Predicting the Popularity of Political Parties through Ensemble Learning

Teenu Sharma[1], Ankita Bhargava[2], Shruti Jain[3]

[1,2]*Department of Computer Science and Engineering, Pacific Institute of Technology, Udaipur, Rajasthan, India*

[3]*Department of Computer Science and Engineering, Indore Institute of Science and Technology, Indore, Madhya Pradesh, India*

***Abstract*— With the advancement in Technology, social media has become a part of our daily life. People use it to share their day-to-day activities, likes, dislikes, opinions regarding any product, service or event. The micro-blogging website Twitter is a rich source of opinionated content where almost 500 million tweets are sent every single day. This rich opinionated content can be used for analysis, studies, research and it can provide beneficial results. In this paper, tweets are extracted from Twitter for the upcoming India General Elections 2019 and Sentiment Analysis (SA) is performed on it. Three classification algorithms- Naive Bayes (NB), Support Vector Machine (SVM) and k-Nearest Neighbor (k-NN) are used to assign polarity to tweets as positive or negative. Then, the accuracy of predictions is improved through Ensemble Learning and based on this, the popularities of both the parties is calculated and compared.**

***Keywords*— Classification, Classifier Combination, Ensemble Learning, Hard Voting, Majority Voting, Multiple Classifier System, Natural Language Processing (NLP), Sentiment Analysis**

I. INTRODUCTION

Now-a-days everything is becoming digital. Life is almost nothing without data. Data is growing exponentially. Each and every domain of life is directly or indirectly connected to data and on the information processed from it. With the advances in technology, it has become very easy to generate, collect and process data. This data is a blessing for Machine Learning.

Machine Learning is an application area of Artificial Intelligence (AI) which focuses on the development of computer programs (i.e. machines) in such a way that they automatically access the data, use it to learn by themselves and then, process real-time data on their own without the intervention of humans.

One of the most trending fields for machine learning, with enormous amount of data is Social Media. Social Network Platforms like Facebook, Twitter, Instagram etc. have gained popularity because they are easy modes of communication and information. The most popular platform is Twitter where people communicate with each other by posting tweets, retweeting them, liking and commenting on tweets, uploading pictures and sharing interesting videos. One of the most interesting aspects of Twitter is that people express their sentiments in their tweets and comments. These sentiments show their satisfaction or dissatisfaction, their liking or disliking towards something or in general, it can be known whether people's opinion is positive, negative or neutral towards any event or thing. This is useful for research in politics. "Politics using Social Media" is a popular research domain.

In this research, Twitter data is used to predict the popularity of political parties for the upcoming Lok Sabha Elections 2019. The datasets are downloaded from the micro-blogging website Twitter for the parties participating in the upcoming India General Elections 2019 and Sentiment Analysis (SA) is carried out on it. Three classification algorithms- Naive Bayes (NB), Support Vector Machine (SVM) and k-Nearest Neighbor (k-NN) are applied to predict the polarity of tweets as positive and negative. Then, Ensemble Voting Classifier is applied to improve the accuracy of the prediction of tweets as positive or negative. The following problems are taken into consideration:

- Determining the polarity of a tweet as positive or negative.
- Improving the accuracy of predictions.
- Reliability on a single algorithm for predictions.
- Calculating the popularity of political parties.

The implementation of the proposed method is carried out in "Python" programming language. The sections of this research paper are divided as follows. Section I describes the needs and advantages of Sentiment Analysis. Section II describes the research work carried out by other researchers in the field of sentiment analysis and ensemble learning. Section III describes the implementation of the proposed methodology. Section IV shows the experimental steps and the calculation and comparison of the experimental results. Section V describes the conclusion with the obtained results and the future work for this research.

II. RELATED WORK

R. Jose and V. S. Chooralil in [1], have tried to predict the election result for the Delhi Election between Arvind Kejriwal and Kiran Bedi using Ensemble Learning. They have used a new combination of classifiers- SentiWordNet Classifier, Naive Bayes Classifier and Hidden Markov Model Classifier. Their proposed system consists of six modules namely- Data Acquisition, Pre-processing, Sentiment Classification using SentiWord-Net, Sentiment Classification using Naive Bayes, Sentiment Classification using HMM, Sentiment Classification using ensemble

approach. They have used lexicon based classifier. The Ensemble Based Classifier obtained the highest accuracy and the results showed that Arvind Kejriwal has a higher public sentiment than Kiran Bedi.

Sharma S. and Shetty N.P., in [2] have carried out Sentiment Analysis (SA) on user-generated tweets to get the reviews about major political parties for The United Kingdom General Elections 2017. Three algorithms, Naive Bayes (NB) Classifier, Support Vector Machines (SVM), and k-Nearest Neighbour (k-NN) are used to determine the polarity of tweets as positive, negative and neutral, and finally based on these polarities, predictions are made to know which party is likely to perform better in the upcoming elections. The results show that k-NN algorithm gives the most accurate results.

Yun Wan et. al. [3] have performed Sentiment Analysis for Airline Services. Very less research has been done for this domain. Data is extracted from Twitter and six classification algorithms- Naive Bayes, Support Vector Machine, Bayesian Network, C4.5 Decision Tree and Random Forest are applied including the Majority Voting Ensemble Classifier. The results show that the Ensemble Classifier has outperformed for the Airline service Twitter dataset.

Catal C. et. al. [4] This research studies the benefit of mutiple classifier system for Turkish sentiment analysis. They have used three classification algorithms- Naive Bayes (NB), Support Vector Machine (SVM) and Bagging in conjunction with Majority Voting. The experimental results show that the Voting method has increased performance on Turkish sentiment classification datasets. Hence, multiple classifier approach is proved to be good for sentiment analysis.

Jain A. et. al. [5] have analyzed the sentiments of users using data mining classifiers. Single classifiers used are- k-Nearest Neighbor, Random Forest, BaysNet and NaivBays. Ensemble classifiers are also used which gives higher accuracy compared to the single classifiers.

III. METHODOLOGY

A. Collection of Data

To perform sentiment analysis for predicting the popularity of political parties through ensemble learning, the first requirement is to download tweets from the micro-blogging platform Twitter through Twitter API and Tweepy.

For the India General Elections 2019, two political parties have a major role in it. It's the Bhartiya Janta Party and the Indian National Congress. So, this sentiment analysis is performed for these two parties.

Tweets are downloaded and two datasets are prepared, one for BJP (Bhartiya Janta Party) and the other for INC (Indian National Congress). Tweets are downloaded by searching for relevant hashtags in the tweets. For example, the tweets for the Bhartiya Janta Party will have the hashtags like "#LokSabhaElections2019" and "#BJP", so the tweets are searched for these hashtags, they are downloaded and a dataset is created for "BJP". In the similar way, tweets with the hashtags like "#LokSabhaElections2019" and "#Congress" are searched, downloaded and a separate dataset is created for "Congress".

B. Selection of Random Data

The datasets for both the parties consists of a large number of tweets. To perform sentiment analysis on it, we randomly select 250 tweets from each dataset and save it to another file. These random tweets contain both positive and negative tweets.

C. Cleaning of Datasets

The tweets downloaded from Twitter are not appropriate for directly performing sentiment analysis on it. There are a lot of ambiguous words in them that needs to be removed. Also, the tweets contain links and usernames which do not play any role in sentiment analysis. So, they are also removed while cleaning the tweets.

Blank spaces and punctuation marks are also eliminated. Stopwords do not play any role in deciding the polarity of a tweet so stopwords are also eliminated. After cleaning the tweets, tokenization is performed. Now, the tweets are divided into individual tokens.

D. Labeling of the Opinionated Words

The tweets are divided into single words called "tokens". Now each token of a tweet is checked upon a dictionary and a score is assigned to it. The dictionary consists of two files of positive and negative words respectively. The dictionary can be edited according to user requirement. The words found in the positive and negative dictionary are mentioned in separate columns of "positive words" and "negative words" each. Then, the tokens in both the columns are counted and are mentioned in new columns of "positive count" and "negative count". The comparison between these counts help in determining the overall polarity of the tweet.

E. Polarity Assignment

The polarities of the tweets are categorized into two classes, either positive or negative. The integer values of the columns "positive count" and "negative count" are compared and the result is assigned to a new column "polarity". If "positive count" is greater than or equal to the "negative count" then the tweet is assigned a 1 (means positive), otherwise a 0 (means negative).

F. Classification Algorithms

After all the tweets are assigned polarities, the datasets are split into training and test sets. Then, three classification algorithms- Naive Bayes (NB), Support Vector Machine (SVM) and k-Nearest Neighbor (k-NN) are applied to the datasets.

G. Ensemble Voting Classifier

After applying the three classification algorithms, an ensemble voting classifier is applied on the datasets. It is a type of '*majority voting*' or '*plurality voting*'. The ensemble vote classifier takes votes from all the classifiers and the predicted value that get more than half of the votes wins the voting and is considered as the final prediction. It is called a 'stable prediction'.

To understand how does majority voting works, let us consider an example. This is an example of hard voting. Here, we have classifier C and we have to predict the class label Y via majority voting.

$$Y = mode\ \{ C1\ (x), C2\ (x), ..., C_m\ (x) \}$$

If we make an assumption that there are three classifiers that have to classify a training sample to either Class 0 or Class 1, then if

- Classifier 1 -> Class 0
- Classifier 2 -> Class 0
- Classifier 3 -> Class 1

then, Y = mode {0,0,1} = 0

Hence, via majority vote, we would classify the sample to "Class 0".

IV. EXPERIMENTAL RESULTS

A. Experimental Steps

1. The datasets are downloaded directly from Twitter through Twitter API and Tweepy in Python for hashtag "LokSabhaElections2019".
2. 250 tweets are selected randomly from both the datasets.
3. The datasets consists of ambiguous words so the datasets are cleaned. Blank spaces, stopwords, punctuation marks are removed and then, tokenization is applied.
4. The tokens are checked upon a dictionary consisting of two files of positive and negative words each and polarity is assigned to each word. After assigning polarity to each word, the sum of the values is calculated and the overall tweet is assigned polarity either positive or negative.
5. After polarity assignment, three classification algorithms - Naive Bayes (NB), Support Vector Machine (SVM) and k-Nearest Neighbor (k-NN) are applied on the datasets which are divided into 80:20 ratios as training and test sets.
6. Ensemble Voting Classifier is applied to the datasets.
7. The accuracy of all the classifiers is calculated.
8. Results of all the algorithms are compared.
9. The performance of both the parties is compared and the popularity is calculated.

B. Experimental Results

The accuracies for the classification algorithms are calculated on the basis of their confusion matrix. The formula for calculating the accuracy score from a confusion matrix is:

$$Accuracy\ Score = \frac{TP+TN}{TP+TN+FP+FN}$$

where,

- *TP or True Positive* is the count of correctly predicted occurrence of an event.
- *FP or False Positive* is the count of incorrectly predicted occurrence of an event.
- *TN or True Negative* is the count of correctly predicted non-occurrence of an event.
- *FP or False Negative* is the count of incorrectly predicted non-occurrence of an event.

Table I

Accuracy Obtained for the Single Classifiers

Classifiers	Political Parties		Average Accuracy (%)
	BJP (%)	INC (%)	
Naive Bayes	78	82	80
k-Nearest Neighbor	80	84	82
Support Vector Machine	80	86	83

Table I shows the overall accuracies and the average accuracies obtained for the three classification algorithms- NB, SVM and k-NN for both the parties- BJP and INC. It is found out that SVM has the highest average accuracy of 83%.

This average accuracy is further improved by applying Ensemble Voting Classifier to 86% as shown in Table II.

Table II

Accuracy Obtained for Ensemble Voting Classifier

Classifier	Political Parties		Average Accuracy (%)
	BJP (%)	INC (%)	
Naive Bayes	78	82	80
k-Nearest Neighbor	80	84	82
Support Vector Machine	80	86	83
Ensemble Voting Classifier	84	88	86

Also, the mean accuracies have proved to be the highest for Ensemble Voting Classifier. See Table III.

Table III

Mean Accuracies for All Four Classifiers

Classifiers	Political Parties	
	BJP (%)	INC (%)
Naive Bayes	80.79	71.18
k-Nearest Neighbor	79.18	77.21
Support Vector Machine	79.99	76.02
Ensemble Voting Classifier	82.02	77.63

To predict the popularity of political parties, the percentage of positive and negative tweets are calculated for both the parties as shown in Table IV.

Table IV

Tweet Polarities for both the Parties according to the Proposed Method

Tweet Polarity	BJP (%)	INC (%)
Positive Tweets	81.2	77.2
Negative Tweets	18.8	22.8

Clearly from the table, it can be inferred that the party which is likely to perform better in the upcoming Lok Sabha Elections 2019 is the Bhartiya Janta Party (BJP) with a percentage of positive tweets equal to 81.2%.

V. CONCLUSION AND FUTURE WORK

From the proposed work, we can see how datasets can be downloaded directly from the micro-blogging website Twitter and can be used for decision-making. These predictions are helpful for election campaigning and for the performance prediction of the political parties. They are also helpful for the people to get an idea of present reputation of any political party or politician.

Also, the proposed model of Ensemble Voting Classifier has proved to improve the accuracy of the predictions for determining the popularity of political parties for the upcoming Lok Sabha Elections 2019. Hence, it is beneficial to use this method instead of using a single classifier algorithm for Twitter Sentiment Analysis. This research work has a lot of scope for the advancement of the elections on an individual level as well as on the political level.

In future, more work can be done on this to further improve the performance of the algorithm. The proposed method can be extended in many ways for using it in election campaigning. Also, other methods can be adopted to perform sentiment analysis on Twitter data like including emoticons, images, links etc. More cleaning methods can be used in the preprocessing of tweets like Stemming, Lemmatization etc.

REFERENCES

[1]. R. Jose and V. S. Chooralil, "Prediction of election result by enhanced sentiment analysis on twitter data using classifier ensemble Approach," *2016 International Conference on Data Mining and Advanced Computing (SAPIENCE)*, Ernakulam, 2016, pp. 64-67.

[2]. Sharma S., Shetty N.P. (2018) Determining the Popularity of Political Parties Using Twitter Sentiment Analysis. In: Satapathy S., Tavares J., Bhateja V., Mohanty J. (eds) Information and Decision Sciences. Advances in Intelligent Systems and Computing, vol 701. Springer, Singapore

[3]. Wan, Yun and Qigang Gao. "An Ensemble Sentiment Classification System of Twitter Data for Airline Services Analysis." *2015 IEEE International Conference on Data Mining Workshop (ICDMW)* (2015): 1318-1325.

[4]. Cagatay Catal, Mehmet Nangir, A Sentiment Classification Model Based On Multiple Classifiers, Applied Soft Computing Journal http://dx.doi.org/10.1016/j.asoc.2016.11.022

[5]. A. P. Jain and V. D. Katkar, "Sentiments analysis of Twitter data using data mining," *2015 International Conference on Information Processing (ICIP)*, Pune, 2015, pp. 807-810.

[6]. Zhaoyu Li, "*Naive Bayes Algorithm For Twitter Sentiment Analysis And Its Implementation In Mapreduce*", M. Sc. Thesis, Univ. of Missouri, Dec. 2014.

[7]. Zhi-Hua Zhou, "*Ensemble Learning*", Nanjing University, Nanjing 210093, China.

[8]. Minqing Hu and Bing Liu. "Mining and Summarizing Customer Reviews." Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004), Aug 22-25, 2004, Seattle, Washington, USA.

[9]. Dr. Saurabh Mukherjee, Neelam Sharma, "*Intrusion Detection using Naive Bayes Classifier with Feature Reduction*", Dept. of Comp. Sc., Banasthali Univ., Jaipur, Raj., 2011.

[10]. H. M. Kubade, "*The Overview Of Bayes Classification Methods*", IJTSRD, Vol. 2, Issue 4, Dept. of I.T., Nagpur, India, May-Jun 2018.

[11]. A. McCallum, K. Nigam, "*A Comparison of Event Models for Naive Bayes Text Classification*", School of Comp. Sc., C. M. Univ., Pittsburgh, PA.