Text Polarity Detection using Multiple Supervised Machine Learning Algorithms

Sagnik Kar, Mousumi Saha, Tamasree Biswas

Abstract*: *Sentiment analysis is the classifying of a review, opinion or a statement into categories, which brings clarity about specific sentiments of customers or the concerned group to businesses and developers. These categorized data are very critical to the development of businesses and understanding the public opinion. The need for accurate opinion and large-scale sentiment analysis on social media platforms is growing day by day. In this paper, a number of machine learning algorithms are trained and applied on twitter datasets and their respective accuracies are determined separately on different polarities of data, thereby giving a glimpse to which algorithm works best and which works worst..

Keywords : accuracy analysis, accuracy comparison, natural language processing, semantic orientation, sentiment analysis, twitter data analysis.

I. INTRODUCTION

Social Media platforms which include Facebook, Twitter, Instagram, etc. are increasingly used to give opinions, thoughts and reviews of products and matters relating to public concern. The analysis of these data and classifying them into categories, would give us a clear picture of the public opinion. These categorized data are of utmost importance to businesses and developers who want a glimpse of the public opinion to improve their products in order to grow their businesses and other stakeholders who needs the public opinion beforehand to improve their performance and decision making capabilities [1]. In studies, it has been shown that there is a major influence of consumer reviews on online purchasing decisions in both older and younger adults [2]. So, it is necessary for companies to take into consideration reviews of their products and opinions made of their products on online platforms. Sentiment analysis generally involves classifying opinions into negative or positive by carefully analyzing the words used in that particular opinion or view. The frequency of a particular word used, the adjective used all are taken into consideration while determining the polarity of that specific piece of content. However, this determination of polarity is impossible to do manually on a large number of opinions generated everyday on a number of social media platforms and websites. So, there is a need to automate the process which not only returns the polarity result on a huge chunk of data in seconds, but also can accurately tell about the polarity, as accuracy is of great importance here. But, accuracy in sentiment analysis is a tricky thing to deal with, as natural language is highly complex, unstructured and in social media platforms it can also include Emoji, which are not detected by text analyzers. Some machine learning algorithms can work best under these circumstances while analyzing the sentiment, so these specific algorithms need to be deployed to accurately determine the particular sentiments on a large group of data. In this paper, we particular focused on negative and positive polarities which can be further classified into any number of emotions. The machine learning approaches such as Naïve Bayes, Support Vector Machine, Decision Tree, Logistic Regression, Random Forest and Decision Tree are used to train a dataset and then these algorithms are tested on another set of data to check of their accuracies on different polarities. The accuracy so obtained can clearly tell us which approach works best and which works worst.

II. LITERATURE SURVEY

Various researches have been conducted and are been conducted presently to accurately predict nature of online reviews using both machine learning techniques and other methods. In some cases, list of words is stored in dictionaries, with their corresponding polarity. Then, the polarity of each word in the document is taken into account to finally calculate the resultant polarity of the whole document. [3]. There are mainly two main methods to obtain the sentiment of a word or a sentence automatically, namely, the lexicon based approach which include figuring out the associated orientation of a word or a piece of text from predefined dictionaries [4], while the machine learning approach involves constructing classifiers from labeled chunks of words or sentences [5] essentially a supervised classification task, however can be improved to unsupervised also. The latter approach could also be described as a statistical or machine-learning approach. However, in the lexicon based methods predefined words are associated with various types of sentiments, so labelling new data can be costly or cannot be allowed in some cases. In the 2016 US election, manually annotated corpus-based hash tags along with negation deletion were used a 7% increase in accuracy level was observed. [6] A different approach is proposed for determining election results before the actual election took place. The proposal includes taking into consideration user influence factor, along with applying SVM algorithm. The model is said to have achieved 88% accuracy in predicting the 2013 Karnataka Assembly elections of India and the US 2012 presidential elections. [7] Another research conducted before the 2014 General Elections of India involved collecting a dataset of tweets containing hashtags which are relating to the candidates and the election of 2014. [8] They intentionally rejected the neutral tweets for prediction, as they made the analysis more

Revised Manuscript Received on January 05, 2020

***** Correspondence Author

Sagnik Kar*, Information Technology, Narula Institute of Technology, Kolkata, India.

Mousumi Saha, Computer Science & Engineering, Narula Institute of Technology, Kolkata, India.

Tamasree Biswas, Information Technology, Narula Institute of Technology, Kolkata, India.

complex. They used bipolar lexicons for the two parties, but in case of involvement of numerous political parties this approach would not be totally adequate.

Our work doesn't propose any new prediction model, but aims to carefully recognise the accuracies of different machine learning algorithms. The approach will determine the positive and negative polarities with the help of Logistic Regression, Random Forest, Naïve Bayes, Support Vector Machine and Decision Tree.

III. MACHINE LEARNING ALGORITHMS

There are mainly three types of machine learning algorithms viz. supervised, unsupervised and semi-supervised. In this paper we had essentially used all supervised algorithms. We had streamed a particular number of tweets about a certain product, tag or person, which could be changed manually and pre-processed the polarity of those tweets using the TextBlob text analyzer. The streamed tweets along with the pre-processed polarity served as a training dataset for different algorithms. The results obtained are then tested to a different variation of dataset to check of their accuracy. Supervised learning algorithms such as Naïve Bayes, Logistic Regression, Decision tree, SVM (Support Vector Machine) and Random Forest are used to analyze for their respective accuracies on different polarities, in three different keywords. Some supervised algorithms like Random Forest and Logistic Regression models are being used first in text sentiment analysis.

A. Multinomial Naïve Bayes classifier

The Naïve Bayes classifier is a basic and well-known classifier extensively used in Natural Language Processing(NLP). It has the ability to achieve above-average performance in areas like sentiment analysis.

Consider we want to classify a given document into positive or negative. These are the two classes to which the document can belong.

The equation can be represented as below-

$$\hat{c} = \text{argmax } P(c \mid d), \text{ where } c \in C \qquad (1)$$

Here, C is the set of all possible classes that is, in this case, positive or negative. c is one of those classes either positive or negative whose probability is being checked and d is the document which we are classifying. We can write the same equation with the help of the Bayes rule and can drop the denominator as it is not dependent on class c.

$$\text{Now, } \hat{c} = \text{argmax } P(d \mid c)P(c) \text{ , where } c \in C \qquad (2)$$

Here, P (d | c) is the likelihood and P(c) is the prior. And, P(c) is the probability of having a document from class c, and P (d | c) is the probability that given it is class c document d belongs to it. The Naïve Bayes assumption is that, given a class c the presence of a particular feature of a document is independent of others. So, we can write-

$$c_{NB} = \text{argmax } P(c) \prod_{f \in F} P(f \mid c) \qquad (3)$$

Here, every word in the document is considered as a feature and is considered to be independent.

Now, we will use the logarithm of the function for formulating the classifier.

$$c_{NB} = \text{argmax } \log P(c) + \sum_{i \in positions} \log P(w_i \mid c) \quad (4)$$

Here, P(c) denotes the probability of a document of class c and P $(w_i \mid c)$ is the probability that a word w_i belongs to a document of class c. The term multinomial signifies that we have a multiple feature and we will count the relative frequency of those features.

In the Naïve Bayes approach, we found that for positive polarity, it has successfully predicted the correct polarity for about 118 tweets and there is an error in polarity detection for about 105 tweets for the keyword "#apple". Likewise for negative polarity, it has predicted accurate polarity for 4 tweets whereas for 13 tweets it made an erroneous prediction. So, the net accuracy of this algorithm is about 54.58%, 67.94% and 68.11% in the three respective test cases.

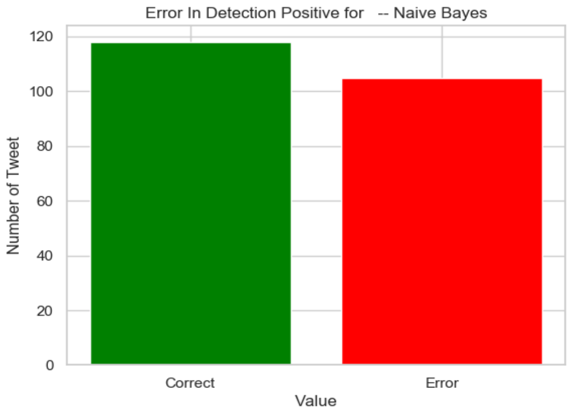

Fig 1: The experimental result of accuracy of Naïve Bayes classification, when the polarity is positive.

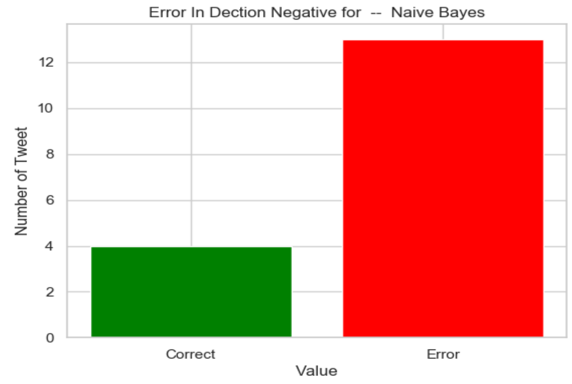

Fig 2: The experimental result of accuracy of Naïve Bayes classification, when the polarity is negative.

B. Logistic Regression classifier

Logistic regression is a machine learning algorithm which is generally used in classification problems, and it is based on the concept of probability.

Logistic regression model uses the cost function known as the sigmoid function, whose value lies between 0 and 1, unlike in case of linear regression which can be greater than 1 or less than 0.

$$0 \leq h_{\Theta}(z) \leq 1 \qquad (5)$$

The sigmoid function helps to map real values into a range between 0 and 1, which helps us to map predictions to probability.

The sigmoid function,

$$h_{\Theta}(z) = 1/ (1 + e^\wedge- z) \qquad (6)$$

$$\text{Now, } z= \beta_0 + \beta_1 X$$

$$\text{i.e. } h_{\Theta}(z) = 1/ (1 + e^\wedge- (\beta_0 + \beta_1 X) \qquad (7)$$

In logistic regression, suppose (1, x_1, x_2…. x_m) are the different features of a given sample, which can be different words in a given document. These are known as the input vector The vector of weights W are the group of weights (w_0, w_1, w_2…w_m) which the model will adjust to predict more accurately. The dot product W.V is known as the Net Input function. Then, the sigmoid function transforms this real value into a value between 0 and 1. Small inputs are transformed into a value close to 0 and big inputs are transformed into a value close to 1.

Using the Logistic Regression approach, we found that for positive polarity, it has successfully predicted the correct polarity for about 223 tweets and there is no error in positive polarity detection. Similarly, for negative polarity, it has predicted correct polarity for 17 tweets with no erroneous prediction, when the keyword used is “#apple”. The net accuracy of this algorithm is about 93.75%,89.95% and 90.06% when the keywords are “#apple”, “#trump” and “#brexit” respectively. In all the three test cases, the logistic regression classifier performed pretty well.

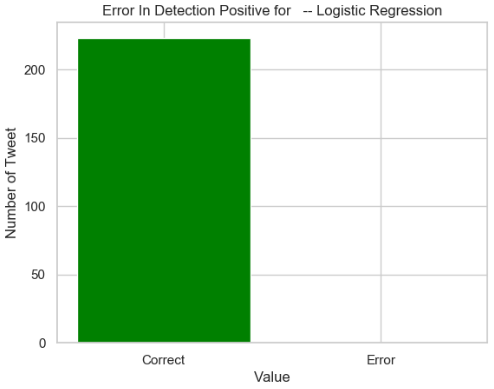

Fig 3: The experimental result of accuracy of Logistic Regression classification, when the polarity is positive and the keyword is “#apple”

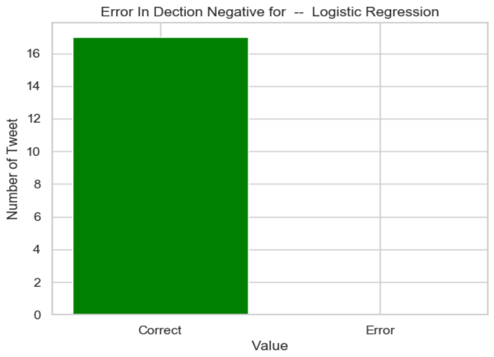

Fig 4: The experimental result of accuracy of Logistic Regression classification, when the polarity is negative and the keyword is “#apple” respectively is shown above.

Table I: The accuracy of logistic regression classifier in three different test cases

Keyword	Accuracy
#apple	93.75%
#trump	89.95%
#brexit	90.06%

C. Support Vector Machine Classifier(SVM)

SVM is a machine learning algorithm which can be used both for classification or regression problems. The main objective of the SVM is to find a hyperplane in a N-dimensional plane, where N is the number of different features that distinctly classifies the data points. In order to separate two classes, there can be any number of hyperplanes possible, but we choose the one which have the maximum margin. Data falling on the either side of the hyperplane can be attributed to different classes. The hyperplane is drawn with the help of mathematical functions called kernels. Different types of kernels are linear, sigmoid, non-linear, etc.

For our paper we observed that for positive polarity, it has successfully predicted the correct polarity for about 225 tweets and there is no error in positive polarity detection. Similarly, for negative polarity, it has predicted correct polarity for 15 tweets with no erroneous prediction. The net accuracy of this algorithm is about 93.75%,91.08% and 90.78% when the keywords are “#apple”, “#trump” and “#brexit” respectively.

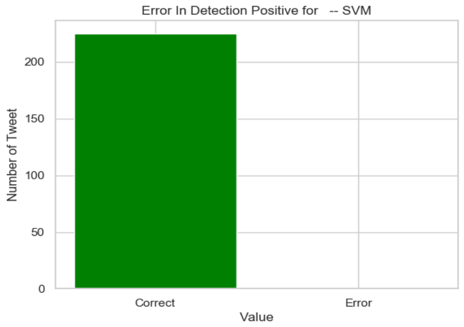

Fig 5: The experimental result of accuracy of SVM classification, when the polarity is positive.

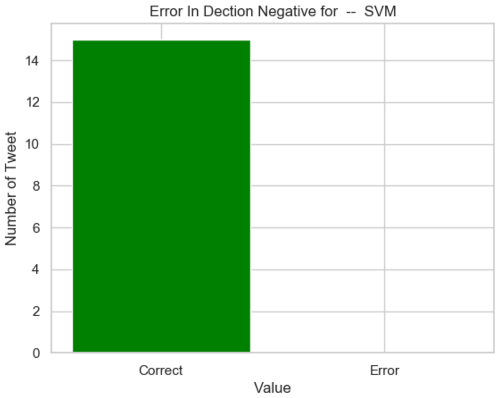

Fig 6: The experimental result of accuracy of SVM classification, when the polarity is negative.

Table II: The accuracy of Support Vector machine(SVM) classifier in three different test cases

Keyword	Accuracy
#apple	93.75%
#trump	91.08%
#brexit	90.78%

D. Decision Tree Classifier

Similar to the support vector machines, both classification and regression problems can be solved using decision tree classifiers. In a decision tree, internal nodes are tests and leaf nodes are categories. The internal nodes test one attribute and each branch selects one value for the attribute. We usually use the attribute which gives the most information. The leaf node predicts category or class. They can be used for generally categorically value functions, other than Boolean functions. The decision tree classifier has successfully predicted the correct polarity for about 213 tweets and there is error in about 12 tweets in positive polarity detection. Similarly, for negative polarity, it has predicted correct polarity for 14 tweets with 1 erroneous prediction. The net accuracy of this algorithm stands at about 89.16%,87.13% and 86.23% when the keywords are "#apple", "#trump" and "#brexit" respectively.

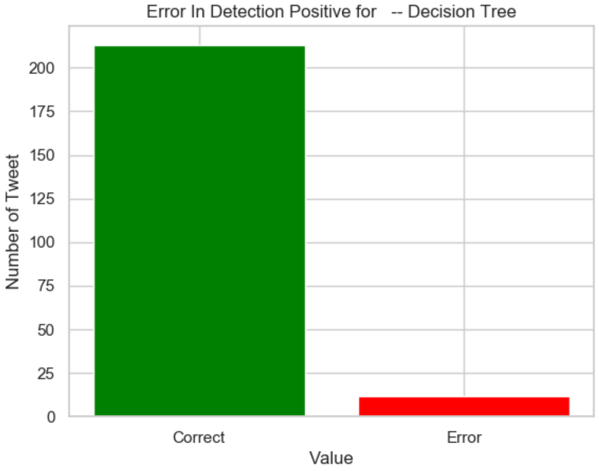

Fig 7: The experimental result of accuracy of Decision Tree classification, when the polarity is positive.

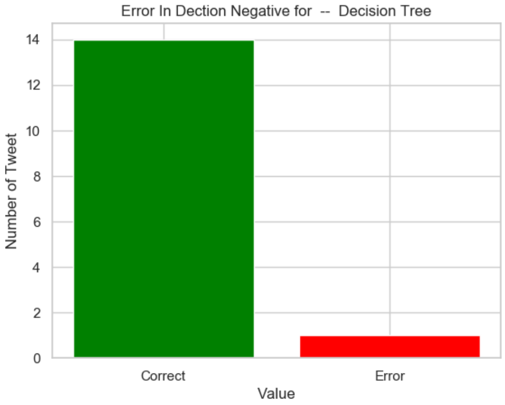

Fig 8: The experimental result of accuracy of Decision Tree classification, when the polarity is negative.

Table III: The accuracy of Decision Tree classifier in three different test cases

Keyword	Accuracy
#apple	89.16%
#trump	87.13%
#brexit	86.23%

E. Random Forest Classifier

Random Forest classifier is a supervised learning algorithm. The working principle of the random tree classifier can be broadly divided into two stages. In the first stage the randomly selected data samples are used for the creation of the decision tree. Thereafter, based on the prediction from every tree the best solution is chosen by means of the votes. It is also used in image classification problems [9]. Firstly, the random samples are selected from a given dataset, then from each sample it creates decision trees and gets a prediction tree from each of those tree. Then, voting is performed for each prediction, and consequently the prediction result associated with the most votes is considered as the final prediction. The random forest classifier has successfully predicted the correct polarity for about 224 tweets and there is error in 1 tweet in positive polarity detection. Similarly, for negative polarity, it has predicted correct polarity for 13 tweets with 2 erroneous predictions.

The net accuracy of this algorithm stands at about 94.16%,91.76%and 92.23%when the keywords are "#apple", "#trump" and "#brexit" respectively. So, the Random Forest Classifier worked best among all the five classifiers, in all the three test cases.

Table IV: The accuracy of Random Forest classifier in three different test cases

Keyword	Accuracy
#apple	94.16%
#trump	91.76%
#brexit	92.23%

IV. RESULTS

As, the twitter data is first pre-processed to construct the dataset the result of pre-processed analysis of the tweets with the help of Text blob is shown below: -

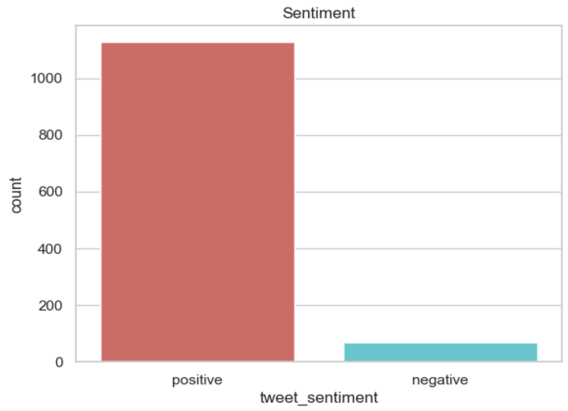

Fig 9: Result of pre-processed analysis of the tweets with the help of Text blob

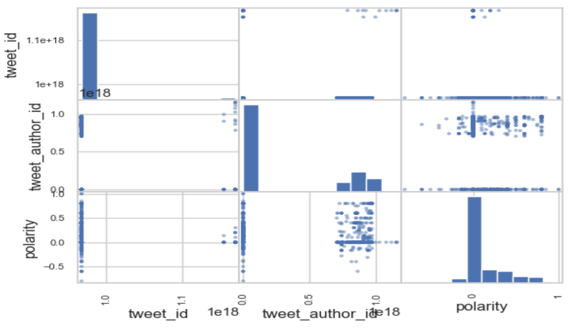

Fig 10: The data visualization histograms.

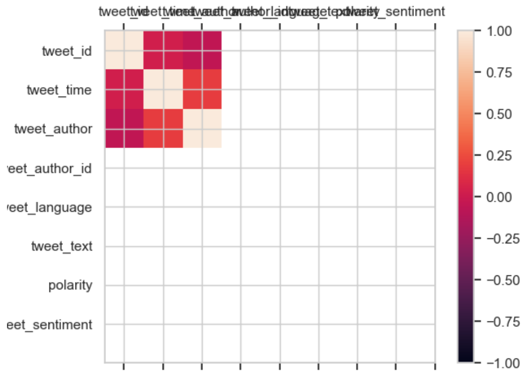

Fig 11: The correlation matrix of the parameters such as tweet_id, tweet_time, tweet_author, etc.

Table V: Experimental result of the proposed work of each individual classifier, when the keyword is "#apple"

CLASSIFIER	ACCURACY	PERFORMANCE
Naïve Bayes	49.58%	Worst
Decision Tree	89.16%	Average
SVM	93.75%	Average
Logistic Regression	93.75%	Average
Random Forest	94.16%	Best

Table VI: Experimental result of the proposed work of each individual classifier, when the keyword is "#trump"

CLASSIFIER	ACCURACY	PERFORMANCE
Naïve Bayes	67.94%	Worst
Decision Tree	87.13%	Average
Logistic regression	89.95%	Average
SVM	91.08%	Average
Random Forest	91.76%	Best

Table VII: Experimental result of the proposed work of each individual classifier when the keyword is "#brexit"

CLASSIFIER	ACCURACY	PERFORMANCE
Naïve Bayes	68.11%	Worst
Decision Tree	86.23%	Average
Logistic Regression	90.06%	Average
SVM	90.78%	Average
Random Forest	92.23%	Best

The graphical representation of the accuracies for each classifier, when the keyword is:

(A) #apple

(B) #trump

(C) #brexit

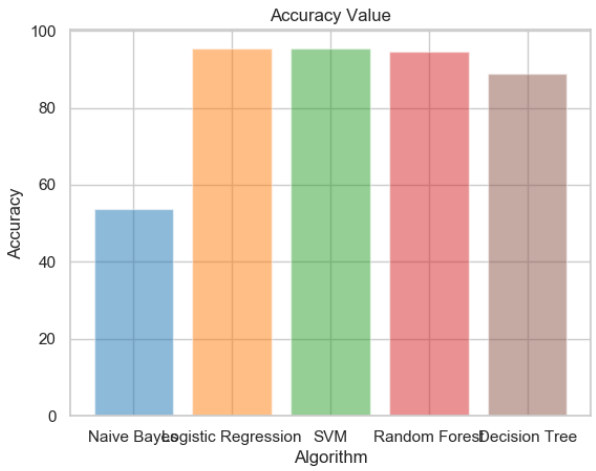

Fig 12: Accuracy graph for classifiers using "#apple"

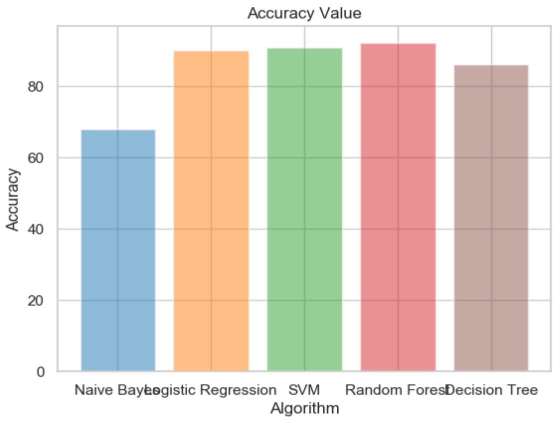

Fig 13: Accuracy graph for classifiers using "#trump"

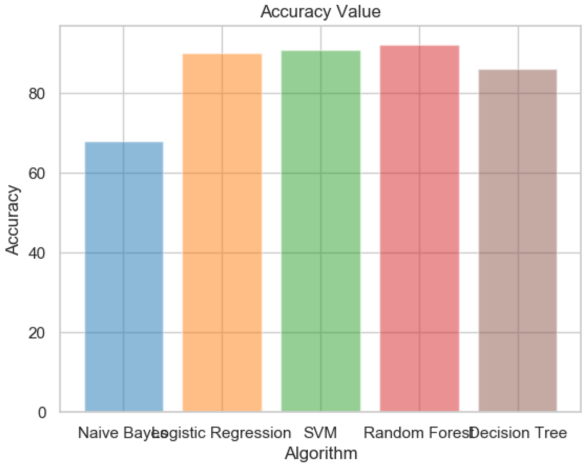

Fig 14: Accuracy graph for classifiers using "#trump"

From the accuracies so obtained we can clearly recognize that the Naïve Bayes algorithm works worst in all the three test cases. So, it is not at all suitable to use this approach for sentiment analysis prediction. The logistic regression approach and the SVM classifier had an above average performance. Here, in our experimental result, Random Forest worked best in all the three test cases. So, we can conclude that the random forest approach works best with polarity detection, which was unknown before.

V. CONCLUSION AND FUTURE WORK

In this paper, we have applied different machine learning approaches to the dataset obtained by live streaming of the twitter data based on a keyword. Here, the key word we have used are "#apple"," #trump" and" #brexit", which can be changed to any other keyword or hashtag. Then the polarity attribute of each tweet is pre-processed and added to the dataset with the help of text blob inbuilt analyzer. Then 80% of the dataset is trained and all the algorithms are tested on the remaining 20% of the data to check for their accuracies. This, paper, is first of its kind where algorithms like Logistic Regression and Random Forest are applied to determine for polarity from a piece of text and also a comparative analysis is done among a bunch of algorithms.This paper is also first of its kind, where the accuracies of each algorithm is calculated on two opposite polarities separately, that is, the accuracy is calculated both on positive and negative tweets of algorithms separately. And, we had successfully found out from our experimental data that Random Forest Algorithm works best in polarity detection, which was unknown before. Future work will include detecting and giving an explanation as to why some algorithms are performing best and why some are worst in polarity detection, and the sensitivity of the algorithms towards emoji. The use of emoji and other symbols has increased the complexity of sentiment analysis, and some algorithms may not comprehend emoji and symbols, which are very critical in determining the polarity of a text. So, it is important to see why and which algorithms specially had failed to detect special symbols. Also, in future course of work I wish to detect those particular words whose polarity cannot be detected by some approaches and thereby they failed to predict the polarity, and as to why some algorithms cannot successfully detect the polarity after training.

REFERENCES

1. Rambocas, Meena (2013). Marketing research: The role of sentiment analysis FEP-WORKING-PAPER-SERIES.
2. Von Helversen, Bettina & Abramczuk, Katarzyna & Kopeć, Wiesław & Nielek, Radoslaw. (2018). Influence of Consumer Reviews on Online Purchasing Decisions in Older and Younger Adults. Decision Support Systems.10.1016/j.dss.2018.05.006.
3. Taboada M, Brooke J, Tofiloski M, Voll K, Stede M. Lexicon-Based Methods for Sentiment Analysis. Computational Linguistics. 2011;37(2):267-275.
4. Asghar, Dr. Muhammad & Kundi, Fazal & Khan, Aurangzeb & Ahmad, Shakeel. (2014). Lexicon-Based Sentiment Analysis in the Social Web. journal of basic and applied scietific research. 4. 238-248.
5. Ikonomakis, Emmanouil & Kotsiantis, Sotiris & Tampakas, V.. (2005). Text Classification Using Machine Learning Techniques. WSEAS transactions.on.computers.4.966-974.
6. Rezapour, Rezvaneh & Wang, Lufan & Abdar, Omid & Diesner, Jana. (2017). Identifying the Overlap between Election Result and Candidates' Ranking Based on Hashtag-Enhanced, Lexicon-Based Sentiment Analysis. 93-96. 10.1109/ICSC.2017.92.
7. Hasan, Ali & Moin, Sana & Karim, Ahmad & Shamshirband, Shahaboddin. (2018). Machine Learning-Based Sentimental Analysis for Twitter Accounts. Mathematical and Computational Applications. 23. 11.10.3390/mca23010011.
8. Khatua, A.; Ghosh, K.; Chaki, N. Can# twitter trends predict election results? Evidence from 2014 Indian general election. In Proceedings of the 48th Hawaii International Conference on System Sciences, Kauai, HI, USA, 5-8 January 2015; pp. 1676–1685.
9. Xu, Baoxun & Ye, Yunming & Nie, Lei. (2012). An improved random forest classifier for image classification. 2012 IEEE International Conference on Information and Automation, ICIA 2012. 795-800.10.1109/ICInfA.2012.62

AUTHORS PROFILE

Sagnik Kar, currently pursuing his B.Tech in Information Technology from Narula Institute of Technology, Agarpara. He will be completeing his degree of B.Tech in the year 2020. He is immensely interested in machine learning and deep learning.

Mousumi Saha, received her B.Tech and M.Tech from West Bengal University of Technology, currently recognized as Maulana Abul Kalam Azad University of Technology, in Computer Science & Engineering. Currently she is working as Assistant Professor in the Dept. of Computer Science & Engineering at Narula Institute of Technology, Agarpara. Her research interest includes Artificial Intelligence, machine learning, deep learning.

Tamasree Biswas, received her B.Tech in Information Technology and M.Tech in Computer Science & Engineering from West Bengal University of Technology, currently recognized as Maulana Abul Kalam Azad University of Technology. Currently she is working as Assistant Professor in the Dept. of Information Technology at Narula Institute of Technology, Agarpara. Her research interest includes image processing, data mining, machine learning and deep learning.