DCU SCHOOL OF COMPUTING
ASSIGNMENT SUBMISSION

Student Name(s):	**Harshali Patil**
Student Number(s):	**18212797**
Email Id:	harshali.patil2@mail.dcu.ie
Programme:	**Masters in Computing (Data Analytics)**
Project Title:	**Twitter Data Analysis of 2019 Indian Election**
Module code:	**CA685**
Supervisor:	**Prof. Yvette Graham**
Project Due Date:	**11/Aug/2019**

Declaration

I declare that this material, which I now submit for assessment, is entirely my own work and has not been taken from the work of others, save and to the extent that such work has been cited and acknowledged within the text of my work. I understand that plagiarism, collusion, and copying is a grave and serious offence in the university and accept the penalties that would be imposed should I engage in plagiarism, collusion, or copying. I have read and understood the Assignment Regulations set out in the module documentation. I have identified and included the source of all facts, ideas, opinions, viewpoints of others in the assignment references. Direct quotations from books, journal articles, internet sources, module text, or any other source whatsoever are acknowledged, and the source cited are identified in the assignment references.

I have not copied or paraphrased an extract of any length from any source without identifying the source and using quotation marks as appropriate. Any images, audio recordings, video or other materials have likewise been originated and produced by me or are fully acknowledged and identified.

This assignment, or any part of it, has not been previously submitted by me or any other person for assessment on this or any other course of study. I have read and understood the referencing guidelines found at http://www.library.dcu.ie/citing&refguide08.pdf and/or recommended in the assignment guidelines.

I understand that I may be required to discuss with the module lecturer/s the contents of this submission.

I/me/my incorporates we/us/our in the case of group work, which is signed by all of us.

Signed: HARSHALI PATIL

Twitter Data Analysis of 2019 Indian Election

Harshali Patil
Dublin City University, School of Computing, 2018/19
Dublin, Ireland
harshali.patil2@mail.dcu.ie

***Abstract*—Twitter is a popular platform for microblogging where millions of users share their views on current affairs and interact with each other. Twitter has become one of the largest sources for extraction of information that can subsequently be analyzed in terms of content to find patterns or make predictions. Traditional content analysis can take up to weeks during the electoral process while at the same time automatic sentiment analysis of political tweets can be carried out by machine learning algorithms in seconds. This paper proposes predictive methods of analyzing Twitter traffic that discusses the Indian Elections of 2019. We have extracted sentiment information from microblogs to carry out the analysis of likes and dislikes for each party. The work also describes emotions and hashtag analysis from the tweets and develops a tool for providing timely analysis of the election process and public sentiment.**

Keywords— microblogs, Twitter, sentimental analysis, election.

I. INTRODUCTION

The world's largest democracy, India has witnessed the General Election for 17th Lok Sabha in the month of May 2019. The General Elections plays a vital role to form a government and elect the Prime Mister of India. These elections are conducted every five years. The major parties for the 2019 elections are Bhartiya Janta Party (BJP) and Indian National Congress (Congress). For years, standard methodologies of analyzing the views and intentions of the society during the election process have been used by the researchers but it is time-consuming and involves a lot of human efforts. It is undeniable that the social media platforms are engaged with the millions of users which made it easier to take advantage of real time analysis. Every current topic and scenario are supported by comments, reviews due to the popularity of social media, which reflects views of the people on the topic. Although there are some challenges with social media like security, privacy but still it allows users to share, collaborate and engage on various sites like Facebook, Google etc. According to Internet World Stats [1], India is now at the second position for being the country of second largest Internet users. Specifically, in India, there are 7.75 active users of Twitter in 2019 [2]. Furthermore, Twitter is accepted by most world leaders and ministers for political campaigns.

Twitter is a microblogging site where people can share their opinion and thoughts on various aspects of life in real time. Moreover, with the rapid growth of text-based social media, opinion mining and predictive analysis is assisted in plethora of areas like reviewing products, discussing current issues etc. It has motivated people to get involved in the political activities by sharing their opinions about the party and candidate in the form of microblogs. Microblogs are generally referred as "Tweets" and are short messages: less than 280 words. These messages carry sentiments of people related to the social and political issues. Thus, the sentimental analysis can provide the views of people for a party during the Election period.

This paper examines the sentiments expressed by people for the parties, which are BJP and Congress, participating in Indian Election of 2019 and categorize them in positive and negative sentiment. The proposed paper also examines emotions associated with the tweets tweeted during the Election period. Natural Language Processing with some Machine Learning Algorithms are used to obtain the results which is later compared against the actual sentiment obtained in the election results.

The rest of the paper is structured as follows: Section II provides the review of the previous work done in this field. Section III provides the details of the data collection and pre-processing of the Indian Election twitter dataset. While Section IV relates to the research methodologies used for the prediction of Indian Election using twitter data. Evaluation of the models is represented in Section V. Lastly; the summary and Future work of the paper is given in Section VI.

II. LITERATURE REVIEW

Years ago, text based work was mostly focused on metaphoric models, document management, evidentiality in text [3] [4]. In decades, the area of sentiment analysis and opinion mining has enjoyed a large increase in research activities. [5] . Textual Information can be divided into two groups: factual and sentimental. Facts carry objective information while the sentiments carry subjective information. For example: "We should decrease the price of oil", is a subjective expression. On the other hand, "Trump is the President of US", is an objective expression as it contains Claim but no sentiments regarding Trump being the President of US.

The term "sentiment analysis" and "Opinion mining" are often used interchangeably [6].Sentiment analysis (sentiment mining, opinion mining, or polarity classification) deals with emotions, ideas, judgements or emoticons that can be present explicitly or implicitly [7] [8]. The terms "Holder, Topic, Claim and Sentiment" describes an Opinion. [8] .

Earlier the sentiment analysis was focused on long text-based documents, because sentiment contained in short messages are compact or explicit. Work in sentiment analysis and opinion mining have seen a dramatic rise due to incremental improvement in text-based social media, Machine Learning methods and availability of large volume of datasets [5]. [9] used microblogs from twitter to classify sentiment and discovered that the it is easy to classify sentiments from microtext than from blogs. In the same manner, Twitter data has been used in different domains like Stocks, politics and social- economic phenomenon. [10] [11]. In [11] , investigated whether twitter is right platform or not

to predict the polling results. They gathered tweets for the German federal election 2009 and found analysis of the political tweets correspondence to the election results [12].

There are typically two techniques to classify sentiments: Knowledge-based approach and Machine Learning approach [13]. Knowledge-based techniques includes use of lexicons which are basically a corpus of known and precompiled words. Dictionary tools such as WordNet and Sentiful are used to examine the polarity of lexicons. [14] [15]. On the other hand, Machine Learning methods involves training a model using past examples and experiences which will improve the performance of the system. It is further classified into Supervised and Unsupervised Learning. Supervised Learning deals with subjective data and the selection labelled training data plays a vital role in training a model. [16] [17].

[12] collected tweets that contained the phrase "McCain" or "Obama" from the US presidential Election of 2012. They counted sentimental polarity words taken from pre-existing lexicons. For this, they used the subjectivity lexicon from OpinionFinder, which has around 1,600 and 1200 words referred as positive and negative, respectively. O'Connor et al. used the ratio of positive versus negative message for presidential candidate to define the sentiment score. The obtained count was low, as the lexicon is designed for the standard English and the Twitter messages contains informal English, with emoticons and differently spelled words. They found the new and events are more favorable towards Obama as compared to McCain. It is concluded that the modes of communication should be considered for the textual analysis apart from designing well- suited lexicon. Furthermore, advanced NLP techniques are required for the opinion mining.

In a paper [18] a method is proposed to analyze sentiments expressed by the tweets for US Presidential Election of 2016. They compared these sentiments with the actual polling results to obtain the degree of correlation they share. They used lexicons from OpinionFinder to collect positive and negative words. They calculated sentimental score by subtracting the count of positive words from the count of negative words. If the result is positive, they labelled them as positive tweet. If the result obtained is negative, then it is labelled as negative tweet. In the rest cases, tweet is considered as neutral. Furthermore, Naïve Bayes Algorithm was used for classifying tweets as positive or negative on the basis of hashtags. To implement the Naïve Bayes Algorithm, Joyce and Deng used the National Language Toolkit (NLT). It was observed that the Naïve Bayes algorithm did well in sentimental analysis but did not outperform the lexicon analysis when compared to Trump polling data. While the automatic labelled tweets outperformed the manually labelled tweets in comparison to Clinton polling data. They found the 94% of correlation to the polling data.

In [19]Is Twitter a valid source for predicting the results of US Presidential Election of 2012? Can Topic Modelling be used for extracting the important topics out of the twitter discussion? Do they match with offline discussions? Moreover, they compared the text analysis results of Twitter data with the traditional poll results. They employed Twitter Crawler to use Twitter Streaming API to collect tweets in real time. "Tweepy" library from Python was used to collect the political tweets and they gathered around 39 million tweets. In this paper, Firstly, they analyzed the tweets ranging in the time frame of first and last observed tweets. They observed the number of positive and negative tweets for each candidate. Secondly, they used geo-spatial method of sentimental analysis for identifying the popularity of the candidate in US. Lastly, Authors used Latent Dirichlet Allocation Model which defines probability distribution for extracting the unobserved topic from the tweets posted during US Presidential Election. The results suggested that the Twitter is a valid source of predicting public opinions and trends of the future events.

Prediction of French Election of 2017 using Twitter data proposed in a paper [20], in which author tried to predict the popularity of candidates by using sentimental information from microblogs. They used term frequency to extract more keywords and then used it for classifying tweets. Apart from these keywords they considered most common words used for election in their study such as vote, win, attack, fail. They compared the proposed method of calculating the popularity of candidate with Tumasjan's method. The factor distinguishing both the methods is the neutral comments; the proposed method considered neutral comments in calculating the popularity of candidate while Tumasjan's method do not consider the neutral comments. They found that the prediction of the proposed method is about 2% different from the real polling results while the other method was 38% away from the actual election results. It is concluded that the proposed method surpassed the Tumasjan's method.

Features such as emoticons, emojis and hashtags are very common to microblogs. Many researchers used emoticons, emojis as a feature using pre-trained corpus, as they are a good indicator of writer's emotions and sentiments. [21] [22]

In a [23] research paper "Twitter Sentiment Classification using Distant Supervision" introduced a novel approach for automatic classification of the sentiments from the Tweets [23]. They used tweets with emoticons to train the model as, emoticon reflects the emotions of the Holder. They implemented Naïve Bayes, SVM and Maximum Entropy as a Distant supervised learning algorithm.

Many deep learning methods have been introduced in recent years [24]. [25] Experimented with Convolutional Neural Network (CNN) and showed that a simple CNN with hyper tuning can achieve good results. [26] introduced a better and efficient method of neural network, Long-short term memory (LSTM) for twitter sentiment prediction. They found that LSTM doubles the performance of non-neural model.

III. APPROACH

A. Data Collection:

Microblogging service Twitter is used as a Data source as it contains large volume of political discussions and personal opinions regarding the 2019 Indian General Election. In order to estimate the elections, we collected around 1.5 million tweets from December 2018 to March 2019 prior to elections using RESTful API provided by twitter. We continuously search for the tweets which contains the hashtags like #BJP, #Congress, #Narendramodi, #LoksabhaElection2019, #BJPwins, #Congresswins, #RahulGandhi in order to get the data relevant to the 2019 Indian Election. Twitter provides Application Platform Interfaces (API) that was used to extract the microblogs

from twitter and is free but are limited to per user access token, it restricts user to request the certain amount of data call per window or the frequency of calls made. To overcome this challenge, we refreshed our API calls periodically.

a) Chracterestics of tweets

Data collected from twitter is examined to get the maximum length of the tweet. As per the twitter policy, it allows maximum of 280 characters, but this length is rare in tweets. In order to feed input into the LSTM network, tweet length is an important parameter to form the input sequence matrix.

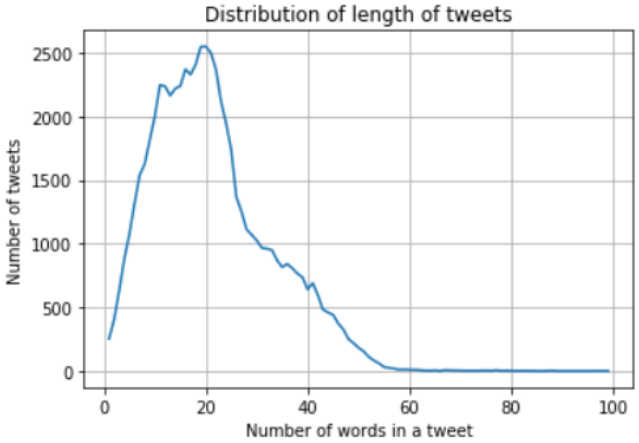

Figure 1:Distribution of tweets in Indian Election dataset

b) Data Pre-Processing:

- Tweets fetched from Twitter API is returned in json format which were further merged into CSV by parsing all JSON data.
- Duplicates tweets are dropped as they doesn't bring new information to the dataset and computationally inefficient.

c) Feature Reduction:

The text of tweet is different from the text of blogs, articles or other text-based documents. It consists of idiosyncratic uses like URL, user mentions, hashtags, emoticons, emojis etc. Thus, we processed tweets as follows:

- Replaced emoticons and emojis with the corresponding CLDR short name using Demojify library.
- Remove Pattern such as user mentions (@) and hashtags (#) , URL (http , https and www links) and special twitter word for retweets ("RT")
- Replaced elongated words or casual language (e.g.: Awwwww or hurrrrraayyyyy) with text which contains an occurrence of a character consecutively no more than twice using a wordnet dictionary.
- Spell Checker is applied to correct the spellings in the tweets.
- Special Characters (e.g.: "" , / . ! [] * &) are removed from the dataset.

d) Normalization:

- Stopwords Removal : There are stop words which are common in human language and are used for sentence composition but are not useful to the model. Therefore, we have removed stopwords from the data using NLTK library which filter outs all the stop words from a sentence.
- Tokenization : After Removing Stopwords from the data, word tokenizer is used from NLTK to get individual words.
- Lemmatization : WordNetLemmetizer is used from NLTK to return dictionary form of a word. (eg: Playing to Play)
- Parts-of-Speech: using a POS tagger , we assigned a tag to a word as adjective. Verb, Noun, Adverb.

e) Feature Extraction:

After getting the normalized state of data, we vectorized he tweets that can be understood by the Machine Learning algorithm. In order to do get vectors of data, TfIdf vectorizer, Keras Tokenizer and Elmo is used.

TfIdf Vectorizer: It gives word a weightage on the basis of how important that word is to a dataset. It consist of two terms:

- *Term Frequency (TF):* It shows how many times a word occurred in the entire dataset.
  TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document).

  Inverse Document Frequency (IDF): It measures how important the word is to our dataset
  IDF(t) = log_e(Total number of documents / Number of documents with term t in it).
  And Lastly, this is how we get weights:
  TFIDF = TF*IDF
- *Keras Tokenizer:* It has two functions:
  - *fits_on_text*: It creates a word index dictionary based on the frequency of word. Indexing starts with 1 as 0 is reserved for padding sentence.
  - *texts_to_sequences:* This method transforms each word in he sentences with a corresponding integer returned from the word index dictionary.
- *Elmo:* It is a pre-trained word embedding model. Elmo works on top of two-layer bidirectional language model. It takes entire sentence in context to calculate word embedding. Elmo built different elmo vectors for a word under different context. For example6

  Text 1: I like to read books
  Text 2: Can you please book tickets for a movie?

  But Elmo is computationally expensive and time consuming.

B. Model:

To create a baseline sentiment model, we chose Multinomial Naïve Bayes classifier and Long-Short Term Memory. State of the art model RNN are suitable for sentiment analysis as they can learn from the past experiences. When the sequence is long enough, RNN become unable to connect information. This problem of short-term dependency is overcome by LSTM. It also consists of several neural layers in its network. LSTM are composed of gates which learns that an information is relevant or not in order to store them

during training. This helps LSTM to learn long-term dependencies.

On the other hand, Multinomial Naïve- Bayes Classifier is a probabilistic model which uses Multinomial Distribution. It counts occurrence of a word in a document. It need discrete features as an input that is obtained using TfIdf vectorizer and the output is generated on the basis of highest probability obtained from the set of probabilities calculated for the text. Multinomial Naïve Bayes algorithm is generally fast and reliable.

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

Where,
P(A|B) is posterior probability
P(B|A) is likelihood
P(A) is prior
And, P(B) is evidence.

In order to classify the tweets, we assigned label 1 as BJP and label 0 as Congress.

a) Model architecture:

LSTM architecture comprises of embedding layer, LSTM units, activation function, Dense layers, input layer, output layer, and other hyperparameter like dropout, max words.

We can fine tune the hyperparameters to have better performance of the model. For our model, LSTM specification is shown below:

TABLE I. SPECIFICATIONS FOR THE LSTM ARCHITECTURE

Parameter	***Description***
LSTM	64 neurons
Dense	1
Dropout	0.5
Activation Function	Sigmoid
Optimizer	RMSprop
loss	binary_crossentropy
Total Parameters	1087999

Sigmoid Function is used because it squishes values between 0 and 1. This helps gates in forget and update an information during the training.

Also, in our training data i.e. Sentiment140, we have two classes namely Positive and Negative. We kept loss as binary cross-entropy as we are interested in binary classification here.

b) Input Output representation

We have used Keras tokenizer as a word embedding with a 500000 tokens vocabulary. Maximum length of a tweet is 167 so a sequence matrix is formed by padding each tweet till the maximum length. This input is fed into the LSTM network through embedding layer. As we want a label for each tweet, we kept shape as [None, 1].

Information gained from the embeddings are extracted by bottom LSTM which consists of input output gates. Input gate control the information by checking the relevance of information and then forwards to higher order LSTM unit. The output gate adjusts the information from the memory cell to represent the current word.

c) Working of LSTM

The working of LSTM can be defined as follows:

$$I^{(t)} = \sigma(W^{(i)} x^{(t)} + U^{(i)} h^{(t-1)})$$
$$f^{(t)} = \sigma(W^{(f)} x^{(t)} + U^{(f)} h^{(t-1)})$$
$$o^{(t)} = \sigma(W^{(o)} x^{(t)} + U^{(o)} h^{(t-1)})$$
$$\tilde{c}^{(t)} = \tanh(W^{(c)} x^{(t)} + U^{(c)} h^{(t-1)})$$
$$c^{(t)} = f^{(t)} \circ \tilde{c}^{(t-1)} + i^{(t)} \circ \tilde{c}^{(t)}$$
$$h^{(t)} = o^{(t)} \circ \tanh(c^{(t)})$$

$I^{(t)}$ represents Input gate
$f^{(t)}$ represents forget gate
$o^{(t)}$ represents output gate
σ represents sigmoid function
$W^{(x)}$ represents weight for respective gates (x)
$h^{(t-1)}$ is the output of the previous lstm block
$x^{(t)}$ is input at current timestamp t
$\tilde{c}^{(t)}$ represents new memory cell
$c^{(t)}$ shows Final memory cell

We have used two LSTM units that can share the same weights. It helps in reducing the risk of overfitting issue of the model. In addition, LSTM learns the temporal dependencies by sharing weights between the units.

C. Experiments

Opinion words shows people's attitude towards a particular topic or discussion and can be used in different aspects to quantify audiences' emotions. In our work, we are focusing on three distinct analysis to get insight of the people's view towards the Indian General Election of 2019.

a) Sentiment Analysis:

Sentiments gives the general feel about what people thinks of the topic. It does not deal with personal specific emotions. It could be defined as "positive" or "negative" responses towards the election in our case. Many researchers also considered "neutral" as another category of sentiment, but we have not used it in our case.

To predict the sentiments associated to Indian Loksabha Election of 2019, we have used Sentiment140 [27]dataset labelled as "positive" and "negative" provided by Stanford University. It contains 1.6 million tweets with the balanced share of both the classes. Sentiment140 dataset is human annotated and contains tweets of different subjects. Data cleaning is applied to the Sentiment140 in the way mentioned above.

Algorithms used for Sentiment Analysis are Multinomial Naïve Bayes and LSTM.

Results for this are shown below:

TABLE II. EVALUATION RESULTS FOR SENTIMENT ANALYSIS USING MULTINOMIAL NAÏVE BAYES

Metrics	Score
Accuracy	0.7534
F1 score	0.7526

TABLE III. EVALUATION RESULTS FOR SENTIMENT ANALYSIS USING LSTM

Metrics	Score
Accuracy	0.8143

F1 score	0.801
Evaluation loss	0.4101

These trained models are used to predict the labels for tweets collected for Indian election.

b) Stance Analysis:

Unlike sentiment analysis where we describe author's sentiment as positive or negative, Stance analysis determines whether the writer of tweet is in favor of or against the target. In our case target is the two major political parties (BJP and Congress) involved in the election. For Stance Analysis, we have used a training dataset consisting of 23610 records and is a subset of the main dataset extracted from the twitter API. It is labelled as BJP and Congress based on hashtags used in the tweets. LSTM model is considered to asses the stance for rest of the tweets.

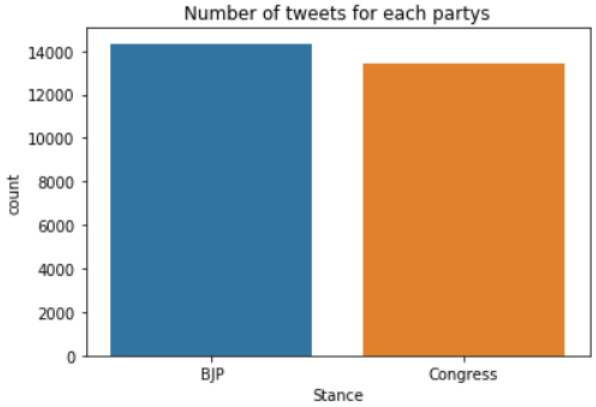

Figure 2: Distribution of tweets with positive and negative sentiment in the Stance data.

TABLE IV. HASHTAGS USED FOR STANCE ANALYSIS

Hashtag	Context
BJP	in favor of BJP
Congress	In favor of Congress

TABLE V. EVALUATION RESULTS FOR STANCE ANALYSIS USING LSTM

Metrics	Score
Accuracy	0.8892
F1 score	0.872
Evaluation loss	0.2596

These trained models are used to predict the labels for tweets collected for Indian election.

c) Emotional Analysis:

It relies on human emotions and sensitivities which expresses various feelings of the author like anger, fear, joy, happiness, sadness etc. The dataset used for this study has 40000 records with 13 different labels representing tweets. But we only considered hate, happiness, relief, sadness, surprise, worry

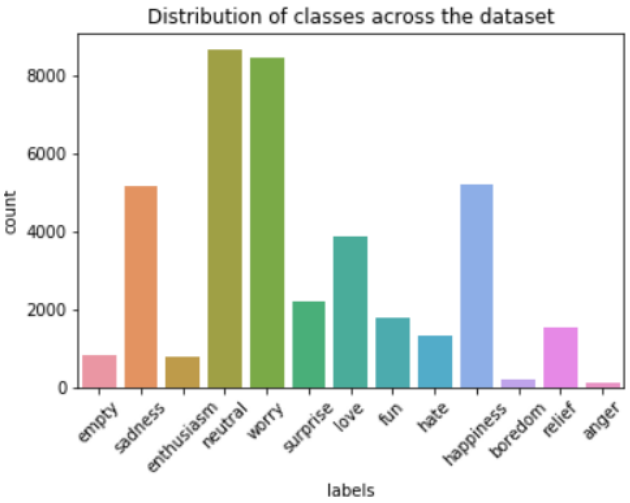

Figure 3: Unbalanced share of classes in the Text-emotion dataset

As from Figure 2, we can see that the training dataset used for emotional analysis is unbalanced.

To perform over-sampling, we have applied Adaptive Synthetic Sampling (ADYSN) which is derived from SMOTE. It finds n-nearest neighbors in the minority class from each pf the samples in the class and then generates random new points on the vector between the two points. After creating a vector of new points, it adds small variation to the points.

Balanced dataset obtained is cleaned using above mentioned steps and the fed into the LSTM model to predict the emotions on the Election data.

TABLE VI. EVALUATION RESULTS FOR EMOTIONAL ANALYSIS USING LSTM

Metrics	Score
Accuracy	0.623
F1 score	0.604
Evaluation loss	1.22

These trained models are used to predict the labels for tweets collected for Indian election.

IV. RESULTS AND DISCUSSION

After training the Multinomial model for sentiment analysis and LSTM models as common model, predicted labels for the tweets were examined. As we can depict from the graph provided in Figure 4, there is rise in positive tweets after new year. As, Indian Election is held in March, Twitter get more engaged with the election Tweets after December 2018. The first phase of Election was in April 2019 and at this period political parties also begin to do campaigns on the Twitter network through tweeting which resulted in high number of positive tweets started to post in comparison to negative tweets against the Indian Election of 2019.

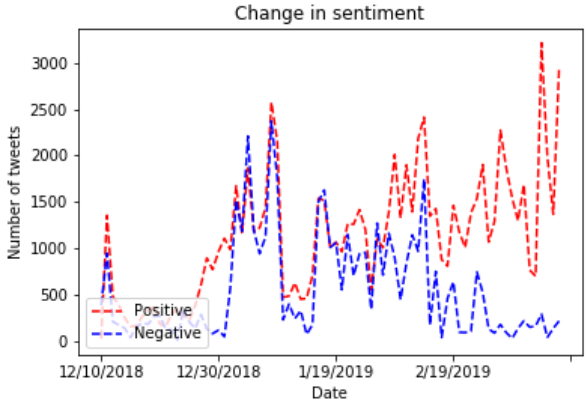

Figure 4: Sentiment of Election Tweets Classified by LSTM

While, the results obtained from Multinomial Naïve Bayes classifier in Figure 5, also shows the dominance of positive tweets over negative ones but the fraction of positive tweets after December 2018 is higher than the prediction using LSTM.

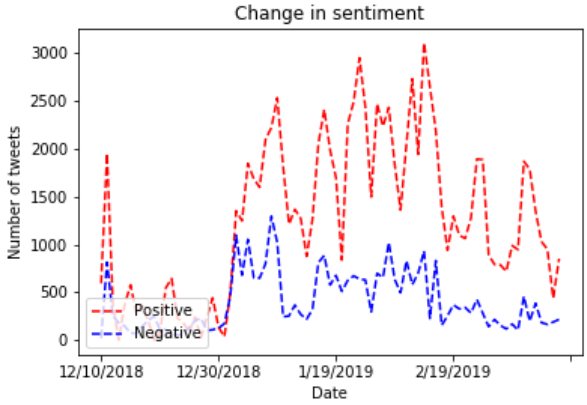

Figure 5: Sentiment of Election Tweets classified by Naive Bayes

Stance Analysis shown in the Figure 6 clearly shows, that people has supported BJP more than Congress on the twitter platform. BJP won the previous election which really affects the Election of 2019. In addition, Congress and BJP have almost equal supporters till start of January 2019 but the number of tweets in favor of BJP and Congress gradually deceases after an increment during January-February 2019. Maximum number of tweets posted in the first week of January 2019 for the Indian election.

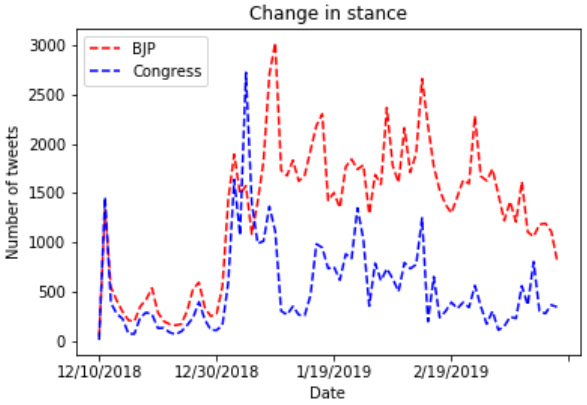

Figure 6: Stance Classification of tweets using LSTM

Emotional Analysis carried out for this study is shown in Figure 7. We have plotted the top 5 emotions observed in the tweets obtained for the Elections. It is clearly seen from the below provide graph that the worry is ascending over the other moods. “Worry” shows a kind of negative feeling as the “Balakot airstrike” occurred in start of 2019.

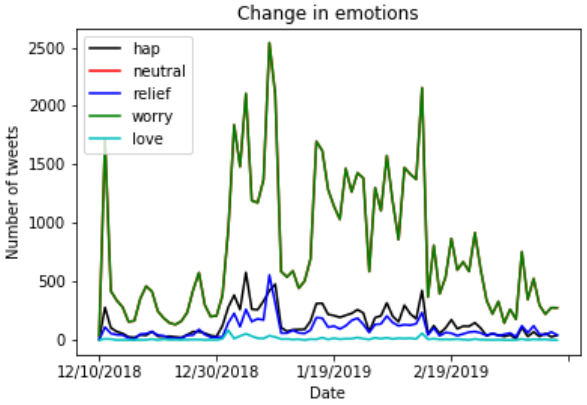

Figure 7: Emotional Analysis of election tweets using LSTM

V. CONCLUSION & FUTURE WORK

For this study, we have collected around 1.5 million tweets related to Indian Election of 2019. These tweets were collected during the period of four months prior to 2 months of Election result. The tweets were filtered by the hashtags mentioned in the post for the two main parties of the Lok Sabha Election i.e. BJP and Congress. We then carried out Stance Analysis to predict the popularity of a party.

This study attempts to observe the nature of political disclosure that took place on Twitter during the time of Indian Election using Machine learning methods like LSTM and Naïve Bayes classifiers. It was observed that the overall sentiment of tweets was positive, and BJP is the dominating party in the Indian Election of 2019.

For future work, this study can be carried out with huge amount of election data with advance machine leaning algorithms and techniques like Bert, Elmo, Fair in order to get more accurate results. The lessons learned from this study can be used to gauge the sentiments of other public discussions in order to understand the flow and authenticity of information provided on social media.

VI. REFERENCES

[1] "Internet World Stats," Miniwatt Marketing Group, 30 June 2019. [Online]. Available: https://www.internetworldstats.com/top20.htm. [Accessed 2019].

[2] "Statistica," Statistica, 18 July 2019. [Online]. Available: https://www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries/. [Accessed 08 August 2019].

[3] Hearst and M. A., "Direction-Based Text Interpretation," 1992.

[4] HUETTNER, Alison, SUBASIC and Pero, "Fuzzy Typing for Document Management," USA, 2000.

[5] Lee and B. P. a. Lillian, "Opinion Mining and Sentiment Analysis," *Foundations and Trends in Information Retrieval*, vol. 2, no. 1-2, pp. 1-135, January 2008.

[6] B. Liu, Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers, 2012.

[7] Boiy, Erik, Hens, Pieter, Deschacht, Koen, Moens and Marie-Francine, "Automatic Sentiment Analysis in Online Text," in *Proceedings of the 11th International Conference on Electronic*, Vienna, June 2007.

[8] S.-M. Kim and E. Hovy, "Determining the Sentiment of Opinions," in *COLING '04 Proceedings of the 20th international conference on Computational Linguistics*, Geneva, Switzerland, 2004.

[9] Bermingham, Smeaton, A. & and Alan, "Classifying sentiment in microblogs: Is brevity an advantage?," in *CIKM 2010 - 19th International Conference on Information and Knowledge Management*, Toronto, 2010.

[10] J. Bollen, H. Mao and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1-8, March 2011.

[11] A. Tumasjan, T. O. Sprenger, P. G. Sandner and I. M. Welpe, "Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment," *Journal Of The International Linguistic Association*, 2010.

[12] B. O'Connor, R. Balasubramanyan, B. R. Routledge and N. A. Smith, "From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series," *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, vol. 11, pp. 122-129, 2010.

[13] G. Gautam and D. Yadav, "Sentiment analysis of twitter data using machine learning approaches and semantic analysis," in *Seventh International Conference on Contemporary*, 2014.

[14] B. Gokulakrishnan, P. Priyanthan, T. Ragavan and N. P. a. A. Perera, "Opinion Mining and Sentiment Analysis on aTwitter Data," in *International Conference on Advances in ICT for Emerging Regions (ICTer2012)*, Colombo, Sri Lanka, 2012.

[15] M. Taboada, J. Brooke, K. V. Milan Tofiloski and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational Linguistics*, vol. 37, no. 2, pp. 267-307, 2011.

[16] X. Chen and M. V. a. K. Madhavan, "Mining Social Media Data for Understanding Students' Learning Experiences," in *IEEE Transactions on Learning Technologies*, 2014.

[17] V. &. K. D. S. Singh, "Opinion mining and analysis: A literature review," in *Proceedings of the 5th International Conference on Confluence 2014: The Next Generation Information Technology Summit*, Noida, 2014.

[18] D. J. and B. Joyce, "Sentiment analysis of tweets for the 2016 US presidential election,," in *2017 IEEE MIT Undergraduate Research Technology Conference (URTC)*, Cambridge, 2017.

[19] K. Jahanbakhsh and Y. Moon, "The predictive power of social media: On the predictability of us presidential elections using twitter," *arXiv preprint arXiv:1407.0622*, 2014.

[20] L. Wang and J. Q. Gan, "Prediction of the 2017 French election based on Twitter data analysis," in *9th Computer Science and Electronic Engineering (CEEC)*, Colchester, 2017.

[21] E. Kouloumpis, T. Wilson and J. Moore, "Twitter Sentiment Analysis:The Good the Bad and the OMG!," in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, 2011.

[22] D. Ramage, S. Dumais and D. Liebling, "Characterizing Microblogs with Topic Models," in *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, 2010.

[23] G. Alec, B. Richa and H. Lei, "Twitter Sentiment Classification using Distant Supervision," Stanford, 2009.

[24] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proceedings of SIGIR'15*, 2015.

[25] Kim and Yoon, "Convolutional Neural Networks for Sentence Classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014.

[26] X. Wang, Y. Liu, C. Sun, B. Wang and X. Wang, "Predicting Polarities of Tweets by Composing Word Embeddings with Long Short-Term Memory," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Beijing, China, 2015.

[27] "http://help.sentiment140.com," [Online]. Available: http://help.sentiment140.com/for-students. [Accessed 2019].

[28] Communication Department of the European Commission, "European Union," europa.eu, [Online]. Available:www.europa.eu/european-union/about-eu/history_en. [Accessed 25 Nov 2018].