Journal of
Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.776539

OPINION MINING AND SNARK ANALYSIS IN GENERAL ELECTION TWEETS +5

Dr. V.V. Jaya Rama Krishnaiah
Associate Professor, Department of Computer Science and Engineering Koneru Lakshmaiah Education Foundation Vaddeswaram, India, jkvemula@gmail.com

Ravi kumar Kuchipudi
research scholar, ANU, Computerscience and Engineering ravikumary4@gmail.com

Abstract. In today's hyper-digital environment, the number of people using social media platforms has reached an all-time high. The vast majority of people participate in community discussions by posting their ideas and experiences on various social media platforms, such as Twitter, Facebook, and YouTube, amongst others. It is highly vital for both the government and the people in business to conduct research and analysis on the feelings and viewpoints of the general population. This is the motivation behind the active participation of a large number of media organizations during the time of the election in the conduct of a variety of different types of opinion surveys. Using the data from Twitter throughout the time period of the 2019 Lok Sabha election, we have worked in this article to conduct an analysis of the feelings of the people of India during the campaign for that election. Because this is an unsupervised learning problem, we have developed an automatic tweet analyzer that uses the Transfer Learning technique. In our Machine Learning model, we made use of the Linear Support Vector Classifiers approach. Additionally, the Term Frequency Inverse Document Fre-quency (TF-IDF) methodology was utilised in order to manage the textual data that was collected from tweets. In addition to this, we have improved the model's ability to deal with the caustic tweets that are made by some of the users, which is something that the researchers that study this field have not yet taken into consideration.
Keywords: Sentiment Analysis, Sarcasm Detection, Linear SVC, TF-IDF, Political Tweets.

1. Introduction

Everything in the present digital age is moving towards an online format, and during the COVID-19 epidemic, this trend has accelerated dramatically. Everything from school to work to medical care to retail to health and wellness to customer satisfaction polls and more has moved online. Twitter, Facebook, YouTube, etc. are just some of the popular online social networking sites individuals use today to voice their opinions and connect with others. People frequently use these venues to talk about the problems they've encountered or the incidents they've witnessed. When compared to traditional methods of opinion dissemination, the use of such platforms allows the user to reach a far wider audience. Many different types of groups put a premium on analysing public opinion; whether they be political parties, businesses, investors, or even individual professionals, they all want to know what the masses think.

Many political parties, especially during election season, make an effort to gauge the prevailing public opinion. The opinions and feelings of the general public are often studied through polls and surveys conducted by news organisations and other media outlets. However, during the COVID-19 epidemic, it may not always be able to physically reach out to someone [1]. Sentiment analysis of political tweets is highly important because elections are a periodic activity that will occur periodically even during the pandemic time, thereby reducing the risk of the field workers of the polling firm for performing survey polls. Using sentiment analysis techniques and accounting for sarcastic tweets—a factor not yet taken into account in the state-of-the-art research in this domain—we extract the political feelings of the people using tweets from the Lok Sabha Elections - 2019. Tweets from the data science site Kaggle [2] are used in both the training and testing phases of the 2019 election. This article analyses the official election-related Twitter accounts of the major political parties. To deal with the problem's lack of supervision, it used a Transfer learning technique. The following are the paper's most significant contributions:

- Developing a robust sentiment analysis model for election data.
- Making the model capable enough to handle the sarcastic tweets.
- Implementing the transfer learning approach for handling the unsupervised nature of the problem.
- Analysing the results of the model with respect to the actual election results.

This paper is organized as follows. The paper is introduced in Section 1. In the second section, we examine previous research on predicting election outcomes and conducting sentiment analysis. Material and methods, including recommended methodologies for dealing with sentiment analysis and sarcasm detection of textual data, are presented in Section 3. The sarcasm detection and sentiment analysis model is evaluated experimentally in Section 4. Future research is discussed in the paper's last section (Section 6).

2. Related Works

Sentiment analysis of Twitter data is not new, nor is it restricted to the realm of political posts. An opinion mining and textual analysis technique was created by Pang and Lee [3] in 2008. Jhanwar and Das [4] suggested a method for analysing Indians' feelings by examining writings written in Hindi and English. To determine whether a tweet is favourable or negative regarding a certain topic, Go et al. [5] attempted to train a sentiment analysis algorithm on the use of emoticons. Pak and Paroubek [6] worked in the same area, attempting to increase functionality via a wider variety of subjective and objective terms that can be used to build classifiers. These classifiers can be used to gather information that can be utilised to control the tone of a tweet. VADER [7] is a free and open-source programme that was first presented in [7]. This programme follows a set of rules to calculate emotional weightings for textual information. Tweets about COVID-19 were analysed by Zhang et al. [8] in four Canadian and four American cities. Using the VADER [7] and NRC [9] approaches, we analyse the sentiment intensity ratings and visualise the data from the pandemic. Many methods were tried by Das and Bandyopadhyay [10] before they were able to draw conclusions on the emotions conveyed in tweets. They tried out several new methods, such a gamified dictionary, a bilingual wordbook, and the WordNet database. To expand their dictionaries, authors in the literature [11, 12] did what WordNet [13] does: they started with a small set of opinion terms and searched over a huge corpus of writings. Taboada et al. [14] also presented a lexicon-based method for sentiment analysis. The tweets were analysed using a lexicon of visualised positive and negative phrases. Bhadane et al. [15] surveyed a variety of techniques for analysing textual data in natural languages according to the moods expressed in that data, i.e. whether the text is upbeat or downbeat.

Many interested parties have focused their attention on political tweets. Multiple writers have made contributions to the field of election and political tweet sentiment analysis. In [16], Hamling and Agrawal examine the tweets on the 2016 US election. The election results were compared to the sentiments expressed in these tweets. The sentiment analysis was performed using the Sentiwordnet database [17]. The values for positive words were between +0.0625 and +1.0, whereas the values for negative terms were between -0.0625 and -1.0. The sentiment algorithm assigns a value of +0.125 to a positively connoting keyword such as "helpful," and a value of -0.25 to a negatively connoting keyword such as "unhappy." Since sarcasm is also a widespread form of expressing opinion, the importance of sarcasm recognition cannot be underestimated in this field, and their algorithm was unable to manage the sarcastic tweets done by some of the users. In [18], the authors attempted to compute a distance metric to extract people's favourable and negative feelings towards the most popular political parties. It represents the relative proximity of tweets. A tighter clustering of tweets in favour of competing political parties is indicative of a more heated campaign.

Many studies have shown that conducting sentiment analysis of Twitter data can accurately forecast election outcomes; for example, the lexical method was used to accurately predict the winners in the 2015 Swedish elections [19]. Jose and Chooralil [20] have devised a novel technique for collecting data, which makes use of Twitter's streaming API. They used lexical resources like WordNet [13] and SentiWordNet [10] to try to glean information and emotions from the tweets. They also developed a technique for dealing with negation in the pre-processing stage of data to boost their productivity.

Researchers have been curious about politicians' and political campaigns' use of Twitter. Interest in Twitter's significance in political struggles has been stoked by Barack Obama's campaign tactics in the 2008 U.S. elections [21][22]. Many members of the United States Congress have taken to Twitter, where they express their views on national politics and local concerns in their respective districts [22, 24].

Sharmistha Chatterjee [25] performed a sentiment analysis of the two major parties, BJP and INC, by scraping data from Twitter via API. She classified the two major parties' emotions using standard ML and Deep learning techniques. For a few months, she crawled and combined the tweets once a week. For this Sentiment Representation, we employed the WordCloud and N-gram Model [26]. Together with the retweet frequency distribution, she implemented an extra location mapping feature for the tweets. Using sentiment analysis on Hindi Twitter data,

Sharma and Moh [27] tried to foretell the outcome of the Indian election. The number of Hindi tweets they were able to glean is 42,345. The unsupervised problem was converted into a supervised one by doing data cleaning to remove unnecessary tweets, ultimately leaving 36,465 tweets. The Twitter data was then put through an I Bayes, support vector machine.
General Elections-2019 in India were also forecasted by Gaikar and Sapare [28], who employed the LSTM Neural Network method. To train their model, they used over 1500 tweets annotated with sentiment categories such as positive, negative, and neutral. To put their model to the test, they used the Twitter API to pull a total of 40,000 tweets from January 2019 through March 2019 that discussed elections. Word clouds were used to visualise the data, and the findings were compared to those of the ABP-C and the India Today Survey. Classification of tweets about India's General Elections-2019 was also conducted by Ansari et al. [29], who used the long short-term memory (LSTM) model to do so. They analysed tweets for their propensity to guess election outcomes using the categorization model. Regarding the Indian election, Sharma and Ghose [30] used named entity identification in text mining to weed out irrelevant tweets. They employed the Rapid-Miner AYLIEN [31] model to analyse the sentiment of relevant tweets. To conduct their analysis of tweets about India's General Election-2019, Naiknaware and Ka-wathekarm [32] turned to the Sentiment analysis score method built into the R computer language. To determine the overarching mood of a political event from tweets, Bose et al. [33] applied the NRC emotion lexicon method. The tweets were then put via the deep learning application ParallelDots, which can sort them into good, negative, and neutral. Katta and Hegde [34] present an approach called the Adaptive Neuro-Fuzzy Inference System (ANFIS), which uses Non-Linear SVM classifier analysis to enhance fuzzy principles and create a Fuzzy-based ontology. They determined that an ANFIS Non-linear SVM-based model is simpler and more accurate than other approaches to sentiment analysis of social media text. In [35], Bansal and Sri-vastava employed an emoji and n-gram feature-based Lexicon-based method to Twitter sentiment analysis to estimate vote shares. Hitesh et al. [36] used Word2vec and the Random Forest Model to conduct a real-time sentiment analysis of the 2019 Indian General Elections. To foretell the results of the 2019 Indian general election, Joseph [37] employed the Decision Tree classifier method. He thought about English as a series of tweets. His plan was to track voters' emotions in real time during the electoral process. Kristiyanti et al. [38] used the Support Vector Machine (SVM) with selection characteristics of Particle Swarm Optimisation (PSO) and Genetic Algorithm to predict the outcomes of Indonesia's elections. (GA). Predictions for the Indonesian presidency and vice presidency were attempted. The deep learning approach, including multiple algorithms including Convolutional Neural Networks, was also used by Hidayatullah et al. [39] to forecast the outcomes of Indonesia's elections.

Network (CNN), Long short-term memory (LSTM), CNN-LSTM, Gated Recurrent Unit (GRU)-LSTM, and Bidirectional LSTM. They compared the results with various traditional machine learning algorithms and concluded that the Bidirectional LSTM achieved the best accuracy.

Opinion and sentiment mining employ a wide variety of approaches, however they can be divided roughly into two classes. The first makes use of machine learning techniques, while the second takes a more linguistic approach through the use of a lexicon [40]. In this study, we employ a Machine Learning strategy for gathering people's thoughts. Supervised learning and unsupervised learning are two major categories of machine learning algorithms. To construct a machine learning model, the supervised learning method necessitates labelled data from the specific domain of interest. When there isn't any labelled data available to use in training the model, however, the unsupervised method comes into play. Since the tweets about the 2019 Loksabha election do not have any assigned labels, sentiment analysis falls under the umbrella of unsupervised learning. Because we did not have access to the ideal labelled dataset, we resorted to transfer learning to address this problem. Unsupervised problem-solving techniques like transfer learning exist. The idea behind this method is to use a dataset that has already been labelled for one problem to train a model that can be used to make predictions on an unlabeled dataset. There have been a number of works on sarcasm detection [41], but none in the arena of political tweets until now, which is where this study comes in. It uses sarcasm recognition technology to examine tweets about political campaigns.

3 Material and Methods

3.1 Dataset

The testing of our trained model is done using the India Lok Sabha Elections-2019 tweets. During elections, people used to express their views, opinions, and experiences related to major political parties of that time i.e. BJP and INC. The complete dataset of the election-related tweets is available on the data science platform Kaggle [2]. The dataset consists of the following fields.

Last_updated: This column contains the information about the time stamp at which the particular tweet was last updated.

Tweet_id: The unique id which is assigned to every tweet.

Created_at: This column contains the timestamp at which the tweet was created.

Full_text: The column contains the complete tweet text on which we will perform the text analysis.

Quote_count: This column contains the frequency the current tweet was retweeted with a comment.

Reply_count: This column contains the frequency the current tweet was replied to or commented on by any user.

Retweet_count: This column contains the frequency the current tweet was retweeted.

Favorite_count: This column contains the frequency of likes the current tweet is hav-ing.

In this paper, we have considered only the column with textual data for the sentiment analysis process,i.e. 'full_text' column.

3.2 Machine Learning Algorithms

TF-IDF. Term Frequency – Inverse Document Frequency (TF-IDF) [42] is a matrix that provides the word frequency table in a document/ sentence. Term frequency (TF(t,d)) is the measure for the word occurring in a document(d), whereas Document Frequency(DF(t)) is counter for the number of documents the word is occurring. In-verse Document Frequency (IDF(t)) gives the relative weight for a word. If the word is occurring often, its IDF(t) measure will be low whereas the IDF(t) will be high for less occurring words. Hence, it can be mathematically defined as Eq. (1) [43][44].

$$IDF(t) = \frac{N}{DF(t)} \quad \text{……………(1)}$$

where, N is Number of documents, DF(t) is Document Frequency and IDF(t) is Inverse Document Frequency.

In Eq. (1), N can be a large value hence it will explode the value of IDF(t) or DF(t) can possibly be 0 during query time. Hence the log function is taken to control the former once and the latter one is resolved by adding 1. Then the new equation for IDF(t) will be as Eq. (2) [43][44].

$$IDF(t) = \log \frac{N}{DF(t)+1} \quad \text{……..(2)}$$

Finally from Eq. 2, TF-IDF can be mathematically expressed as Eq. (3) [43][44].

$$TF - IDF(t,d) = TF(t,d) * \log \frac{N}{DF(t)+1} \quad \text{……..(3)}$$

Linear SVC. Support Vector Machine (SVM), a machine learning algorithm that can perform classification, regression, and outliers identification. Linear Support Vector Classifier (Linear SVC) is a classifying algorithm that gives out the best fitting line or hyperplane depending upon the dimensions of the problem. Dimension refers to the number of features. SVC is chosen because it ignores all the outliners and only chooses the best hyperplane to distinguish between the classes. Fig. 1 shows a Linear SVC ex-ample in two-dimensional space [45].

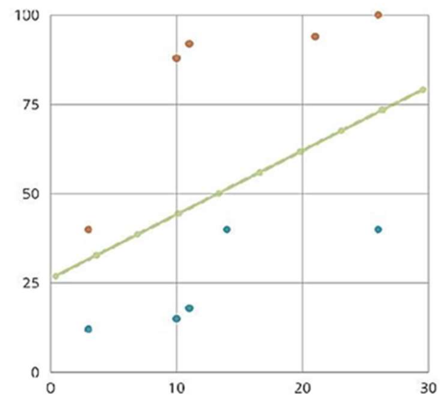

Fig. 1. Linear SVC example in 2-D space

3.3 Proposed Model

In this research, we attempt to address the unsupervised character of sentiment analysis by using transfer learning to the problem of tweets about the 2019 elections. We have trained our model with Linear SVC and then performed sentiment analysis on the standard Twitter review dataset that is publicly available on Kaggle [2]. Since the data was in textual form, we processed it using the TFIDF technique, which involves constructing a Term Frequency Inverse Document Frequency (TFIDF) matrix. When compared to a previous study on the 2016 US Presidential Elections, in which sarcasm was not addressed, this one is an improvement. Figure 2 and Figure 3 show the process of training a sentiment analysis and sarcasm detection model, and then putting that model to the test on the election dataset..

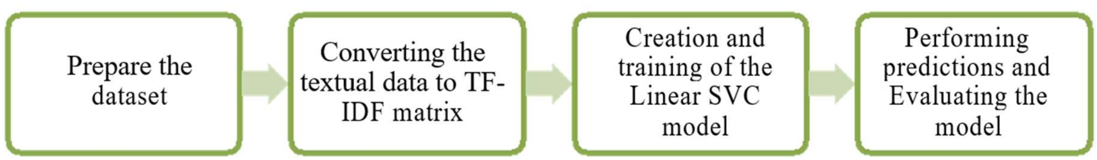

Fig. 2. The flow of work for the training of sentiment analysis and sarcasm detection model

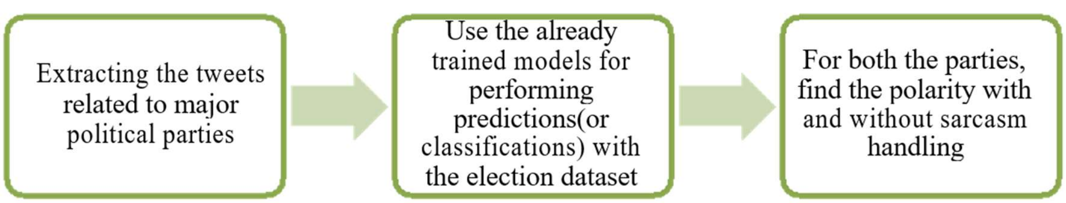

Fig. 3. The flow of work for the testing of the trained models on the election's dataset

Algorithms for training the sentiment analysis model, training of the sarcasm detection model, and transfer learning approach for sentiment analysis and sarcasm detection of the election's dataset are presented in 1, 2, and 3, respectively.

We have introduced an extra feature of sarcasm detection to the existing methodol-ogies for performing sentiment analysis of political tweets. Since the nature of the in-coming data makes the problem an unsupervised problem, i.e.it is not known before-hand which tweet is sarcastic and which one is not, we applied the Transfer learning approach to resolve this problem. For the training of the sarcasm detection model, a standard sarcasm detection dataset available on Kaggle is used, and then the trained model can be used to predict results on the actual tweets.

Algorithm 1: Procedure for the training of the sentiment analysis model.

1. Split the dataset into training and the testing data by the ratio of 70:30, setting the random state to be 42.
2. Use the TFIDFVectorizer of the sklearn library, transforming the textual data into the TFIDF matrix.
3. Create a pipeline passing the 2 phases for it as TFIDF and the LinearSVC model.
4. Fit the newly created pipeline with the training dataset along with the labels.
5. Perform the predictions using the testing data.
6. Compare the predicted results with the actual results by using the confusion_matrix and the classification_report which provides the accuracy score for the model.

Algorithm 2: Procedure for the training of the sarcasm detection model.

1. Use the TFIDFVectorizer of the sklearn library, transforming the textual data into the TFIDF matrix.
2. Create a pipeline passing the 2 phases for it as TFIDF and the LinearSVC model.
3. Fit the newly created pipeline with the training dataset along with the labels.
4. Perform the predictions using the testing data.
5. Compare the predicted results with the actual results using the confusion_matrix and the classification_report provides the accuracy score for the model.

Algorithm 3: Procedure for using Transfer learning approach for sentiment analysis and sarcasm detection.

1. Use the already created sentiment analysis and sarcasm detection models for per-forming predictions (or classifications) with the election dataset.
2. Add two new columns with the prediction results for both the models, i.e. one col-umn having the sentiment result, whether positive or negative, while the second column holds the result for whether the tweet is sarcastic or not.
3. For both the parties, find the positive and negative popularity with and without sar-casm detection and display the results of both.
4. Plot the results for both the parties as bar graphs and pie charts.

4 Experiment Evaluation

4.1 System Configuration and Programing Environment

Operating System Windows 10 is used for the experiments, together with 8 GB of RAM and a 2 TB hard drive on a computer powered by an Intel Core i5 processor. Python 3.7 and Jupyter

Notebook 6.4.3 were used for the actual development and implementation. The following is a list of the additional libraries and tools used. Version 0.22 of Sklearn: Free and open-source machine learning techniques for Regressions, Classifications, and Clustering can be found in the Scikit-learn Python module..

Matplotlib(Version 3.4.3): MatplotLib is a plotting library in Python which provides the API for plotting embedded plots for the data passed to it.

Pandas(Version 1.3.2): Pandas is a data table handling library in Python which provides the methods to perform all the operations on the data tables conveniently.

Pipelines (Version 1.0): Pipelines in python provide a sequence of transformation and prediction on the data as per the machine learning models passed to it.

json library: The json library can parse JSON from strings or files. The library parses JSON Python dictionary or list. It can also convert Python dictionaries or lists into JSON strings.

4.2 Results and Discussions

After the model has been trained successfully, we have utilised the sklearn Python library's confusion_matrix and classification_report methods to evaluate the model's accuracy and classification quality. The quality of the classifications made by the algorithms can be determined using tables called classification reports and confusion matrices, respectively. Tables 1 and 2 show the outcomes of the sentiment analysis model..

Table 1. Classification Report of Sentiment Analysis Model

Type	Precision	Recall	F1-score	Support
Negative	0.80	0.79	0.79	239819
Positive	0.79	0.80	0.80	240181
Accuracy			0.80	480000
Macro Accuracy	0.80	0.80	0.80	480000
Weighted Average	0.80	0.80	0.80	480000

Table 2. Classification Report of Sarcasm Detection Model

Type	Precision	Recall	F1-score	Support
0	0.85	086	0.86	4498
1	0.82	0.81	0.82	3515
Accuracy			0.80	8013
Macro Accuracy	0.80	0.80	0.80	8013

Weighted Average	0.80	0.80	0.80	8013

Our sentiment analysis model achieves an accuracy of 80%, as shown in Table 1. This makes it suitable for transfer learning. According to Table 2, the sarcasm detection model is transfer learning-ready since it obtains an accuracy of 84%.

We have analysed election-related tweets using the aforementioned trained models, both for and against sarcasm. Figures 4–6 and Figures 7–9 display the outcomes for the BJP and INC in the two scenarios, without and with sarcasm, respectively.

Terms like "positive tweets" and "positive polarity tweets" denote that the tweets in question express favourable feelings towards a specific political party. Similarly, words like "negative tweets" or "negative polarity tweets" denote tweets that express disapproval or hostility against a particular political party.

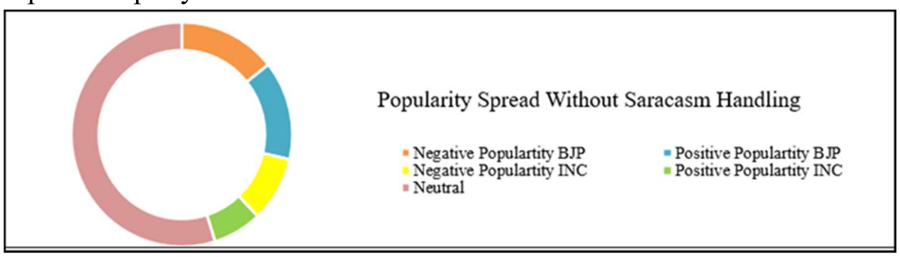

Fig. 4. Popularity spread of tweets without sarcasm handling

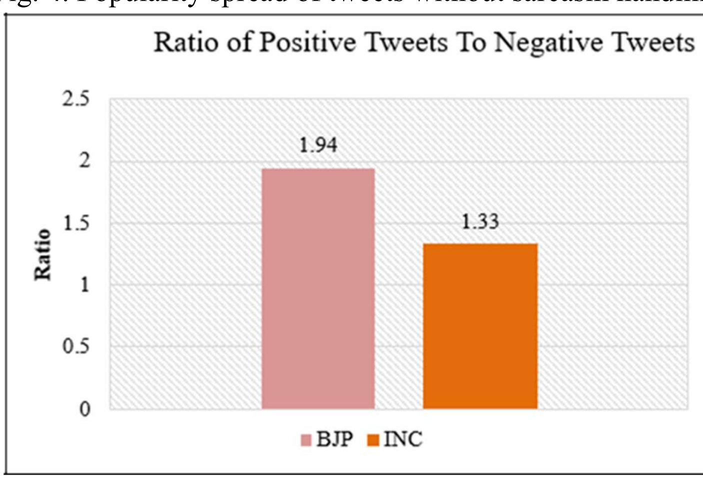

Fig. 5. Ratio of Positive to Negative tweets without sarcasm handling

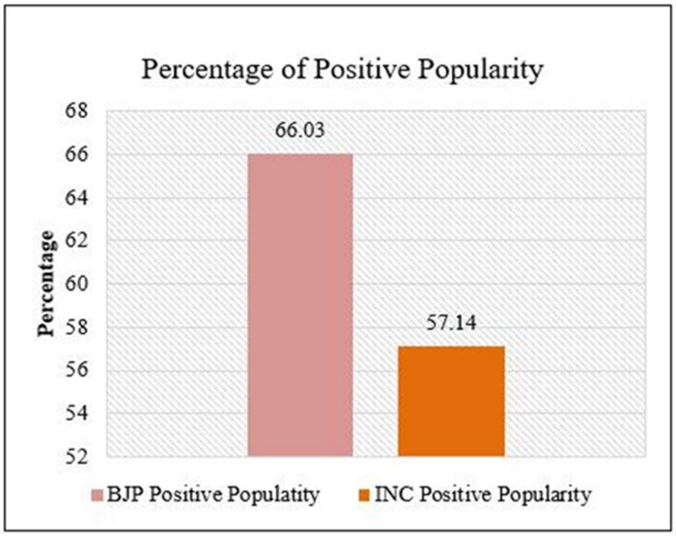

Fig. 6. Percentage of positive tweets among total tweets without sarcasm handling

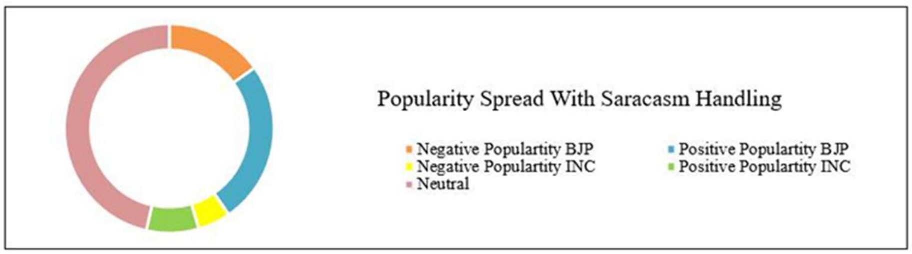

Fig. 7. Popularity spread of tweets with sarcasm handling

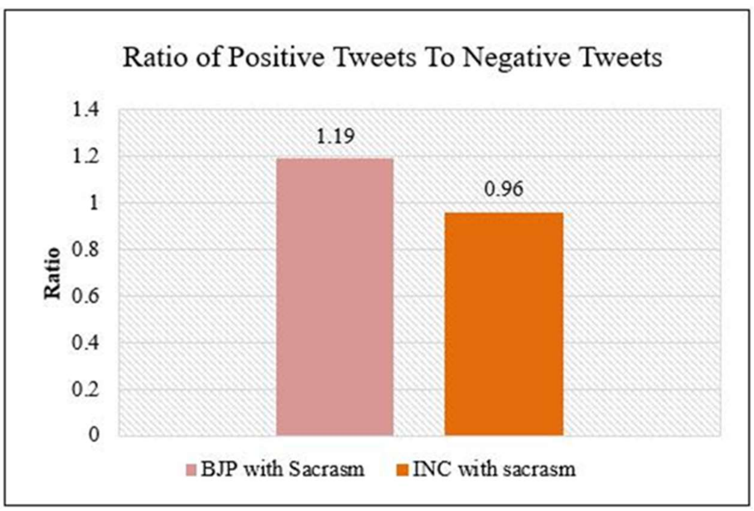

Fig. 8. The ratio of Positive to Negative tweets with sarcasm handling

Fig. 4 displays that when all tweets were analyzed, only 25.58% had a positive polarity towards the Bharatiya Janata Party. (BJP). Similarly, 13.16 percent of all tweets had a negative polarity towards the BJP. Figure 7 displays the outcomes after sarcasm was taken into account, showing that 23.76 percent of all tweets were favourable in polarity for BJP and 20.01 percent were negative.

Similarly, in Fig. 4, we can see that for the second largest party, the Indian National Congress, 6.50% of the total tweets were of positive polarity and 4.88% of the total tweets were of negative po-larity when we evaluated the tweets without treating sarcasm. Figure 7 shows that after the sarcasm was removed using the transfer learning method, only 6.34 percent of all tweets about the Indian National Congress (INC) were favorable, while 6.64 percent were negative. Table 3 depicts the aforementioned findings.

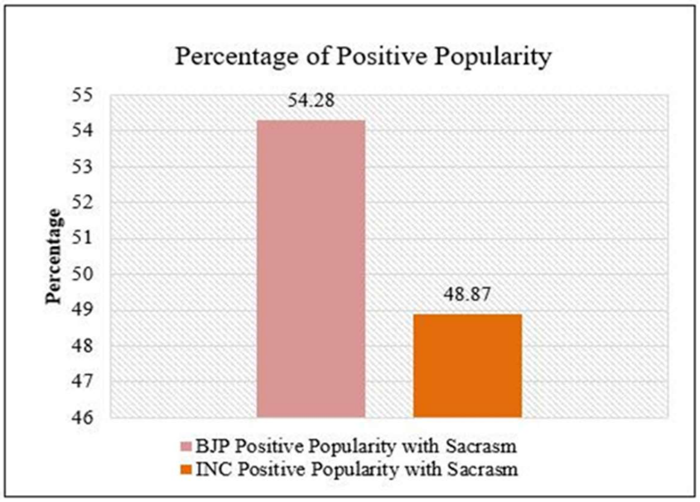

Fig. 9. Percentage of Positive tweets among total tweets with sarcasm handling

Table 3. Polarity Percentage of tweets with respect to total tweets

Party	Positive Polarity of tweets (without sar-casm handling)	Positive Polarity of tweets (with sarcasm handling)	Negative Polarity of tweets (without sarcasm handling)	Negative Polarity of tweets (with sar-casm handling)
BJP	25.58%	23.76%	13.16%	20.01%
INC	6.50%	6.34%	4.88%	6.64%

We charted the ratio of good to negative tweets about both parties to help make the findings clearer. Fig. 5 and Fig. 8 depict the graphs for the examples without and with sarcastic handling, respectively. We found that when we used our trained model to account for sarcasm, the ratio of positive to negative tweets for the BJP party increased to 1.94 from 1.33. For the BJP, it rose to 1.19, while the INC saw its support fall to 0.96. Table 4 displays the results of all ratio calculations.

Table 4. Positive: Negative polarity ratio of tweets

Party	Positive: Negative (without Sarcasm Han-dling)	Positive: Negative (with Sarcasm Han-dling)
BJP	1.94	1.19
INC	1.33	0.96

We then made a second bar chart, this time showing the proportion of good to negative tweets about each political party. Figure 6 (without sarcasm treatment) and Figure 9 show graphs for this value. (with sarcasm handling). Without accounting for sarcasm, we observed that 66.03% of all tweets regarding the BJP party and its leaders had a positive polarity, while 57.14% of all tweets about the INC party had a positive polarity. However, when we attempted to account for irony, we found that 54.28 percent of all tweets about the BJP party were positive in polarity, while 48.87 percent of all tweets about the INC party were negative. Table 5 provides a summary of the aforementioned findings.

Table 5. Percentage of positive tweets for a political party

Party	Percentage (without Sarcasm handling)	Percentage (with sarcasm handling)
BJP	66.03%	54.28%
INC	57.14%	48.87%

5 Conclusion and Future Works

This research proposed the use of transfer learning to deal with sarcastic tweets by analysing them to determine if they were positive or negative. Our trained models have been found to perform admirably. When we compared the results from our model to the actual election results, we found that 37.4% of the national vote went to the BJP and 19.5% went to the INC. Additionally, our model indicated that the BJP party will win the election with a vote share difference of almost 19%, which is roughly in line with the results of the 2019 elections.

Our model's predicted difference in votes from the actual election results is very close to what really happened. Although the actual vote percentage for any party differs from our model's forecasts, the following points should be taken into account for a diverse country like India. The fact that certain people in a given area do not have access to the Internet is known as the "digital divide." Not everybody who has access to the Internet also utilises Twit-ter. It's also possible that there were inconsistencies introduced during data collection. In addition, for both political and personal reasons, many members of society choose to refrain from airing their dirty laundry in public.

The following are potential areas for future expansion of the project. To help the country progress, it can classify unfavourable tweets into relevant categories like agriculture, education, infrastructure, and price increases. Tweets containing hate speech or abusive language can be filtered out, leaving only those containing constructive comments. In addition, a more effective unsupervised learning method can be used to upgrade the model as an alternative to the transfer learning strategy.

References

1. Mohamed Ridhwan, K., Hargreaves, C.A.: Leveraging Twitter data to understand public sentiment for the COVID-19 outbreak in Singapore. Int. J. Inf. Manag. Data Insights. 1, 100021 (2021). https://doi.org/10.1016/J.JJIMEI.2021.100021.
2. Indian Political Tweets 2019 (Feb to May) | Kaggle.
3. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. Found. Trends® Inf. Retr. 2, 1–135 (2008). https://doi.org/10.1561/1500000011.

4. Jhanwar, M.G., Das, A.: An Ensemble Model for Sentiment Analysis of Hindi-English Code-Mixed Data. (2018).
5. Go, A., Bhayani, R., Huang, L.: Twitter Sentiment Classification using Distant Supervision.
6. Pak, A., Paroubek, P.: Twitter as a Corpus for Sentiment Analysis and Opinion Mining. undefined. (2010).

7. Hutto, C., Gilbert, E.: VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Proc. Int. AAAI Conf. Web Soc. Media. 8, 216–225 (2014).
8. Zhang, Q., Yi, G.Y., Chen, L.-P., He, W.: Text mining and sentiment analysis of COVID-19 tweets. (2021).
9. Mohammad, S.M., Kiritchenko, S., Zhu, X.: NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. *SEM 2013 - 2nd Jt. Conf. Lex. Comput. Semant. 2, 321–327 (2013).
10. Das, A., Bandyopadhyay, S.: SentiWordNet for Indian Languages. 21–22 (2010).
11. Kim, S.-M., Hovy, E.: Determining the sentiment of opinions. 1367-es (2004). https://doi.org/10.3115/1220355.1220555.
12. Emotions in social psychology: Essential readings. - PsycNET.
13. WordNet | A Lexical Database for English.
14. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-Based Methods forSentiment Analysis. Comput. Linguist. 37, 267–307 (2011). https://doi.org/10.1162/COLI_A_00049.
15. Bhadane, C., Dalal, H., Doshi, H.: Sentiment Analysis: Measuring Opinions. Procedia Comput. Sci. 45, 808–814 (2015). https://doi.org/10.1016/J.PROCS.2015.03.159.
16. Hamling, T., Agrawal, A.: Sentiment Analysis of Tweets to Gain Insights into the 2016 US Election.
17. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining, (2010).
18. Tumasjan, A., Sprenger, T.O., Sandner, P.G., Welpe, I.M.: Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. (2010).
19. Liu, B.: Sentiment Analysis and Opinion Mining. http://dx.doi.org/ 10.2200/S00416ED1V01Y201204HLT016 . 5, 1–184 (2012). https://doi.org/10.2200/S00416ED1V01Y201204HLT016.
20. Jose, R., Chooralil, V.S.: Prediction of election result by enhanced sentiment analysis on Twitter data using Word Sense Disambiguation. 2015 Int. Conf. Control. Commun. Comput. India, ICCC 2015. 638–641 (2016). https://doi.org/10.1109/ICCC.2015.7432974.
21. Abroms, L.C., Lefebvre, R.C.: Obama's Wired Campaign: Lessons for Public Health Communication. http://dx.doi.org/10.1080/10810730903033000. 14, 415–423 (2009). https://doi.org/10.1080/10810730903033000.
22. Jarvis, S.E.: Communicator-in-Chief: How Barack Obama Used New Media Technology to Win the White House – Edited by John Allen Hendricks and Robert Denton, Jr. Pres. Stud. Q. 40, 800–802 (2010). https://doi.org/10.1111/J.1741-5705.2010.03815.X.

23. Glassman, M.E., Straus, J.R.: CRS Report for Congress Social Networking and ConstituentCommunication: Member Use of Twitter During a Two-Week Period in the 111 th Congress. (2009).
24. Golbeck, J., Grimes, J.M., Rogers, A.: Twitter use by the U.S. Congress. J. Am. Soc. Inf. Sci. Technol. 61, 1612–1621 (2010). https://doi.org/10.1002/ASI.21344.
25. Twitter Sentiment Analysis for the 2019 Lok Sabha Elections | Hacker Noon.
26. Cavnar, W.B., Trenkle, J.M.: N-Gram-Based Text Categorization.

27. Sharma, P., Moh, T.S.: Prediction of Indian election using sentiment analysis on Hindi Twitter. Proc. - 2016 IEEE Int. Conf. Big Data, Big Data 2016. 1966–1971 (2016). https://doi.org/10.1109/BIGDATA.2016.7840818.
28. Gaikar, D., Sapare, G., Vishwakarma, A., Parkar, A., Professor, A.: Twitter Sentimental Analysis for Predicting Election Result using LSTM Neural Network. Int. Res. J. Eng. Technol. (2019).
29. Ansari, M.Z., Aziz, M.B., Siddiqui, M.O., Mehra, H., Singh, K.P.: Analysis of Political Sentiment Orientations on Twitter. Procedia Comput. Sci. 167, 1821–1828 (2020). https://doi.org/10.1016/J.PROCS.2020.03.201.
30. Sharma, A., Ghose, U.: Sentimental Analysis of Twitter Data with respect to General Elections i India. Procedia Comput. Sci. 173, 325–334 (2020). https://doi.org/10.1016/J.PROCS.2020.06.038.
31. Das, S., Kolya, A.K.: Sense GST: Text mining & sentiment analysis of GST tweets by Naive Bayes algorithm. Proc. - 2017 3rd IEEE Int. Conf. Res. Comput. Intell. Commun. Networks, ICRCICN 2017. 2017-Decem, 239–244 (2017). https://doi.org/10.1109/ICRCICN.2017.8234513.
32. Naiknaware, B.R., Kawathekar, S.S.: Prediction of 2019 Indian Election using sentiment analysis. Proc. Int. Conf. I-SMAC (IoT Soc. Mobile, Anal. Cloud), I-SMAC 2018. 660– 665 (2019). https://doi.org/10.1109/I-SMAC.2018.8653602.
33. Bose, R., Dey, R.K., Roy, S., Sarddar, D.: Analyzing Political Sentiment Using Twitter Data. Smart Innov. Syst. Technol. 107, 427–436 (2019). https://doi.org/10.1007/978-981-13-1747-7_41.
34. Katta, P., Hegde, N.P.: A Hybrid Adaptive Neuro-Fuzzy Interface and Support Vector Machine Based Sentiment Analysis on Political Twitter Data. Int. J. Intell. Eng. Syst. 12, (2019). https://doi.org/10.22266/ijies2019.0228.17.
35. Bansal, B., Srivastava, S.: Lexicon-based Twitter sentiment analysis for vote share prediction using emoji and N-gram features. Int. J. Web Based Communities. 15, 85–99 (2019). https://doi.org/10.1504/IJWBC.2019.098693.
36. Hitesh, M.S.R., Vaibhav, V., Kalki, Y.J.A., Kamtam, S.H., Kumari, S.: Real-time sentiment analysis of 2019 election tweets using word2vec and random forest model. 2019 2nd Int. Conf. Intell. Commun. Comput. Tech. ICCT 2019. 146–151 (2019). https://doi.org/10.1109/ICCT46177.2019.8969049.
37. Joseph, F.J.J.: Twitter Based Outcome Predictions of 2019 Indian General Elections Using Decision Tree. Proc. 2019 4th Int. Conf. Inf. Technol. Encompassing Intell. Technol.

Innov. Towar. New Era Hum. Life, InCIT 2019. 50–53 (2019). https://doi.org/10.1109/INCIT.2019.8911975.

38. Kristiyanti, D.A., Normah, Umam, A.H.: Prediction of Indonesia presidential election results for the 2019-2024 period using twitter sentiment analysis. Proc. 2019 5th Int. Conf. New Media Stud. CONMEDIA 2019. 36–42 (2019). https://doi.org/10.1109/CONMEDIA46929.2019.8981823.

39. Hidayatullah, A.F., Cahyaningtyas, S., Hakim, A.M.: Sentiment Analysis on Twitter using Neural Network: Indonesian Presidential Election 2019 Dataset. IOP Conf. Ser. Mater. Sci. Eng. 1077, 012001 (2021). https://doi.org/10.1088/1757-899X/1077/1/012001.

40. Mehta, P., Pandya, S.: A Review On Sentiment Analysis Methodologies, Practices And Applications. Int. J. Sci. Technol. Res. 9, 2 (2020).

41. JoshiAditya, BhattacharyyaPushpak, J., C.: Automatic Sarcasm Detection. ACM Comput. Surv. 50, (2017). https://doi.org/10.1145/3124420.

42. Das Sarit Chakraborty Student Member, B., Member, I.: An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation.

43. Tf-idf :: A Single-Page Tutorial - Information Retrieval and Text Mining.

44. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Inf. Process. Manag. 24, 513–523 (1988). https://doi.org/10.1016/0306-4573(88)90021-0.

45. sklearn.svm.LinearSVC — scikit-learn 0.24.2 documentation.