N-Gram Based Sentiment Mining for Bangla Text Using Support Vector Machine

Conference Paper · September 2018

DOI: 10.1109/ICBSLP.2018.8554716

3 authors:

Sm Abu Taher
Khulna University of Engineering and Technology
2 PUBLICATIONS **6** CITATIONS
SEE PROFILE

Kazi Afsana Akhter
Khulna University of Engineering and Technology
2 PUBLICATIONS **6** CITATIONS
SEE PROFILE

K. M. Azharul Hasan
Khulna University of Engineering and Technology
69 PUBLICATIONS **317** CITATIONS
SEE PROFILE

N-gram Based Sentiment Mining for Bangla Text Using Support Vector Machine

SM Abu Taher
Department of Computer Science and Engineering
Khulna University of Engineering &Technology
Khulna, Bangladesh
shayokh71@gmail.com

Kazi Afsana Akhter
Department of Computer Science and Engineering
Khulna University of Engineering &Technology
Khulna, Bangladesh
kaziafsana373@gmail.com

K.M. Azharul Hasan
Department of Computer Science and Engineering
Khulna University of Engineering &Technology
Khulna, Bangladesh
az@cse.kuet.ac.bd

***Abstract*— Opinion Mining is a valuable knowledge resource to understand the collective opinions and to take better decisions. It is a Natural Language Processing (NLP) task that decides whether a text expresses positive or negative sentiment. Web contents are increasing rapidly and providing a huge number of information. It is an important research issue to analyze and organize these enormous information for better knowledge extraction. In this paper, we emphasis on opinion mining for Bangla text using web based diverse data. We apply both Linear and Nonlinear Support Vector Machine as machine learning technique and N-gram method to classify Bangla documents collected from social media sites. Most of works in this arena take a single word as a vector. Instead of thinking a single word as a vector, we used one vector containing more than one words using N-gram. N-grams of texts are extensively used in text mining and natural language processing tasks. We found better results using N-grams for different values of n.**

Keywords—Opinion Mining, N-gram, SVM, Bangla text classification, Sentiment classification, POS.

I. INTRODUCTION

Cause of the social networking, people will share their view via social media sites as Facebook and other Social sites. To analyze the huge amount of information intelligent system is used that classifies these vast data into a predefined classes as positive opinion and negative opinion which is referred to as opinion mining[1]. Opinion Mining is a valuable knowledge resource to understand collective sentiments and helps make better informed decisions[12]. In the last decade, opinion mining has attracted an increasing interest. It is a hard challenge for language technologies, and achieving good results. The task of automatically classifying a text written in a natural language into a positive or negative feeling, opinion or subjectivity is sometimes so complicated that even different human annotators disagree on the classification to be assigned to a given text[3]. Personal interpretation by an individual is different from others, and this is also affected by cultural factors and each person's experience. And the shorter the text, the worse written, and the more difficult the task becomes, as in the case of messages on social networks like Twitter or Facebook. In recent days, social media has become a comfortable place for the people to express their opinion on various topics. We can easily get idea about what people are thinking. Around 226 million people speaks Bangla and a large portion of them are active on internet[14]. So it is important to extract opinion hidden in Bangla text for sentiment detection.

We apply N-gram for Bangla text. N-grams are simply all combinations of adjacent words or letters of length *n* that can be found in a source text. One important facility that provides N-gram is that it captures the language structure from the statistical point of view, like what letter or word is likely to follow for the given sentence. If the value of n is high then more context you have to work with. Optimum length of n depends on the application and context of use. If the N-grams are too short it may fail to capture important difference and if it is too long it captures the general knowledge only.

In this paper we use Support Vector Machine (SVM)[5] for classification and N-gram method for vectorization. SVM is widely used for text classification but generally all the individual words are the target vectors there. In this paper we change the vectors by applying the N-gram technique. Hence large N-grams the vectors becomes less. This definitely improves the performance. N-gram performs well when two or more sentences express a single opinion specially negative sentences. We generate Bangla dataset from different authentic sources to train the SVM. The technique can easily be applied to sentiment classification[4][15], emotion detection[19][20] and other NLP processing purposes such as summarization[16].

II. RELATED WORKS

Support Vector Machine is a very effective tool for text classification and is widely used for opinion mining[9][16][18]. Many researchers have proposed methods to overcome some problems based on feature selection or algorithm improvements [2]. Features extracted for SVM applying N-gram method works effectively [3]. Some supervised learning methods to classify Bangla text proposed in this decade. For opinion mining in Bangla, training and test set of Bangla language are generated using google translator[4] and used naive biased method to classify [15]. Introduced the contextual valency to detect sentiment using sentiwordnet[17]. A machine learning algorithm is employed in [12] to detect sentiment from Bangla text using negative or positive pattern of the sentence. The superiority of SVM classification algorithm shows in [10] in topic based categorization. A hybrid system is proposed to classify

978-1-5386-8207-4/18/$31.00 ©2018 IEEE

overall opinion polarity from Bangla text that works with linguistic syntactic features [11]. A hybrid approach of Support Vector Machine and Particle Swarm Optimization is used to mine opinion of movie reviews [18]. In this paper we have formulated the target vectors by applying N-gram (N>=2) technique. In fact all the techniques that used SVM used unigram (N=1). We apply N-gram for different values of N. Hence improved performance has been received specially when more than one sentence possess the sentiment.

III. N-GRAM AND SVM

An N-gram model models sequences, notably natural languages, using the statistical properties of N-grams. N-grams gives the idea of word prediction with probabilistic models. N-grams of texts are sensitive to text mining and hence are extensively used in text mining and natural language processing tasks. They are basically a set of co-occurring words within a given window and when computing the N-grams we typically move one word forward. For example, for the sentence *"The cow jumps over the moon"*. If N=2, the 2-grams are {<the cow>, <cow jumps>, <jumps over>, <over the>, <the moon>}. So we have 5 2-grams in this case. When n=1, this is referred to as unigrams and this is essentially the individual words in a sentence. When n=2, this is called bigrams and when n=3 this is called trigrams. When n>3 this is usually referred to as four grams or five grams and so on. N-gram models are quite robust for information retrieval without much dependency on language. Google and Microsoft have developed web scale N-gram models that can be used in a variety of tasks such as spelling correction, word breaking and text summarization. In this paper, we use N-gram method for developing features for opinion mining as a supervised Machine Learning model using SVM.

SVM is an algorithm that uses a nonlinear mapping to transform the original training data into a higher dimension. In this algorithm, each data item is plotted as a point in n-dimensional space (where n is number of features) with the value of each feature being the value of a particular coordinate. Then, classification is performed by finding the hyper-plane that differentiate the two classes very well so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. A hyperplane H in n dimensional space is the set of points (x_1 , x_2 , ……. , x_n) that satisfy a linear equation

$a_1x_1 + a_2x_2 + \text{.....} + a_nx_n = \text{b}$ (i)

For SVM, there can be infinite number of separating hyper planes. The best one has to be found which will have the minimum classification error on previously unseen tuples.

The optimal hyper plane will be the one with the biggest margin. That is why the objective of the SVM is to find the optimal separating hyper plane which maximizes the margin of the training data. The hyperplane with larger margin shown in Figure 1 is better than the one with smaller margin. This optimal separating hyperplane is called Maximum Marginal Hyperplane (MMH).

The equation of hyperplane is

$\text{W.X} + \text{b} = 0$ (ii)

Here,

W = weight vector perpendicular to the plane

X = training tuple

b = bayes

Let, X_n be the nearest data point to the hyperpalane

Unit vector of W is $\widehat{W} = \frac{W}{||W||}$

Take any point y on the plane.

We have to find the distance between X_n and the plane W.X + b = 0 where

$|\text{W}.X_n + \text{b}| = 1$. The required distance is the projection of $X_n - \text{y}$ on W (can be positive or negative depending on the position of X_n).

distance, $\text{d} = |\widehat{W}(X_n - \text{y})|$

$$= \frac{1}{||W||}|W.X_n - \text{W.y}|$$
$$= \frac{1}{||W||}|W.X_n - \text{W.y} + \text{b} - \text{b}|$$
$$= \frac{1}{||W||}|W.X_n + \text{b} - (\text{W.y} + \text{b})|$$
$$= \frac{1}{||W||}|1 - 0|$$
$$= \frac{1}{||W||}$$

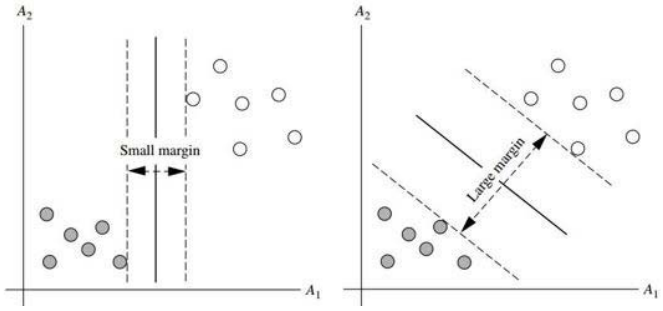

Figure 1: Maximum Marginal Hyperplane

Now, we have to maximize $\frac{1}{||W||}$ to find the MMH.

Maximizing $\frac{1}{||W||}$ is same as minimizing $\frac{1}{2}W.W$, subject to $y_n(W.X_n + \text{b}) \geq 1$

Here, y_n is the class label of X_n n = 1,2,3 … N [N is the number of training data]

Using any Support Vector,

$y_n(\text{W}X_n + b) = 1$ (iii)

From W and b, we get the desired optimal hyperplane.

From the sign of the equation, it is decided whether the class is positive or negative.

IV. DATA GENERATION

To classify the opinion, we have collected around 9,500 comments from different sources as shown in Table 1. Table 1 also shows the topics of comments along with their sources and the date of collection. We have used Facebook Graph API [13] for collecting comments from verified Facebook pages as JSON file. Then we have generated dataset by labeling the comments as positive or negative appropriately. Table 2 shows an example of this file.

Table 1: Data, Source and Date

Sl. No.	Topic	Source	Date
1.	Bangladesh Test match win against England	BBC Bangla Prothom Alo	8/3/2017
2.	Sakib Al Hasan suspended by BCB	BBC Bangla, Prothom Alo	23/11/2016
3.	VAT applied on Private Universities	BBC Bangla Prothom Alo	06/01/2017
4.	US Election	BBC Bangla, Prothom Alo	15/12/2016
5.	Rohingya Crisis	BBC Bangla, Prothom Alo	12/1/2017
6.	Biometric Registration	BBC Bangla, Prothom Alo	23/01/2017
7.	Excessive Advertisement in TV channel	BBC Bangla	5/3/2017

Table 2: Template of the dataset

pos	যে ভালো তাকে ভালোই বলতে হয়
neg	আপনারা তো মানুষের ভাল কাজের কদর করতেও জানেন না

V. DATA PREPROCESSING

To make the dataset usable, we perform a number of preprocessing tasks and they are as follows.

A. *Separating Words According to POS*

We have a main vocabulary of Bangla words where around 1,10,000 words of Bangla language are present. But all words are not equally important to express sentiment. We consider all *adjectives* and *adverbs* as important words. Some nouns and verbs are also necessary. So, we separate the words according to Parts of Speech (POS) to extract *adjectives*, *adverbs*, *nouns* and *verbs*. From *nouns* and *verbs*, we select those words that shows sentiment. We convert all important verbs to their base forms. To reduce the search space, we partition the words according to their POS and stored in a XML file. Figure 2 shows the number of words in each partition.

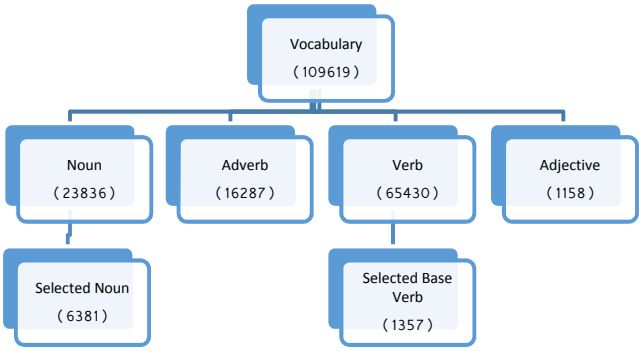

Figure 2: No. of Words in Each File

B. *Stemming*

Bangla verbs have a large number of varieties. To reduce number of vector, we convert all forms of a verb to its base form by eliminating suffix. By performing this task we can keep the meaning unchanged and get high accuracy. Figure 3 shows an example.

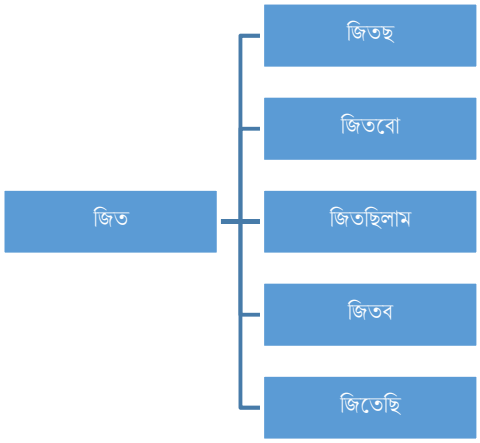

Figure 3: Example of stemming

C. *Recognize Emoticons*

Emoticons are recognized in this step. Symbols for emoticons are found in comments and replaced with appropriate words which plays an important role for expressing opinion.

Table 3: Emoticons recognition

Input	After Processing
মন্ত্রিরা এমনি হয় :/	মন্ত্রিরা এমনি হয় বিরক্ত
এই হল মাটির মানুষ ☺	এই হল মাটির মানুষ খুশি

D. *Removing Unnecessary Punctuation*

All special symbols are removed except interrogation mark (?) and exclamation mark (!). These two marks are important for expressing sentiment and are not removed. For example "বুড়া ব্যাটা বলে কি !'' shows a negative sentiment.

E. *Separating Negativity*

A white space is inserted before 'না', 'নি', 'নাই' where they are concatenated with other words as negation. Sometimes people concatenate these with other words and the resulting string cannot be found in the vocabulary. But if these are as a part of a single word, they are not separated. Table 4 shows example of negativity separation.

Table 4: Negation separation

No.	Input	After Processing
(a)	কয়লা ধুলে ময়লা যায়না	কয়লা ধুলে ময়লা যায় না
(b)	কাজটা ভাল হয়নি	কাজটা ভাল হয় নি
(c)	আজ জয়ের সম্ভাবনা বেশি	আজ জয়ের সম্ভাবনা বেশি

'নি', 'নাই', 'নেই' these three are converted to 'না', 'নি', 'নাই', 'নেই' and 'না' all four words express negative sentiment. So this conversion reduces the number of vector dimension keeping the meaning unchanged.

F. Removing Unnecessary Words

Each word of dataset is checked. If it is found directly in adverb, adjective or selected noun (see Figure 2), then the word is a candidate. Otherwise the word is checked whether it is in the all forms of verb. If the word is present there, then it is converted to its base form and becomes candidate. Algorithm 1 shows the details. This task also helps to reduce vector dimension.

Algorithm 1:

i. *repeat until each word of text is checked*
(a) *if word is in verb convert it to base form*
candidate=true;
(b) *if word is in adverb, adjective or selected nouns*
candidate=true;
else
if word is in vocabulary
discard it
else
convert it to base form
and go to ***i(b)***
ii. *stop*

G. Applying range of N-grams

Since the combination of words possess different sense in Bangla, we produced different N-grams and combine them to increase the vector dimension. We call it range of N-grams. Table 5 shows some range of N-grams with input texts. Some range of N-grams and their corresponding vector dimensions produced by our data set are as follows:
For [1]-gram, vector dimension = 4,479
For [1-2]-gram, vector dimension = 55,037
For [1-3]-gram, vector dimension = 1,31,312

Table 5: Range of N-grams

N-gram	input	output
1 gram	হার না খুশি	‘হার’, ‘না’, ‘খুশি’
1-2 gram	হার না খুশি	‘হার’, ‘হার না’, ‘না’, ‘না খুশি’, ‘খুশি’
1- 3 gram	হার না খুশি	‘হার’, ‘হার না’, ‘হার না খুশি’ ‘না’, ‘না খুশি’, ‘খুশি’
2 gram	হার না খুশি	‘হার না’, ‘না খুশি’
2-3 gram	হার না খুশি	‘হার না’, ‘হার না খুশি’, ‘না খুশি’

H. An Illustrative Example

This implementation can be discussed with an example.

Let some texts,

a = আনন্দের সংবাদ ;b = সে খুবই সুন্দর গান করে ;c = মাসরাফির তুলনা হয় না ;
d = অত্যন্ত বিরক্তিকর কাজ ; e = মুভিটা ভালোনা
After preprocessing these sentences we get,
p_a =আনন্দ
p_b = খুব সুন্দর কর
p_c = তুলনা না
p_d = অত্যন্ত বিরক্তিকর কাজ
p_e = ভালো না

Vocabulary generated from these sentences using 1-gram are as follows
['অত্যন্ত', 'আনন্দ', 'কর', 'কাজ', 'খুব', 'তুলনা', 'না', 'বিরক্তিকর', 'ভালো', 'সুন্দর']
Now we can generate vector representation (VR) for these sentences using the above vocabulary. Vector representations are given below:
VR of p_a = [0 1 0 0 0 0 0 0 0 0]
VR of p_b = [0 0 1 0 1 0 0 0 0 1]
VR of p_c = [0 0 0 0 0 1 1 0 0 0]
VR of p_d = [1 0 0 1 0 0 0 1 0 0]
VR of p_e = [0 0 0 0 0 0 1 0 1 0]
From iii, we know,

$$y_n(\mathrm{W}X_n + b) = 1$$

Here,
y_n = the class label.
X_n = training tuple.
W = weight vector perpendicular to the plane

b = bayes
y_a =1 ;y_b =1 ;y_c =1 ;y_d = -1 ; y_e = -1;
X_a= [0 1 0 0 1 0 0 0 0 0]
X_b= [0 0 1 0 1 0 0 0 0 1]
X_c= [0 0 0 0 0 1 1 0 0 0]
X_d= [1 0 0 1 0 0 0 1 0 0]
X_e= [0 0 0 0 0 0 1 0 1 0]
W and b can be calculated. Then the tuples are plotted. Figure 4 shows the vectors and MMH that separate the two class.

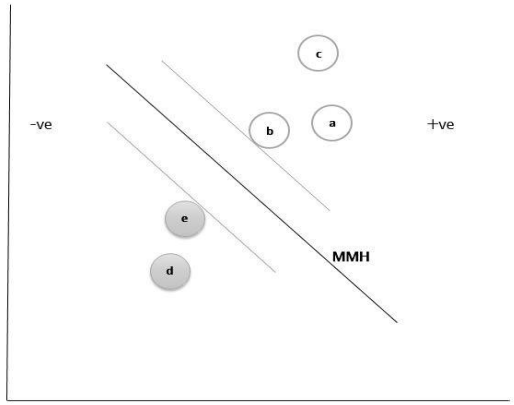

Figure 4: Plotting of Vectors

Here a, b, c is in the positive class and d, e is in the negative class and e, b are the support vectors.
Here, in 1-gram we get 'ভালো' and ‘না’ as separate feature. But it is not appropriate for the given sentence.
For same preprocessed sentences, 2-grams are given below: ['অত্যন্ত বিরক্তিকর', 'তুলনা না', 'বিরক্তিকর কাজ', 'ভালো না', 'সুন্দর কর']
Here we see 'ভালো না' is a single feature, which express the exact meaning of the sentence e. But for sentence a, we get: X_a =[0 0 0 0 0]. This is clearly an undefined condition. It can’t express anything though the sentence is strongly positive. On the other hand if we take the range [1-2]-grams, the problem can be solved. For same preprocessed sentences, [1, 2]-grams are given below:

['অত্যন্ত', 'অত্যন্ত বিরক্তিকর', 'আনন্দ', 'কর', 'কাজ', 'তুলনা', 'তুলনা না', 'না', 'বিরক্তিকর', 'বিরক্তিকর কাজ', 'ভালো', 'ভালো না', 'সুন্দর', 'সুন্দর কর']

Here 'ভালো না' is a single feature which express the meaning perfectly as well as we get X_a =[0 0 1 0 0 0 0 0 0 0 0 0 0 0] which is the appropriate vector representation of sentence a. So, we get good accuracy by using range of N-grams.

VI. EXPERIMENTAL RESULTS

A set of comments is classified into two polarities: positive and negative. Table 6 shows the experimental results after applying Linear SVM where the value of regularization parameter, C = 1.0 . For Linear SVM we observed soft margin performed better than hard margin as we got best accuracy from range 0.6 to 2.8 for the value of C.

Table 6: Experimental Results for Linear SVM

Range of gram	[1,1]	[1,2]	[1,3]	[2,2]	[2,3]
Vector Dimension	4479	55037	131312	50558	126838
Training Data (No. of Comments)	7548	7548	7548	7548	7548
Test Data (No. of Comments)	1948	1948	1948	1948	1948
True (+)ve	566	585	581	596	606
True (-)ve	1146	1199	1205	1023	991
False (+)ve	148	95	89	271	303
False (-)ve	88	69	73	58	48
Accuracy (%)	87.885	91.581	91.684	83.111	81.982

A graphical representation of accuracy level for each considering gram combination is shown in Figure 5.

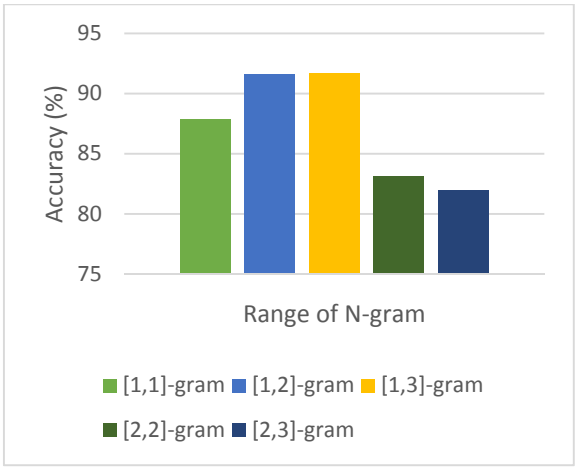

Figure 5: Results from Linear SVM

We have also applied Nonlinear SVM using Radial Basis Function (RBF) and Polynomial kernel. Among these two RBF performed better. For text classification, Linear SVM provides more accuracy than Nonlinear SVM [19]. Our experimental results also say that Linear SVM is better than Nonlinear SVM for text classification as number of feature is much more greater than number of observation and we noticed that accuracy deviates more with the increasing number of features. Results produced by using Nonlinear SVM(RBF) are shown in Table 7.

Table 7: Experimental Results from Nonlinear SVM

Range of gram	[1,1]	[1,2]	[1,3]	[2,2]	[2,3]
Vector Dimension	4479	55037	131312	50558	126838
Training Data (No. of Comments)	7548	7548	7548	7548	7548
Test Data (No. of Comments)	1948	1948	1948	1948	1948
True (+)ve	516	531	562	599	610
True (-)ve	1157	1205	1177	933	828
False (+)ve	137	89	117	361	466
False (-)ve	138	123	92	55	44
Accuracy (%)	85.883	89.117	89.271	78.645	73.819

A graphical representation of this result is shown in Figure 6.

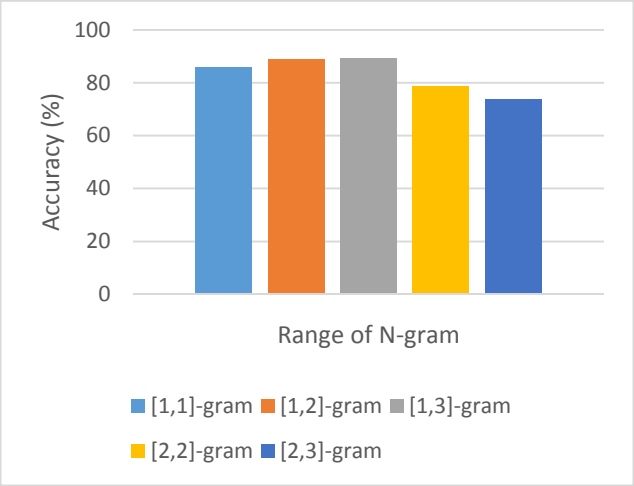

Figure 6: Results from Nonlinear SVM

From the results, it is clear that Linear SVM is more efficient than Nonlinear SVM for text classification.

VII. CONCLUSION

We describe a system that can classify opinion of different sentiment from Bangla text. To accomplish our goal, we collected data from various web based authentic sources. We processed data in a unique way that only essential words to express sentiment is presented hence high accuracy is received. We applied N-gram method for vectorization and Support Vector Machine for classification. We got satisfactory accuracy from this system. We checked notable number of random comments with our classifier and got good outcomes.

REFERENCES

[1] Vikash Singh Rajput, Shirish Mohan Dubey, "Stock Market Sentiment Analysis Based On Machine Learning", 2016 2nd International Conference on Next Generation Computing Technologies (NGCT-2016) Dehradun, India 14-16 October 2016.

[2] Yeonghun Chae, Do-Heon Jeong, Taehong Kim, "Experimental Study of Time Series-based Dataset Selection for Effective Text Classification", 2017 9th International Conference on Knowledge and Smart Technology (KST) .

[3] S. Foroozan, M.A. Azmi Murad, N.M. Sharef, A.R. Abdul Latiff, "Improving Sentiment Classification Accuracy of Financial News using N-gram Approach and Feature Weighting Methods", IEEE Conference Publications, 2015.

[4] K. M. Azharul Hasan, Mir Shahriar Sabuj, Zakia Afrin" ,Opinion Mining using Naïve Bayes", WIE Conference on International IEEE 20-19 ,(ECE – WIECON) and Computer Engineering Electrical December, 2015.

[5] Wikipedia contributors. (2018, August 25). Support vector machine. In Wikipedia, The Free Encyclopedia. Retrieved 12:33, August 2, 2018, from https://en.wikipedia.org/w/index.php?title=Support_vector_machine&oldid=856498176

[6] Boiy, E. ,Hens ,P.,Deschacht, K. &Moens ,M-.F., "Automatic Sentiment Analysis in OnLine Text". In Proceeding of the Conference on Electronic Publishing (ELPUB-2007) ,pp.349-360,2007.

[7] Liu, B. ,"Sentiment Analysis and Subjectivity" .To appear in Handbook of Natural Language Processing ,Indurkhya ,N. & Damerau, F. J. Ed, 2010.

[8] Pang, Bo and Lillian Lee, "Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval", Vol. 2(1-2) , pp. -1 135,2008 .

[9] Rudy Prabowo, Mike Thelwall "Sentiment Analysis: A combined Approach", Journal of Informatics, pp. 143-157, 2009.

[10] P. Turne., "Thumbs Up or Thumbs Down ? Semantic Orientation ."Applied to Unsupervised Classification of ReviewsProceeding of the 40th Annual Meeting of the Association for Computational Linguistics (ACL) ,Philadelphia ,pp.417-424 ,2002.

[11] Das, A. and Bandyopadhyay, S., February, 23th to 24th, "SentiWordNet for Bangla." In Knowledge Sharing Event-4: Task 2: Building Electronic Dictionary Mysore, 2010.

[12] KM Azharul Hasan, Md Sajidul Islam, GM Mashrur-E-Elahi, Mohammad Navid Izhar, "Sentiment recognition from bangla text" Technical Challenges and Design Issues in Bangla Language Processing, IGI Global, 2013.

[13] https://developers.facebook.com/docs/graph-api/

[14] K. M. Azharul Hasan, Amit Mondal, Amit Saha, "Recognizing bangla grammar using predictive parser", arXiv preprint arXiv:1201.2010.

[15] K. M. Azharul Hasan, M Rahman, "Sentiment Detection from Bangla Text using Contextual Valency Analysis" Proc. Of ICCIT 2014.

[16] Duy Duc AnBui , GuilhermeDel FiolMD, John F.Hurdle, Extractive text summarization system to aid data extraction from full text in systematic review development, Journal of Biomedical Informatics, 64, 265-272, 2016. /doi.org/10.1016/j.jbi.2016.10.014.

[17] Esuli, A. and F. Sebastiani: 2006, ' SentiWordNet : A Publicly Available Lexical Resource for Opinion Mining'. In: Proceedings of the 5th Conference on Language Resources and Evaluation (LREC 2006) . Genova, IT, pp. 417–422.

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

[19] Myriam Munezero, Calkin Suero Montero, Erkki Sutinen, and John Pajunen, "Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text", IEEE transactions on affective computing, vol. 5, no. 2, pp-101-112, 2014.

[20] Haji Binali, Chen Wu,Vidyasagar Potdar, Computational Approaches for Emotion Detection in Text, 4th IEEE International Conference on Digital Ecosystems and Technologies (IEEE DEST), pp. 172-177, 2010.