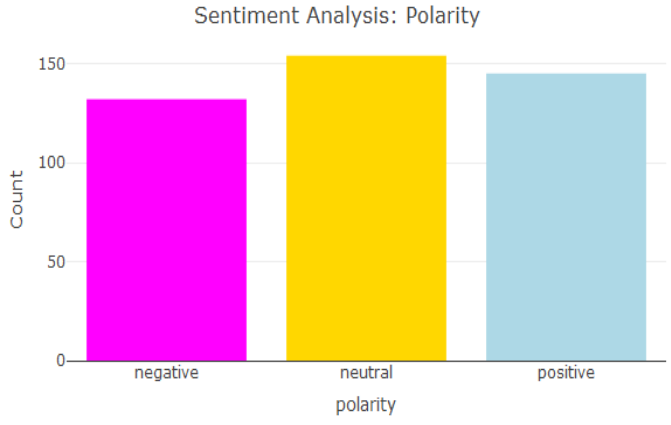


Figure 10. Classification by Polarity

VI CONCLUSION

Text mining generally refers to the process of extracting valuable information from unstructured text. Hidden information in social network sites, bioinformatics and internet security etc. are identified using text mining is a

major challenge in these fields. This paper we investigate various classification algorithms such as Support Vector Machine (SVM), Naive bayes and CART (Decision Tree) for classify the user opinions into category positive or negative. In these we can build a classifier based on this classification algorithm compute the performance measure of these classifiers. And also compare performance of these classification and we found that Decision Tree performs better than NB and SVM classification algorithm so for real time classification we can pick the Decision tree classifier. In these we can classify real time twitter data with the help Decision Tree classifier into various emotions and polarity.

REFERENCES

[01] Chandrasekhar Rangu, Shuvojit Chatterjee, Srinivasa Rao Valluru, "Text Mining Approach for Product Quality Enhancement" in IEEE 2017.

[02] Shruti Kohli, Himani Singal, "Data Analysis with R" in 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing.

[03] Arun Jalanila, Nirmal Subramanian, "Comparing SAS® Text Miner, Python, R" in 2016 IEEE International Conference on Healthcare Informatics.

[04] Upma Kumari, Dr. Arvind k. Sharma, Dinesh Soni, "Sentiment analysis of smart phone product review using SVM classification technique " in IEEE 2017.

[05] T. K. Das , D.P. Acharjya & M. R. Patra, " Opinion Mining about a product by Analyzing Public Tweets in Twitter ", ICCCI- 2014, Jan 03-05, 2014.

[06] Porter M.F, Snowball: A language for stemming algorithms. 2001.

[07] Ning Zhong, Yuefeng Li, and Sheng-Tang Wu, "Effective Pattern Discovery for Text Mining", *IEEE Transactions on Knowledge And Data Engineering*, Vol. 24, No.1, January 2012.

[08] SM Abu Taher, Kazi Afsana Akhter, "N-gram Based Sentiment Mining for Bangla Text Using Support Vector Machine" in IEEE 2018.

[09] Yuling Chen, Zhi Zhang, "Research on text sentiment analysis based on CNNs and SVM" in IEEE 2018.

[10] Imane El Alaoui1,2* , Youssef Gahi3, Rochdi Messoussi1, "A novel adaptable approach for sentiment analysis on big social data" in Springer 2018.

[11] Zhaoxia WANG, Victor Joo Chuan TONG, "Issues of social data analytics with a new method for sentiment analysis of social media data" in IEEE 2014.