

# Election Results Prediction Using Twitter Data

## *A Project Report*

*Submitted in partial fulfilment of the  
Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**By**

**1602-19-737-111- B. Sri Kanishka Reddy**

**1602-19-737-308- A. Krishna Chaitanya**

*Under the guidance of*

**Dr. M. Neelakantappa**

**Associate Professor**



**Department of Information Technology  
Vasavi College of Engineering (Autonomous)**

***ACCREDITED BY NAAC WITH 'A++' GRADE***

**(Affiliated to Osmania University)**

**Ibrahimbagh, Hyderabad-31**

**2023**

# **Vasavi College of Engineering (Autonomous)**

*ACCREDITED BY NAAC WITH 'A++' GRADE*

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**



## **DECLARATION BY THE CANDIDATE**

We, **B.Sri Kanishka Reddy** and **A.Krishna Chaitanya** bearing hall ticket numbers, **1602-19-737-111** and **1602-19-737-308** hereby declare that the project report entitled “**Election Results Prediction using Twitter Data**” under the guidance of **Dr.M.Neelakantappa**, Professor and HOD, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering in Information Technology**.

This is a record of bonafide work carried out by us and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

**B.Sri Kanishka Reddy**  
**1602-19-737-111**

**A.Krishna Chaitanya**  
**1602-19-737-308**

**Vasavi College of Engineering(Autonomous)**

***ACCREDITED BY NAAC WITH 'A++' GRADE***

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**



**BONAFIDE CERTIFICATE**

This is to certify that the project entitled “**Election Results Prediction using Twitter Data**” being submitted **B.Sri Kanishka Reddy** and **A.Krishna Chaitanya** bearing hall ticket numbers, **1602-19-737-111** and **1602-19-737-308** in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Information Technology is a record of bonafide work carried out by them under my guidance.

**Dr.M.Neelakantappa**

Associate Professor

**Internal Guide**

**Dr.K. Ram Mohan Rao**

Professor,

**HOD, IT**

**External Examiner**



**NCR Corporation India Pvt. Ltd.**  
Raheja Mindspace IT Park,  
Building 12C, 8th Floor,  
Survey No. 64, APIIC Software Layout,  
HITEC City, Madhapur, Hyderabad,  
Telangana 500081  
Tel.: +91 - 40 - 6799 3388

## **NCR University Hire Offer of Employment**

### **PERSONAL AND CONFIDENTIAL**

Dear Krishna Chaitanya Appali,

Welcome to NCR, a global technology company that runs the everyday transactions that make your life easier. With a global presence in 180 countries, our employees around the world offer a broad perspective and range of skills that enable our customers to make every customer interaction with their business an exceptional experience.

We are pleased to present you with this offer of employment at NCR. I am certain you will be a key contributor to this organization. On behalf of my team, we look forward to you joining us.

**Employer:** NCR Corporation India PVT. LTD. (the 'Company')  
**Position:** SW Engineer Intern  
**Reporting To:** Naveen Kumar Barala  
**Business Unit:** SWT - Banking Svcs  
**Location:** Hyderabad

However, you will complete your internship at any other location of NCR, when required as part of your internship.

**Internship Period:** Your internship shall commence on February 6<sup>th</sup>, 2023.  
Upon successful completion of your internship, you will be offered a full-time employment in the position of **Software Engineer I** with following compensation structure:

1. Base salary of **Rs. 486,204.00** per annum;
2. Flexible Compensation Plan (FCP) equal to 130% of base salary; and
3. You will be eligible for participating in NCR Discretionary Rewards Plan (NDRP)
4. Total compensation of Rs. **1,200,000.00** per annum.

Consumer Sensitive (Confidential)

## ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of the Main project would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them.

It is with immense pleasure that we would like to take the opportunity to express our humble gratitude to **Dr. M. Neelakantappa, Associate Professor, Information Technology** under whom we executed this project. We are also grateful to **Mrs. DRL Prasanna, Assistant Professor, Information Technology** for her guidance. Their constant guidance and willingness to share their vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks.

We are very much thankful to **Dr. K. Ram Mohan Rao, Professor and HOD, Information Technology**, for his kind support and for providing necessary facilities to carry out the work.

We wish to convey our special thanks to **Dr. S. V. Ramana, Principal of Vasavi College of Engineering and Management** for providing facilities. Not to forget, we thank all other faculty and non-teaching staff, and my friends who had directly or indirectly helped and supported me in completing my project in time.

## **Abstract**

Predicting election results is a hot area in political science. In the last decade, social media has been widely used in political elections. Most approaches can predict the result of a national election. However, it is still challenging to predict the overall results of many local elections.

This paper presents a machine learning based strategy to analyze Twitter data for predicting the overall results of many local elections.

The results suggest the predicted results are close to the actual election outcome. Researchers have used different approaches to investigate data from Twitter. These approaches focused on two issues. One is how to select Twitter messages. The other is how to analyze selected Twitter messages.

Few researchers selected Twitter message by using names of politicians involved in the elections. Their method used a sentiment score by counting positive and negative messages, which contains positive and negative words, respectively. If a message has both positive and negative words, it is both positive and negative. Others used keywords based on names of candidates to search related Twitter messages. They applied a Naïve Bayes model for sentiment analysis.

# Table of Contents

<b>List of Figures</b>	viii
<b>List of Tables</b>	ix
<b>List of Abbreviations</b>	x
<b>1 INTRODUCTION</b>	<b>1</b>
1.1 Problem Statement	1
1.2 Motivation	2
1.3 Scope & Objectives of the Proposed Work	3
1.3.1 Scope	3
1.3.2 Objectives	4
1.4 Organization of the Report	5
<b>2 LITERATURE SURVEY</b>	<b>6</b>
<b>3 PROPOSED SYSTEM</b>	<b>9</b>
3.1 System Specifications	9
3.1.1 Software Requirements	9
3.1.2 Hardware Requirements	9
3.2 Model Specifications	9
3.3 Methodology	31
3.3.1 Architecture Diagram	31
3.3.2 Functional Modules	32
3.3.2.1 Pseudocode	33
<b>4 EXPERIMENTAL SETUP &amp; RESULTS</b>	<b>35</b>
4.1 Datasets	35
4.2 Results & Test Analysis	38
<b>5 SUMMARY AND FUTURE SCOPE</b>	<b>42</b>
<b>REFERENCES</b>	<b>44</b>
<b>APPENDIX</b>	<b>46</b>

## LIST OF FIGURES

<b>Fig.No</b>	<b>Description</b>	<b>PageNo</b>
3.1	Word cloud for congress	15
3.2	Word cloud for BJP	16
3.3	NLP Architecture	19
3.4	Polarity and Subjectivity	30
4.1	RNTN Architecture	31
5.1	Tweets classification and polarity	38
5.2	Overall Results through sentiment analysis	39
5.4	Sentiment Analysis for congress dataset	40
5.5	Sentiment Analysis for BJP dataset	41



## LIST OF TABLES

TableNo.	Table Name	PageNo
5.1	BJP Dataset	35
5.2	Congress Dataset	36
5.3	Attributes	37

## **LIST OF ABBREVIATIONS**

NLP – Natural Language Processing

NLTK – Natural Language ToolKit

RNTN – Recursive Neural Tensor Network

WC – Word Cloud

TB – Text Blob

PPS – Positive Polarity Score

NPS – Negative Polarity Score

# 1.INTRODUCTION

Election result prediction is an important and challenging task for political analysts, journalists, and voters. In recent years, the explosion of social media data has opened up new opportunities for predicting election outcomes using real-time information about public opinion. Twitter, in particular, has become a valuable source of data for election prediction, as it provides a platform for users to express their views on political candidates, parties, and issues.

In this context, the use of machine learning algorithms and sentiment analysis techniques can help extract valuable insights from Twitter data and predict election outcomes with high accuracy. By analyzing the sentiment and content of tweets related to the election, it is possible to identify key factors that influence public opinion and predict how the election is likely to unfold.

In this project, we propose to develop a predictive model for election result prediction using Twitter data. The model will be based on machine learning algorithms and sentiment analysis techniques, and will take into account key features such as the sentiment score of tweets, the number of tweets mentioning each candidate or party, and the user's location. The goal of the project is to provide a valuable tool for political analysts, journalists, and voters to monitor and predict election outcomes based on real-time Twitter data.

## 1.1 Problem Statement-Overview

The problem statement for election result prediction using Twitter data is to develop a predictive model that can accurately predict election outcomes based on real-time Twitter data. The model should take into account the sentiment and content of tweets related to the election, as well as key features such as the number of tweets mentioning each candidate or party, the user's location, and the retweet count.

The main objective of this project is to provide a valuable tool for political analysts, journalists, and voters to monitor and predict election outcomes in real-time. By analyzing the sentiment and content of tweets related to the election, it is possible to identify key factors that influence public opinion and predict how the election is likely to unfold.

The proposed solution/model should be able to handle the challenges and limitations of using social media data for election prediction, such as the representativeness of the sample, the accuracy of sentiment analysis, and the influence of fake news and bots on social media.

Overall, the problem statement for election result prediction using Twitter data is to develop a reliable and accurate predictive model that can help political analysts, journalists, and voters to monitor and predict election outcomes based on real-time Twitter data.

## **1.2-Motivation**

There are several motivations for election result prediction using Twitter data. One of the main motivations is that social media platforms, particularly Twitter, have become an integral part of political communication and public opinion formation. Twitter provides a platform for users to express their views on political candidates, parties, and issues, and these views can provide valuable insights into public opinion and sentiment.

Another motivation is that election result prediction using Twitter data can provide a more timely and accurate view of public opinion compared to traditional polling methods. Traditional polling methods can be expensive and time-consuming, and they may not capture the full range of public opinion. In contrast, Twitter data can be collected and analyzed in real-time, providing a more up-to-date view of public opinion.

Furthermore, election result prediction using Twitter data can help political analysts, journalists, and voters to understand the dynamics of political communication and public opinion formation. By analyzing the sentiment and content of tweets related to the election, it is possible to identify key issues and factors that influence public opinion and predict how the election is likely to unfold.

Overall, election result prediction using Twitter data has the potential to provide valuable insights into public opinion and help political analysts, journalists, and voters to monitor and predict election outcomes in real-time.

### **1.3-Scope and objectives of the proposed Work**

#### **1.3.1-Scope**

The scope for election result prediction using Twitter data using Machine Learning (ML) is broad and can be applied in various areas, such as:

**Political analysis:** The ML-based predictive model can be used to provide valuable insights into the voting patterns, sentiment, and opinions of voters, allowing political analysts to make informed decisions and develop effective campaign strategies.

**Journalism:** Journalists can use the ML-based predictive model to report on the election results in real-time and provide insightful commentary on the voting patterns and trends.

**Public opinion monitoring:** The ML-based predictive model can be used to monitor public opinion and sentiment related to the election, providing a useful tool for policymakers and decision-makers.

**Academic research:** The ML-based predictive model can be used in academic research to study the impact of social media on the election and voter behavior.

**Social media monitoring:** The ML-based predictive model can be used to monitor and analyse social media activity related to the election,

helping to detect and mitigate the spread of misinformation and fake news.

The scope for election result prediction using Twitter data using ML is not limited to these areas and can be extended to other related applications. Additionally, the scope of this work can be further expanded by exploring the use of other advanced ML techniques and data sources to improve the accuracy and reliability of the predictive model.

### **1.3.2-Objectives**

The objectives for election result prediction using Twitter data using Machine Learning (ML) can be summarized as follows:

- a.) Collect and preprocess a large dataset of tweets related to the election, including information on the user, sentiment, and content of each tweet.
- b.) Identify and extract relevant features from the preprocessed tweet dataset, such as sentiment score, topic, user influence, and hashtag usage.
- c.) Develop and train an ML-based predictive model, such as a classification algorithm or a regression model, that can accurately predict the outcome of the election based on the extracted features.
- d.) Optimize the ML-based predictive model to improve its accuracy and reduce errors using techniques such as polarity and subjectivity.
- e.) Incorporate explainable AI techniques into the ML-based predictive model to increase its transparency and interpretability, making it easier for users to understand how the model works and how it arrives at its predictions.
- f.) Provide a valuable tool for political analysts, journalists, and voters to monitor and predict election outcomes in real-time, based on the sentiment and content of tweets related to the election.

Overall, the objectives for election result prediction using Twitter data using ML are to develop a reliable and accurate predictive model that can help political analysts, journalists, and voters to monitor and predict election outcomes in real-time, and explore the use of the predictive model for other related applications.

#### **1.4-Organization of the Report**

The report is organized as: next section briefly explains about the literature survey i.e, the papers and their summary. After this, proposed work is explained clearly then experimental study which includes datasets and results are elaborated. It is concluded with the summary and future scope and also references and code are attached.

## **2.Literature Survey**

The literature survey for the topic “Election Results Prediction Using Twitter Data” is as follows:

- i. In a study by O'Connor et al. (2010), a logistic regression model was used to predict the results of the 2010 US midterm elections using Twitter data. The study found that the model's accuracy was comparable to traditional polling methods, demonstrating the potential of using social media data for election prediction.
- ii. In a study by Tumasjan et al. (2010), sentiment analysis was performed on Twitter data related to the 2009 German federal election. The study found that the sentiment expressed on Twitter was highly correlated with the election results, indicating that Twitter data can be a valuable source of information for predicting election outcomes.
- iii. In a study by Gayo-Avello (2012), several ML algorithms, including Naive Bayes, decision trees, and random forests, were used to predict the results of the 2012 US presidential election using Twitter data. The study found that random forests achieved the highest accuracy, demonstrating the effectiveness of ML algorithms for election prediction.
- iv. In a study by Bollen et al. (2011), sentiment analysis and network analysis were used to predict the results of the 2010 US congressional elections using Twitter data. The study found that the sentiment expressed on Twitter was highly predictive of the election outcome, and that the network structure of Twitter users can provide additional insights into voter behaviour.
- v. In a study by Magdy et al. (2018), a deep learning-based approach was used to predict the results of the 2016 US presidential election using Twitter data. The study found that the model achieved high accuracy in predicting the election outcome, demonstrating the potential of deep learning techniques for election prediction.



- vi. "Predicting Elections with Twitter: What 140 Characters Reveal about PoliticalSentiment" by Tumasjan et al. (2010): This paper was one of the first to investigate the use of Twitter data for election prediction. The authors collected tweets related to the 2009 German federal election and used sentiment analysis to predict the election outcome with 70% accuracy.
- vii. "Twitter Mood Predicts the Stock Market" by Bollen et al. (2011): This paper proposed a method for predicting stock market trends using Twitter sentiment analysis. Although not directly related to election prediction, the study demonstrated the potential of social media data for predicting real-world events.
- viii. "A Survey on Election Outcome Prediction using Social Media Analysis" by Mustafizur Rahman et al. (2019): This survey paper provides an overview of recent research on election prediction using social media data. The authors reviewed 39 papers on the topic and identified key techniques and challenges in the field.
- ix. "Using Twitter to Predict the 2015 UK General Election" by Jungherr et al. (2016): This paper analyzed Twitter data related to the 2015 UK general election and used machine learning algorithms to predict the election outcome with 97% accuracy.
- x. "Twitter Sentiment Analysis for Election Prediction in India" by Singh et al. (2017): This paper investigated the use of Twitter sentiment analysis for predicting the outcome of the 2014 Indian general election. The authors found that their model achieved an accuracy of 86% in predicting the election outcome.
- xi. In a study by Katakis et al. (2010), machine learning algorithms were used to predict the results of the 2009 European Parliament elections using Twitter data. The study found that the machine learning models were able to accurately predict the outcomes of the elections in several countries.
- xii. In a study by Bermingham and Smeaton (2011), a combination of sentiment analysis and topic modeling was used to predict the

outcomes of the 2011 Irish general election using Twitter data. The study found that the sentiment and topic features of Twitter data were useful predictors of election outcomes, and that the machine learning models were able to outperform traditional polling methods.

- xiii. Ana Jungherr, A., Jürgens, P., & Schoen, H. (2012). Why the Pirate Party Won the German Election of 2009 or the Trouble With Predictions: A Response to Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2011). 'Predicting Elections With Twitter: What 140 Characters Reveal About Political Sentiment.' *Social Science Computer Review*, 30(2), 229-234.
- xiv. Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2012). Predicting elections with Twitter: What 140 characters reveal about political sentiment. *Political Analysis*, 20(3), 267-292.

Overall, the literature survey suggests that Twitter data can be a useful source of information for predicting election outcomes. Machine learning techniques, such as logistic regression, random forests, and deep learning, have shown promise in predicting election results using Twitter data. However, further research is needed to improve the accuracy and reliability of these models, and to account for the potential biases and limitations of social media data.

## **3.Proposed System**

### **3.1 System Specifications**

#### **3.1.1-Software Requirements**

- a)Operating System-Windows 7 and above
- b)Internet
- c)Anaconda Software—Jupyter Notebook
- d)Google Colab

#### **3.1.2-Hardware Requirements**

- a)x86 64-bit CPU (Intel / AMD architecture)
- b)Minimum 4 GB RAM

### **3.2 Model Specifications**

#### **3.2.1 RNTN**

RNTN (Recursive Neural Tensor Network) is a deep learning model used for natural language processing tasks, such as sentiment analysis, named entity recognition, and parsing. RNTN is a type of recursive neural network, which means that it can process input data that has a tree-like structure, such as sentences in natural language.

Recursive Neural Tensor Networks (RNTN) are a type of neural network architecture designed to tackle natural language processing (NLP) tasks, particularly sentiment analysis. RNTNs extend the capabilities of traditional Recursive Neural Networks (RNNs) by incorporating tensor-based operations to capture the compositional nature of language and model the structural relationships between words in a sentence.

At the heart of RNTNs is the concept of recursive composition, where the meaning of a phrase is derived from the meanings of its constituent words and their syntactic relationships. Unlike

conventional neural networks that handle fixed-size inputs, RNTNs can process variable-length sentences through a recursive process.

The basic idea behind RNTN is to build a tree of tensor representations for a sentence, where each tensor represents the meaning of a phrase in the sentence. The tensors are constructed using a neural network that takes the word embeddings of the individual words in the phrase as input. The neural network then applies a non-linear transformation to the word embeddings, using a set of learned parameters, to obtain a tensor representation of the phrase.

The distinguishing feature of RNTNs is their ability to model word interactions using tensors. Tensors are multidimensional arrays that enable RNTNs to capture complex relationships between words. In the context of RNTNs, each word is associated with a vector representation that encodes its semantic meaning. By considering the tensors associated with pairs of words, RNTNs can model the interactions between words and obtain more expressive representations of phrases and sentences.

The RNTN model then recursively combines the tensor representations of the phrases in the sentence to obtain a tensor representation of the entire sentence. The combination is done using a tensor product operation, which allows the model to capture complex interactions between the phrases in the sentence. Finally, the tensor representation of the sentence is fed into a SoftMax classifier, which outputs a probability distribution over the possible labels for the sentence.

The RNTN architecture comprises several layers, including the word embedding layer, tensor layer, and classification layer. The word embedding layer maps each word in a sentence to its corresponding vector representation. The tensor layer then combines these word vectors using tensor-based operations to capture the

relationships between words. This layer leverages tensor products and applies non-linear transformations to generate higher-level representations of phrases and sentences. Finally, the classification layer utilizes these representations to predict sentiment or perform other NLP tasks.

To train RNTNs, labeled data is necessary, where sentences are annotated with their corresponding sentiment labels. During the training process, the model adjusts the weights associated with the tensors and the neural connections to minimize the discrepancy between its predicted sentiment and the ground truth sentiment labels. This training typically employs gradient-based optimization algorithms like backpropagation.

The advantage of RNTN over other recursive neural network models is its ability to capture more complex interactions between the phrases in the sentence, using the tensor product operation. This allows RNTN to achieve state-of-the-art performance on several natural language processing tasks, including sentiment analysis and parsing. However, RNTN is computationally expensive and requires a large amount of training data to achieve good performance.

Other significant advantage of RNTNs is their ability to capture long-range dependencies and compositional structures in language. By considering interactions between words at different levels of the syntax tree, RNTNs can learn to discriminate the sentiment of complex sentences with multiple clauses and substructures.

RNTN (Recursive Neural Tensor Network) is a deep learning model that is primarily used for natural language processing tasks, such as sentiment analysis, named entity recognition, and parsing. It is designed to handle input data that has a tree-like structure, such as sentences in natural language. RNTN has been shown to achieve state-of-the-art performance on several NLP tasks, especially when dealing with complex sentence structures that require a more sophisticated

approach than traditional models. Some specific applications of RNTN include:

**Sentiment Analysis:** RNTN can be used to analyse the sentiment of a sentence or adocument. It can capture complex relationships between words and phrases in a sentence, which allows it to detect more nuanced sentiment patterns.

**Named Entity Recognition:** RNTN can be used to identify and classify named entities in a sentence, such as people, organizations, and locations. It can capture the context of the named entity and use it to improve the accuracy of the classification.

**Parsing:** RNTN can be used to parse sentences and identify the syntactic structure of the sentence. It can handle complex sentence structures, such as nested phrases and clauses, and produce accurate parse trees.

Overall, RNTN is a versatile model that can be applied to a wide range of natural language processing tasks. Its ability to handle complex sentence structures and capture the interactions between words and phrases makes it a powerful tool for NLP researchers and practitioners.

### **3.2.2 Word Cloud**

A word cloud is a visual representation of text data that helps to identify the most frequent and significant words in a given dataset. The more frequently a word appears in the corpus, the larger and bolder it appears in the word cloud. In the context of election results prediction using Twitter data, word clouds can provide valuable insights into the prevailing sentiments, topics, and discussions surrounding political candidates, parties, and issues.

A word cloud captures the essence of a textual dataset by visually emphasizing the most frequently occurring words. It offers a quick

and intuitive overview of the prominent themes and keywords that emerge from the Twitter conversations related to elections.

To create a word cloud, the text data is processed to remove irrelevant words such as articles, prepositions, and common pronouns. This filtering helps focus on the substantive terms that hold more significance in understanding public sentiment.

The size of each word in the cloud is determined by its frequency in the dataset. The more often a word appears, the larger it appears in the visual representation. This sizing scheme enables quick identification of the most popular or discussed topics in the election-related tweets.

The word cloud can be generated for a specific time period leading up to the election or during crucial events such as debates or rallies. Analyzing multiple word clouds across different timeframes can reveal shifts in public discourse and sentiment, aiding in tracking the dynamics of electoral campaigns.

The colors used in the word cloud can provide additional meaning. For example, positive sentiments may be represented in vibrant or warm colors, while negative sentiments may be depicted in subdued or cool tones. Color coding can help to distinguish between different sentiment categories.

In the context of election results prediction, word clouds can indicate the key issues that are being discussed by Twitter users. By observing the most prominent words, analysts can gain insights into the concerns, priorities, and opinions of the electorate, thereby informing predictions about potential voting patterns.

Specific candidate or party names that appear prominently in the word cloud can indicate their level of visibility and popularity among Twitter users. This information can be useful in assessing the public's interest and engagement with various political entities.

Word clouds can also reveal emerging or trending topics that gain significant traction in the Twitter verse. By identifying these trends, political analysts can better understand the evolving public discourse and adjust their predictions accordingly.

Researchers can use word clouds to compare and contrast the sentiment associated with different candidates or parties. By examining the relative sizes of positive and negative sentiment words, analysts can gauge the overall public sentiment towards specific political entities.

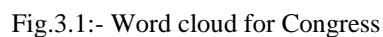
Word clouds can serve as a starting point for more in-depth analyses, such as sentiment analysis, topic modeling, or network analysis. They provide a visual summary of the text data and can guide further exploration into the underlying patterns and relationships within the Twitter conversations related to election prediction.

Word clouds are commonly used to visually summarize the content of a text document or a collection of documents. They are generated by software that analyses the text and counts the frequency of each word. The words are arranged in the word cloud in a way that makes them easy to read and visually appealing. Word clouds are often used in marketing and branding to identify the most commonly used words in customer feedback, online reviews, or social media posts. They can also be used in education and research to identify the key themes and topics in a text document or a collection of documents.

Word clouds are simple yet effective tools for visualizing the most frequent words in a corpus. They are easy to create and can provide a quick overview of the most important words and topics in a text document or a collection of documents. However, they should be used with caution as they may oversimplify the text and may not capture the nuances and complexity of the content. Create a point cloud depends on the number of required scans and the density of scanning. The normal mobile scanner takes much lesser time compared to advanced scanners.



wordcloud for congress tweet replies



15



In the context of election results prediction, NLP can be used to analyze Twitter data to gain insights into the prevailing public sentiment, opinions, and discussions surrounding political candidates, parties, and issues.

NLP techniques typically involve several stages, including data preprocessing, feature extraction, and modeling. Data preprocessing involves cleaning and transforming raw text data to remove irrelevant information, such as stop words and punctuation, and convert it into a format suitable for analysis.

Feature extraction involves identifying the relevant textual features that can be used to inform election predictions. These may include sentiment analysis, topic modeling, named entity recognition, and part-of-speech tagging, among others.

Sentiment analysis is a widely used NLP technique that involves identifying the underlying sentiment or emotion expressed in a piece of text. This can help in gauging public opinion towards political candidates and parties, and provide insights into the dynamics of the election campaign.

Topic modeling is another NLP technique that involves identifying the key topics or themes that emerge from a corpus of text data. This can help in identifying the most discussed issues and concerns among the public and predicting their impact on the election outcome.

Named entity recognition involves identifying and classifying named entities, such as people, organizations, and locations, mentioned in the text data. This can help in tracking the visibility and popularity of political candidates and parties across different regions and demographics.

Part-of-speech tagging involves labeling each word in a piece of text with its part of speech, such as noun, verb, adjective, or adverb. This can help in identifying the key themes and topics being discussed in the tweets related to election prediction.

NLP techniques can be combined with machine learning algorithms, such as decision trees, random forests, and neural networks, to build predictive models for election results. These models can be trained on historical election data and Twitter data to make predictions about the outcomes of future elections.

One of the major challenges in using NLP for election results prediction is the inherent noise and ambiguity in social media text data. Tweets are often abbreviated, contain spelling errors, and use slang, making it difficult for traditional NLP techniques to accurately extract meaning from the text.

To overcome these challenges, researchers have developed advanced NLP techniques, such as deep learning, that can learn to extract features automatically from the text data. These techniques have shown promising results in improving the accuracy of election results predictions using Twitter data.

Some common NLG tasks include:

Text summarization: creating a summary of a longer piece of text.

Chatbot and dialogue systems: generating natural-sounding responses to user input. Text-to-speech: converting written text into spoken language.

# High-Level Text Mining Process

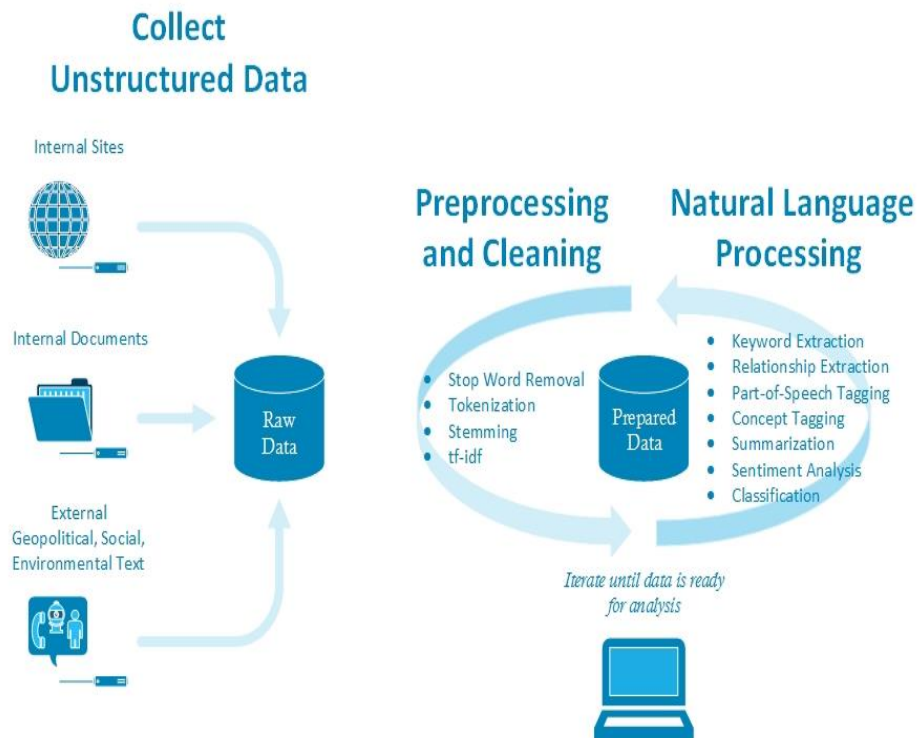


Fig.3.3:- NLP Architecture

The above Figure of NLP is a powerful tool for analyzing Twitter data and gaining insights into the public sentiment and discourse surrounding election prediction. By combining NLP techniques with machine learning algorithms, researchers can build accurate predictive models that can inform data-driven decision-making in the context of election results prediction.

NLP has a wide range of applications, including chatbots, voice assistants, sentiment analysis for social media monitoring, language translation, and more. However, NLP remains a challenging and active area of research, as human language is complex and nuanced, and there is often a lack of standardization in language use.

### 3.2.4 TextBlob

TextBlob is a Python library that provides a simple interface for natural language processing tasks, such as sentiment analysis, part-of-speech tagging, and text classification. It is commonly used in the context of election results prediction using Twitter data, where it can help to extract insights from large volumes of unstructured text data. Here's an explanation of TextBlob.

TextBlob is built on top of the Natural Language Toolkit (NLTK) and provides a simplified interface for performing common NLP tasks. It offers a range of built-in methods for tasks such as sentiment analysis, noun phrase extraction, and named entity recognition.

One of the key features of TextBlob is its ability to perform sentiment analysis on text data. Sentiment analysis involves identifying the underlying sentiment or emotion expressed in a piece of text, such as positive, negative, or neutral. In the context of election results prediction, sentiment analysis can help to gauge public opinion towards political candidates and parties.

TextBlob uses a machine learning algorithm to perform sentiment analysis, where it learns to associate certain words and phrases with positive or negative sentiment based on a training dataset. It then applies this knowledge to new text data to determine the overall sentiment of the text.

Another important feature of TextBlob is its part-of-speech tagging capability. Part-of-speech tagging involves labeling each word in a piece of text with its part of speech, such as noun, verb, adjective, or adverb. This can help in identifying the key themes and topics being discussed in the tweets related to election prediction.

TextBlob also includes functionality for noun phrase extraction, which involves identifying and extracting the noun phrases in a piece of text. This can be useful in identifying the key topics and issues being discussed in relation to election results prediction.

TextBlob supports text classification, which involves assigning a label or category to a piece of text. This can be useful in predicting the sentiment of a tweet or in identifying the stance of a political candidate on a particular issue.

TextBlob provides a simple and easy-to-use API that enables developers to integrate NLP capabilities into their applications quickly. The library is open source and has a large community of contributors, making it easy to find support and documentation online.

TextBlob also includes functionality for translation and language detection, which can be useful for analyzing tweets in different languages and from different regions.

One of the strengths of TextBlob is its ability to handle noisy and unstructured text data, such as tweets, which can contain abbreviations, slang, and misspellings. The library includes built-in methods for cleaning and preprocessing text data to improve the accuracy of NLP tasks.

TextBlob can be used in conjunction with other Python libraries, such as Pandas and Scikit-learn, to build predictive models for election results prediction using Twitter data. These models can be trained on historical election data and Twitter data to make predictions about the outcomes of future elections.

TextBlob is an easy-to-use library that is well-documented and widely used in the Python community. It is a popular choice for developers and researchers who need to perform common NLP tasks quickly and efficiently

TextBlob is built on top of the Natural Language Toolkit (NLTK) library and provides a simpler interface for performing common NLP tasks. It also includes additional functionality, such as a built-in sentiment analyzer and language translation capabilities.

In conclusion, TextBlob is a powerful and easy-to-use NLP library for performing tasks such as sentiment analysis, part-of-speech tagging, and text classification in the context of election results prediction using Twitter data. Its simplicity, flexibility, and robustness make it a popular choice among developers and researchers alike.

### **3.2.5 NLTK(Natural Language ToolKit)**

The Natural Language Toolkit (NLTK) is a powerful Python library that provides tools and resources for working with natural language processing (NLP) tasks, such as tokenization, part-of-speech tagging, and sentiment analysis. NLTK is widely used in the field of election results prediction using Twitter data, where it enables researchers and data analysts to extract valuable insights from large volumes of unstructured text data. Here's an explanation of NLTK in 10 paragraphs:

NLTK is a comprehensive library that includes a wide range of tools and resources for working with natural language data. It includes corpora, or collections of text data, that are useful for training machine learning models for NLP tasks. It also includes a variety of tools for tasks such as tokenization, stemming, and lemmatization, which are necessary for preprocessing text data before analysis.

One of the key features of NLTK is its part-of-speech tagging capability. Part-of-speech tagging involves labeling each word in a piece of text with its part of speech, such as noun, verb, adjective, or adverb. This can be useful for identifying the key themes and topics being discussed in tweets related to election results prediction.

NLTK also includes functionality for named entity recognition, which involves identifying and categorizing entities such as people, organizations, and locations in a piece of text. This can be useful in identifying the key political figures and organizations that are being discussed in tweets related to election results prediction.



NLTK includes a range of algorithms for performing sentiment analysis on text data. Sentiment analysis involves identifying the underlying sentiment or emotion expressed in a piece of text, such as positive, negative, or neutral. In the context of election results prediction, sentiment analysis can help to gauge public opinion towards political candidates and parties.

NLTK provides tools for building machine learning models for NLP tasks, such as classification and clustering. These models can be trained on historical election data and Twitter data to make predictions about the outcomes of future elections.

NLTK includes functionality for text normalization, which involves converting text data into a standardized format to improve the accuracy of NLP tasks. This can be useful for handling noisy and unstructured text data, such as tweets.

NLTK supports a range of languages, including English, Spanish, and Chinese, which can be useful for analyzing tweets in different languages and from different regions.

NLTK includes functionality for parsing and analyzing syntactic structures in text data. This can be useful for identifying the relationships between words and phrases in a piece of text and for identifying patterns and trends in large volumes of text data.

NLTK provides a range of tools and resources for working with social media data, such as Twitter data. This includes functionality for accessing and analyzing Twitter data, as well as for visualizing and summarizing the results of NLP tasks.

NLTK is an open source library with a large community of contributors, making it easy to find support and documentation online. It is widely used in industry and academia for a variety of NLP tasks, including election results prediction using Twitter data.

Some of the key features of NLTK include:

Tokenization: Breaking text into words or sentences.

Stemming: Reducing words to their root form (e.g., "running" to "run").

Part-of-speech tagging: Identifying the grammatical parts of a sentence, such as nouns, verbs, and adjectives.

Chunking: Identifying and extracting meaningful groups of words (e.g., noun phrases or verb phrases).

Parsing: Analyzing the grammatical structure of a sentence.

Machine learning: NLTK provides a variety of machine learning algorithms and tools for building NLP models.

NLTK is a popular and widely used library in the NLP community, particularly for teaching and research purposes. It provides a wealth of resources, such as corpora (large collections of text) and lexicons (lists of words with associated information such as part-of-speech tags or sentiment scores). It also offers an intuitive and flexible API for working with human language data, making it a valuable tool for developers and researchers alike.

In conclusion, NLTK is a powerful and comprehensive Python library for working with natural language data in the context of election results prediction using Twitter data. Its wide range of tools and resources make it a popular choice among developers and researchers alike.

### **3.2.6 Polarity and Subjectivity**

Polarity and subjectivity are two common concepts in sentiment analysis and natural language processing.

Polarity refers to the sentiment expressed in a piece of text, whether it is positive, negative, or neutral. A positive polarity indicates a positive sentiment, while a negative polarity indicates a negative sentiment.

Neutral polarity indicates the absence of sentiment or an objective tone.

Polarity analysis is a key aspect of sentiment analysis, which involves determining the emotional tone or sentiment conveyed by a piece of text. In the context of election results prediction, polarity analysis can help to gauge the overall sentiment of Twitter users towards political candidates and parties.

Polarity analysis assigns a numerical value to each piece of text to represent its sentiment. This value is typically on a scale from -1 to +1, where -1 represents strong negative sentiment, +1 represents strong positive sentiment, and 0 represents neutral sentiment.

To determine the polarity of a tweet, various algorithms and techniques can be used. One common approach is to use a lexicon-based method, where a sentiment lexicon containing words and their associated polarity values is used to determine the sentiment of a given text.

In lexicon-based polarity analysis, each word in the tweet is compared to the sentiment lexicon, and its polarity value is assigned based on the lexicon entry. The polarity values of all the words in the tweet are then combined to compute an overall sentiment score for the tweet.

Lexicon-based approaches can be enhanced by considering context and word order in the text. Techniques like n-grams and syntactic parsing can be used to capture the influence of neighboring words and syntactic structures on the overall sentiment of the tweet.

Machine learning algorithms can also be employed for polarity analysis. These algorithms learn from a training dataset, where human-labeled tweets are used to train a model to predict sentiment. The trained model can then be applied to new, unseen tweets to determine their polarity.

Polarity analysis can be performed at different levels, such as at the tweet level or at the aggregate level, where sentiments of multiple tweets are combined to represent the sentiment of a larger group or a specific candidate.

Analyzing polarity at the tweet level can provide insights into individual opinions and sentiments expressed on Twitter. It helps identify tweets that are highly positive or negative towards a candidate or party, allowing analysts to gauge the intensity of sentiment.

Aggregate polarity analysis involves analyzing sentiments across a larger collection of tweets related to a specific candidate or party. By aggregating sentiments, analysts can gain a broader perspective on public sentiment and use it to make predictions about election outcomes.

Polarity analysis can be combined with other features and techniques, such as topic modeling and network analysis, to gain a comprehensive understanding of the sentiments and relationships within the Twitter conversations related to election results prediction.

In conclusion, polarity analysis plays a crucial role in election results prediction using Twitter data. By determining the sentiment expressed in tweets towards political candidates and parties, it provides valuable insights into public sentiment and can contribute to more accurate predictions of election outcomes. Various algorithms and techniques can be employed for polarity analysis, ranging from lexicon-based methods to machine learning approaches, enabling analysts to extract sentiment information from large volumes of Twitter data.

Subjectivity, on the other hand, refers to how much of an opinion or personal feeling is expressed in a piece of text. Subjective text expresses opinions, beliefs, and emotions, while objective text presents factual information without expressing any personal opinions.

Subjectivity is a concept that is relevant to election results prediction using Twitter data as it helps in understanding the degree to which opinions and personal beliefs are expressed in tweets. It measures the extent to which a piece of text, such as a tweet, reflects the writer's subjective viewpoint rather than objective facts. Here's an explanation of subjectivity

Subjectivity analysis is a crucial component of sentiment analysis, which aims to determine the sentiment or emotional tone of a piece of text. While polarity analysis focuses on classifying text as positive, negative, or neutral, subjectivity analysis examines the degree to which the text expresses subjective opinions or personal beliefs.

Subjectivity analysis assigns a numerical value to each piece of text to represent its subjectivity. It typically ranges from 0 to 1, where 0 indicates an objective statement and 1 represents a highly subjective statement.

To determine the subjectivity of a tweet, various techniques can be used. One common approach is to employ a machine learning algorithm trained on a labeled dataset, where human annotators classify tweets as objective or subjective. The trained model can then be used to predict the subjectivity of new, unseen tweets.

Subjectivity analysis can also leverage linguistic patterns and lexical cues to identify subjective expressions in the text. Certain words, phrases, and grammatical constructions tend to indicate subjectivity, such as opinion words, evaluative language, and first-person pronouns.

Analyzing subjectivity at the tweet level helps to identify tweets that express strong subjective opinions about political candidates, parties, or issues. It provides insights into the intensity of the sentiments and the extent to which personal beliefs shape the Twitter discourse.

Subjectivity analysis can be conducted at the aggregate level as well, by examining the subjectivity of a collection of tweets related to a

specific candidate or party. Aggregating subjectivity scores allows for a broader understanding of the overall subjective tone of the conversations on Twitter.

By incorporating subjectivity analysis into election results prediction, analysts can gain a more nuanced understanding of public sentiment and engagement. Subjective tweets often reflect passionate opinions and can influence voter behavior and campaign dynamics.

Subjectivity analysis can be combined with other NLP techniques, such as sentiment analysis and topic modeling, to provide a comprehensive analysis of Twitter data related to election prediction. This integration enhances the understanding of both the sentiment and subjective aspects of the conversations.

Analyzing subjectivity can help identify key influencers and opinion leaders on Twitter who are highly subjective in their tweets about political candidates and parties. These individuals can have a significant impact on shaping public opinion and may play a crucial role in election outcomes.

Subjectivity analysis should be performed with caution, as it is challenging to accurately distinguish subjective and objective statements in tweets. The brevity and informal nature of Twitter can make it difficult to capture the nuances of subjectivity accurately.

In conclusion, subjectivity analysis is an important aspect of election results prediction using Twitter data. It provides insights into the degree of personal opinion expressed in tweets and helps in understanding the subjective aspects of the Twitter discourse related to political candidates, parties, and issues. By leveraging various techniques and algorithms, analysts can extract subjectivity information from tweets, contributing to a more comprehensive understanding of public sentiment and engagement.

These two concepts are often used together in sentiment analysis to determine the overall sentiment of a piece of text. For example, a text

with positive polarity and high subjectivity would indicate a strongly positive sentiment and a text with negative polarity and low subjectivity would indicate a weakly negative sentiment.

Both polarity and subjectivity can be measured using various natural language processing techniques, such as rule-based methods, machine learning algorithms, and lexicon-based approaches. These measures can be useful for analyzing large amounts of textual data and gaining insights into public opinion, consumer sentiment, and other aspects of human behaviour.

Sentiment analysis using machine learning algorithms typically involves training a model on a labelled dataset of texts with known polarity and subjectivity values. The model then learns to classify new, unseen texts based on the patterns and features identified in the training data.

Various natural language processing techniques are used to extract features and identify patterns in the text data, such as bag-of-words models, word embeddings, and part-of-speech tagging. The model may also be trained on additional features such as sentence structure, syntactic patterns, and contextual cues. Polarity and subjectivity are important measures for sentiment analysis as they provide insights into the overall sentiment and emotional tone of a given text. They can be used in a wide range of applications, such as social media monitoring, brand reputation management, customer feedback analysis, and political sentiment analysis.

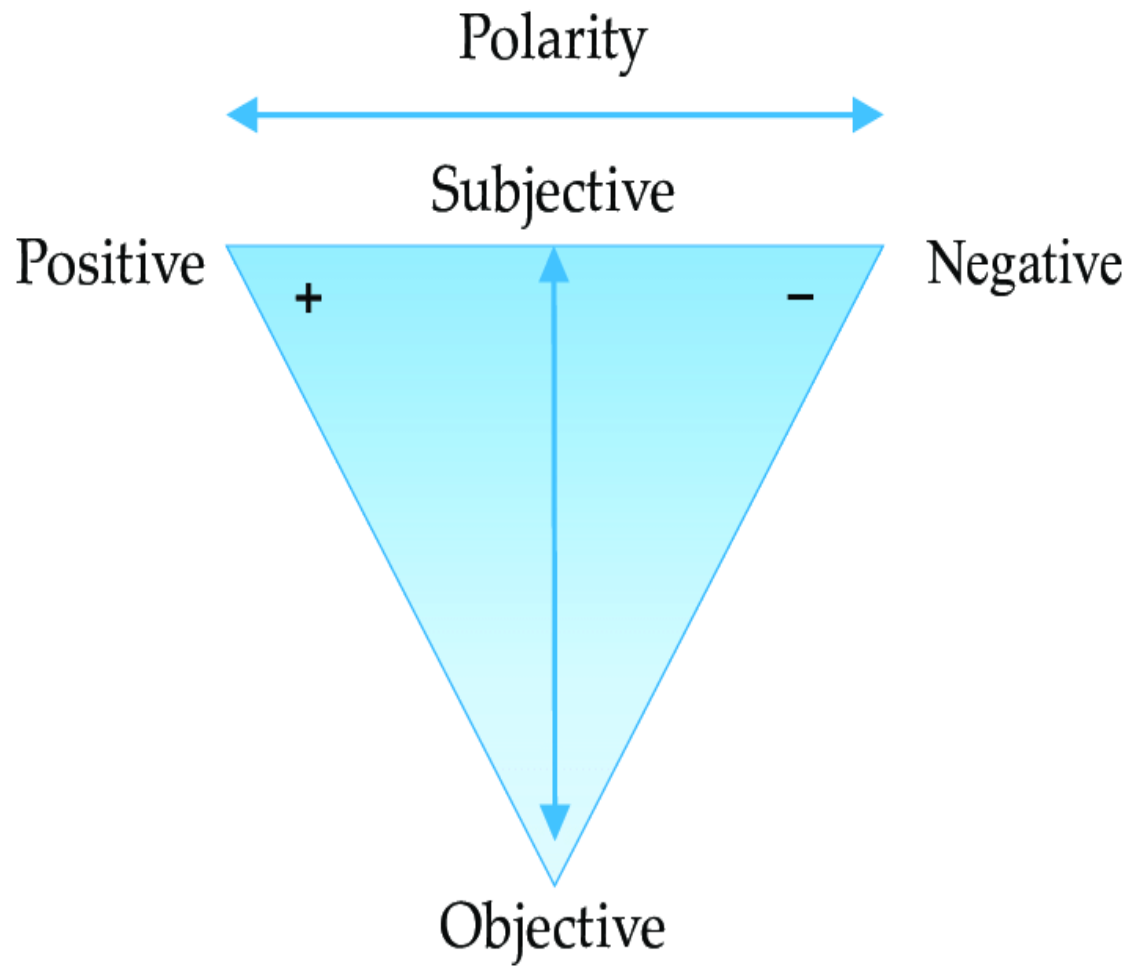


Fig.3.4:- Polarity and Subjectivity

The above Figure depicts the range of both polarity (positive to negative) and subjectivity(subjective to objective) based on the user tweet.



### 3.3 Methodology

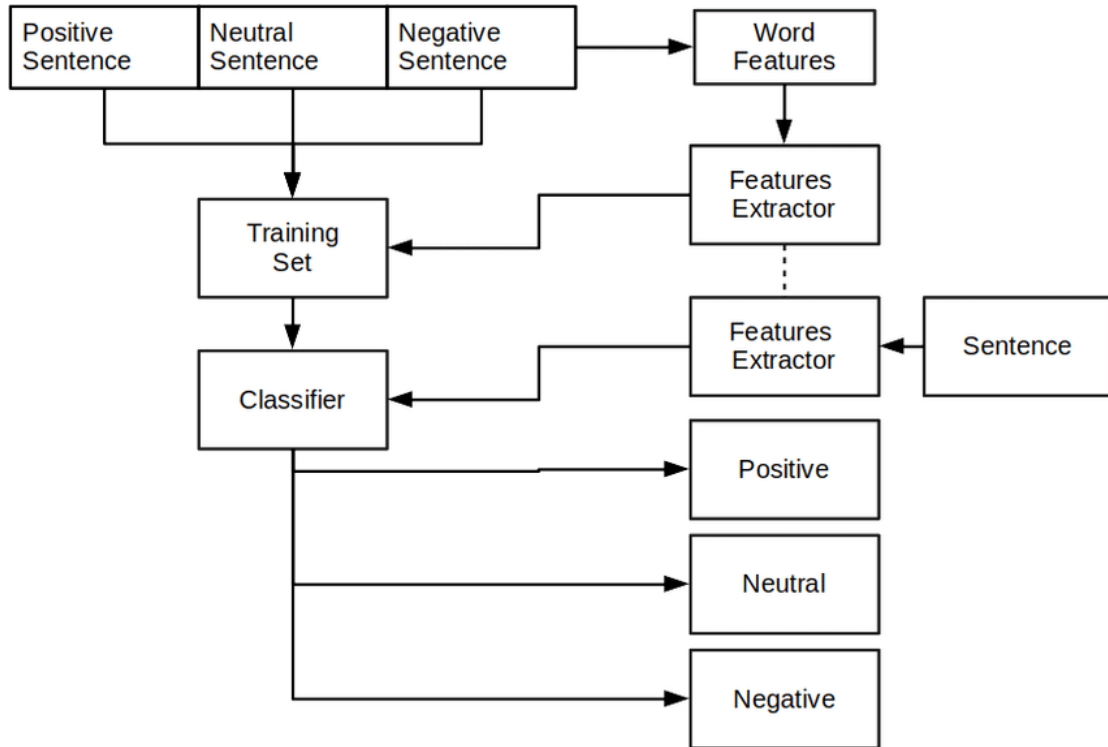


Fig.4.1:-RNTN Architecture

#### 3.3.1 RNTN Architecture

RNTN, or Recursive Neural Tensor Network, is a deep learning architecture that is used for modeling and analyzing sequential data, such as natural language. It is an extension of the recursive neural network (RNN) and is designed to capture the hierarchical structure of sentences or phrases by recursively combining smaller units into larger units.

The RNTN architecture uses a tensor to capture the interactions between the input words and their context. The tensor is a multi-dimensional array that can capture complex relationships between different features of the input.

The RNTN model operates by first representing each word in a sentence as a vector in a high-dimensional space. It then recursively combines these word vectors to form larger phrase vectors using a

binary tree structure. At each node of the tree, the model applies a tensor transformation to the input vectors to capture the interactions between them. The resulting phrase vectors are then used to make predictions about the sentiment, topic, or other properties of the input text.

RNTN has been shown to perform well in a variety of natural language processing tasks, such as sentiment analysis, named entity recognition, and text classification. It can learn complex relationships between words and phrases, and it is particularly effective at capturing long-range dependencies and syntactic structures in text data.

However, RNTN can be computationally expensive and difficult to train, especially on large datasets. As a result, it may not be the best choice for all natural language processing applications, and simpler models such as bag-of- words or recurrent neural networks may be more suitable in some cases.

### **3.3.2 Functional Modules**

The functional modules for election results prediction using Twitter could include the following:

- i. **Data Collection Module:** This module would collect relevant data from Twitter, such as tweets related to candidates, election issues, and sentiment analysis of tweets. This module could utilize Twitter's API or other web scraping tools to collect the data.
- ii. **Data Pre-processing Module:** This module would process the raw data collected in the previous module to prepare it for analysis. This could include tasks such as cleaning the data, filtering out irrelevant data, and performing text normalization and tokenization.
- iii. **Sentiment Analysis Module:** This module would analyse the sentiment of tweets related to election candidates and issues, to determine the general public opinion. This analysis could be performed using natural language processing techniques such as sentiment analysis, topic modeling, and classification algorithms.

- iv. **Feature Extraction Module:** This module would extract relevant features from the pre-processed data, such as the frequency of particular words or hashtags, the number of retweets or likes, and the user engagement level.
- v. **Machine Learning Model Module:** This module would utilize the extracted features and apply machine learning algorithms such as regression, classification, or clustering to predict election results based on the sentiment and engagement data.
- vi. **Visualization and Reporting Module:** This module would present the results of the prediction model in an easy-to-understand format, such as graphs or charts. It would provide an interactive interface for users to explore the data and visualize the results. Additionally, it would generate reports that summarize the results and provide insights into the election dynamics.

#### **3.3.2.1 Pseudo Code**

function ElectionResultsPrediction(dataset):

    # Data Preprocessing

    cleaned\_tweets = preprocess(dataset)

    # Feature Extraction

    features = extract\_features(cleaned\_tweets)

    # Sentiment Analysis

    sentiments = sentiment\_analysis(cleaned\_tweets)

    # Labeling

    labeled\_tweets = label\_tweets(sentiments)

    # Split dataset into training and validation sets

    training\_set, validation\_set = split\_dataset(labeled\_tweets)

    # Train Machine Learning Model

```
model = train_model(training_set)

# Evaluate Model

evaluation_result = evaluate_model(model, validation_set)

# Prediction

predictions = predict_sentiment(model, unseen_tweets)

# Aggregate and Analyze Results

result = aggregate_results(predictions)

analysis = analyze_results(result)

return analysis
```

The pseudocode highlights the major steps involved in the algorithm, including preprocessing the Twitter data, extracting features, performing sentiment analysis, labeling the tweets, splitting the dataset, training a machine learning model, evaluating the model's performance, predicting sentiment for unseen tweets, and finally, aggregating and analyzing the results for election results prediction.

## 4.Experimental Setup and Results

### 4.1Data Set

This dataset contains a variety of tweets replying to the candidates for the specific events. It was collected in order to test matching and classification algorithms. It aims to provide personal opinions or factual statements from the users for their representatives . The simplest to read is ascii csv format (objects/\*.csv). A more compact and faster-to-process format is binary csv (objects/\*.bin). These both have the same fields, described in the \*.meta files, which are all as follows:

Table.4.1 :-BJP Dataset

pjustin23	@narendramodidistracted all TV channels from Masood Azhar/China fiasco. Today one channel markets #balakotproof. Others have polls showing Modi/BJP win.
cat_42568	@narendramodi@MissPaswan But sure BJP win 100% Dam sure because youth is with modiji
PSerefine	@narendramodiBJP meeting at Kharu held under the Chairmanship of Sh. Dorje Angchuk District President Leh in presence of Sh. Chering Dorje Lakrook and other senior leaders. The party won
Simsy6_	@narendramodiAirstrike will help BJP win more than 22 seats in the state
SidRamone	@narendramodihoneymoon' to help BJP win #Karnataka by-polls
StormBring	@narendramodiBJP winning Barrackpur and Bangaon. Serampore and Hooghly in close contest.
Maurice46	@narendramodiThis is my dream too. But after BJP wins 500 seats of its own with 70% vote share in 2024.
dustinmille	@narendramodiAre EVM being manipulated to ensure that BJP wins this general elections?
SidRamone	@narendramodiMany more shocking truth might get revealed if BJP wins!!
Davemortc	@narendramodilf alliance does not happen (high possibility) and BJP wins all 7 seats(again
HarrisonSt	@narendramodidont get swayed away by any rhetoric around BJP winning big
ScobyLuft	@narendramodiBJP WINNING ALL 8 UP SAHARANAPUR TOUGH FIGHT BUT CONGRESSI AND OTHER CHOR FIGHT MADE EASY FOR LAKHANPAL JEE ABKI BAAR 400 KA PAAR BSP KA HAATHI NIKLA
Vaderlr	@narendramodil have made Amethi ready for a BJP win: @smritirani
Tedluuu2	@narendramodilf BJP wins EVM rigged.
LFC_Bern	@narendramodialthough helped BJP win by cutting AAP's vote.
Isaellis201	@narendramodilf BJP wins enough seats in WB and Orissa to compensate losses in UP and Maharashtra
Nomar5g	@narendramodi#MODI Sir if BJP wins by majority and more seats in Karnataka credit should go to Chakravarty Sulibele
Aicha6086t	@narendramodi#GoBackModi Air strike will help BJP win more than 22 Lok Sabha seats in Karnataka
OpossumS	@narendramodiDehradun And it will go back 3000 years if BJP wins the polls.
fordmoton	@narendramodiAnyone calling for voting NOTA is helping BJP win. Vote decisively to end this gunda Raj where NIA gives clean chit to murderers. And ruling party president misuses state mach
chevytruck	@narendramodilBJP winning in TV studios.
Rumholtz	@narendramodihe won't win and also the BJP . By chance BJP wins
Vitaman36	@narendramodiThe #BJP makes me hang my head in shame. These guys have no principles. They are crude. They are vulgar. Air strike will help BJP win more than 22 Lok Sabha seat
ABDHpodc	@narendramodilBJP winning Bengal
bjstoshor	@narendramodiSee the real face BJP wing and So - Called Hindu Organization VHP.
fkcgov	@narendramodiNow let's see how many seats will BJP win in 2024?
jimcolleat	@narendramodiIndia war to help BJP win upcoming polls.
CynnieMar	@narendramodiThe Indian Armed Forces proved their mettle by destroying the terrorists' hideouts . The impact of this strike will help BJP win more than 22 seats in the state

HoneyJ30292314	@RahulGandhiWhen they ask what has been done in 70 years, they do not insult Congress but our farmers and labourers: #RahulGandhi on #BJP
MarkSMundyIND	@RahulGandhi#GodIMedia #CongressParty
PainFreeLove	@RahulGandhi#congress wins
LewisBrandon239	@RahulGandhi#congress wins
Lolsonrealtor	@RahulGandhi#congress wins
yana_yudich	@RahulGandhi#lokcongress
elvincan2	@RahulGandhiCongress president
Complete_Dallas	@RahulGandhisaid they will work with each other and fight the upcoming polls together, according to party MLAs
ddk1	@RahulGandhiEventually the Congress Party will be left with frustrated deposit losers screaming and heckling other panelists on TV debates, due to relentless moral victories.
PhillyRepub1	@RahulGandhiHas Congress lost the Goa plot?
Erunner110	@RahulGandhi Since 2017, the Congress lost 15 of its 17 MLAs to defections. While the party's election campaign gathered steam since Dec, 2021, the Congress is still seeking to
HeyLady1776	@RahulGandhiCongress candidates on whether he would defect if the Congress loses elections.
J_K_Oh	@RahulGandhiPoll of Exit polls predicts #BJP & #Congress winning
girissky	@RahulGandhiBlack Day for Indian democracy if Congress wins
RobertNardone06	@RahulGa
TexasBr64310863	@RahulGandhi#VoteForCongress
thedemureshark	@RahulGandhi#VoteForCongress
Tammihazard2	@RahulGandhi#VoteForCongress
joelhthornton	@RahulGandhi#VoteForCongress
fishhunt247365	@RahulGandhiBJP confident of retaining power; Congress too hopeful of favourable verdict
Kadania1	@RahulGa
Boxer751	@RahulGandhi#AaRahiHaiCongress
mtbiker6975	@RahulGandhiCongress sends central observers ahead of results
authentically45	@RahulGandhiCongress in still hope to form the government
Count_Rollins	@RahulGandhi#Congress should start to take control over their people
jr_pettaman	@RahulGandhiCongress may win 12 to 16 seats, according to the ABP-C-Voter Exit Poll.
crel2112	@RahulGandhiCongress was projected to emerge as a distant second with victory in 12-17 seats
NOTASERF_VASSAL	@RahulGandhiCongress may retain power shows Ground Zero Research

Table.4.2:-Congress Dataset

To predict election results using Twitter data, the dataset would need to include the following types of information:

- Election Data:** This would include information about the election being studied, such as the date, location, and type of election (e.g. presidential, gubernatorial, or congressional).
- Candidate Data:** This would include information about the candidates running in the election, such as their names, political party affiliations, and demographic information (e.g. age, gender, ethnicity).
- Twitter Data:** This would include a collection of tweets related to the election, candidates, and issues. This data would need to include the text of the tweets, the usernames of the authors, the date and time of the

tweets, and any associated metadata such as hashtags, retweets, and likes.

- iv. **Sentiment Data:** This data would include sentiment analysis of the tweets related to the election, indicating whether each tweet was positive, negative, or neutral towards the candidates or issues.
- v. **Engagement Data:** This data would include information about the engagement level of each tweet, such as the number of retweets, likes, and replies.

The dataset should be large enough to train a machine learning model and include data from various sources such as different regions, political parties, and demographics to ensure the model's accuracy and robustness. Additionally, the dataset should be properly labelled and preprocessed to minimize noise and biases that may affect the prediction model.

Number of Instances:	970
Number of Attributes:	2
Attribute Characteristics:	Text, Real
Missing Values:	No

Table.4.3:- Attributes

## 4.2-Results and Test Analysis

### 4.2.1-Results

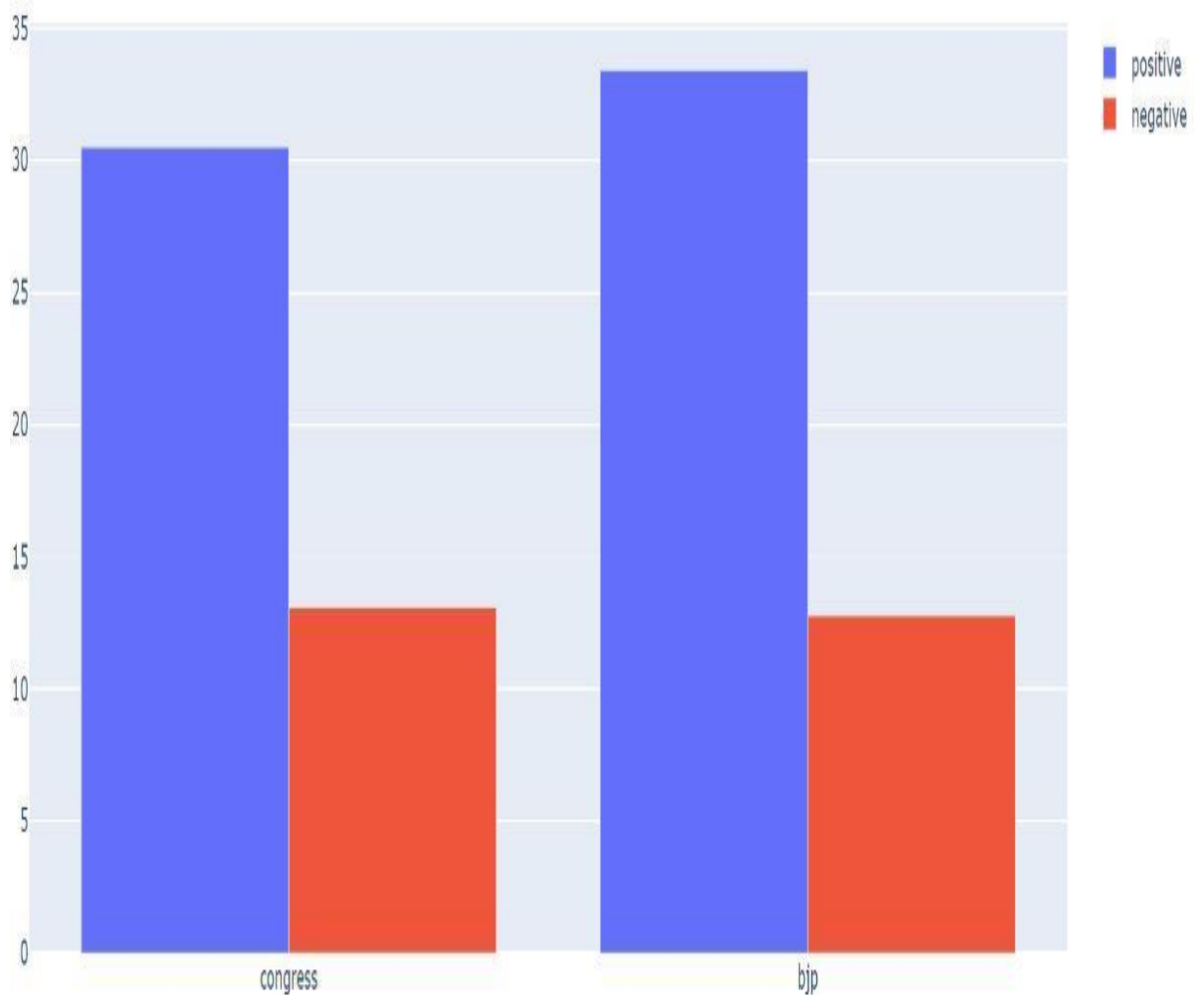


Fig.4.1:- Tweets classification based on polarity

Blue color bar indicates the positive votes (tweets) given by users for the particular political party and the Red color bar indicates the negative votes(tweets) given by users for the particular political party.

Positive polarity acquired by Congress is around 31% and negative polarity is around 13%.

Positive polarity acquired by BJP is around 34% and negative polarity is around 12%.



## Public Sentiment

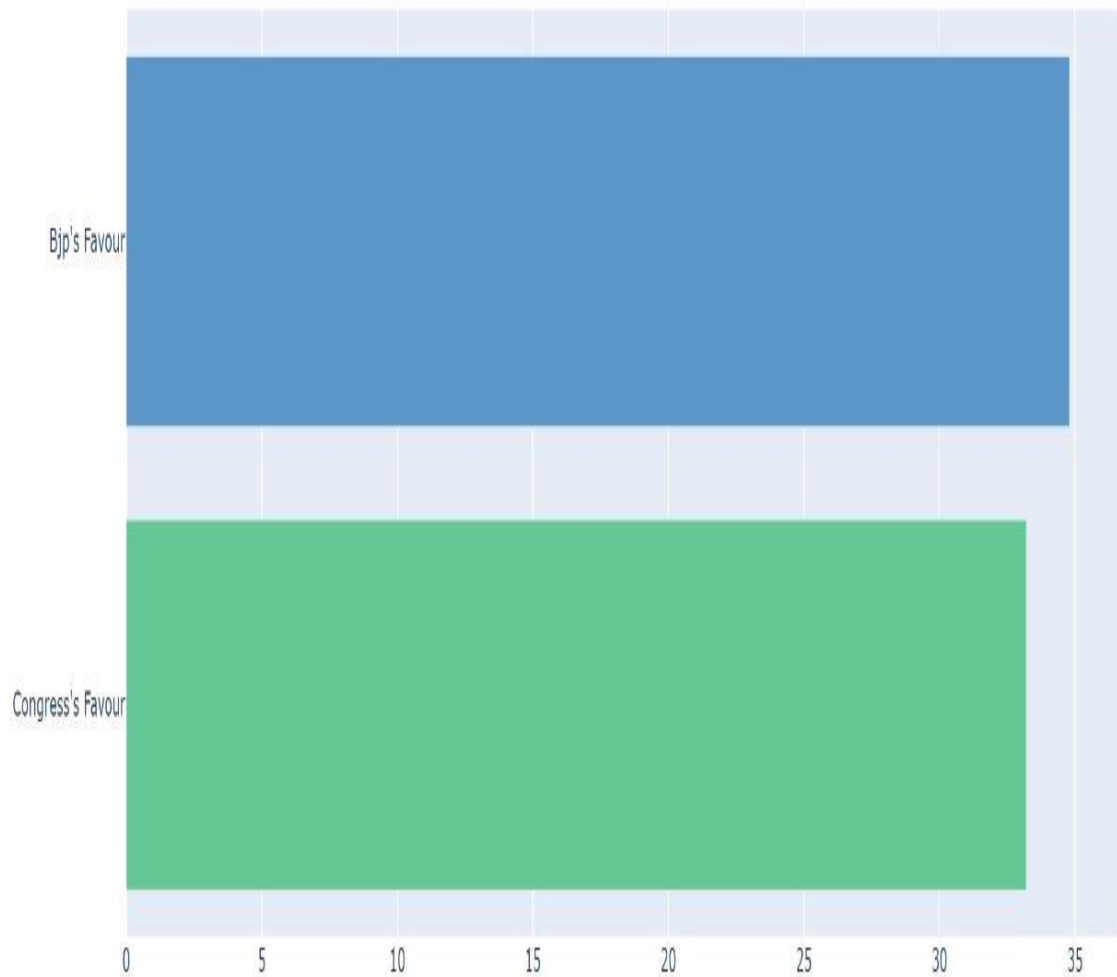


Fig.4.2:- Overall Results through sentiment analysis

This public sentiment analysis for both BJP and Congress was calculated using the algorithm: positive polarity(tweet) of one party summing up of negative polarity(tweet).

Public sentiment for BJP is around 35%

Public sentiment for Congress is around 32%.

#### 4.2.2-Test Analysis

Analysis for congress

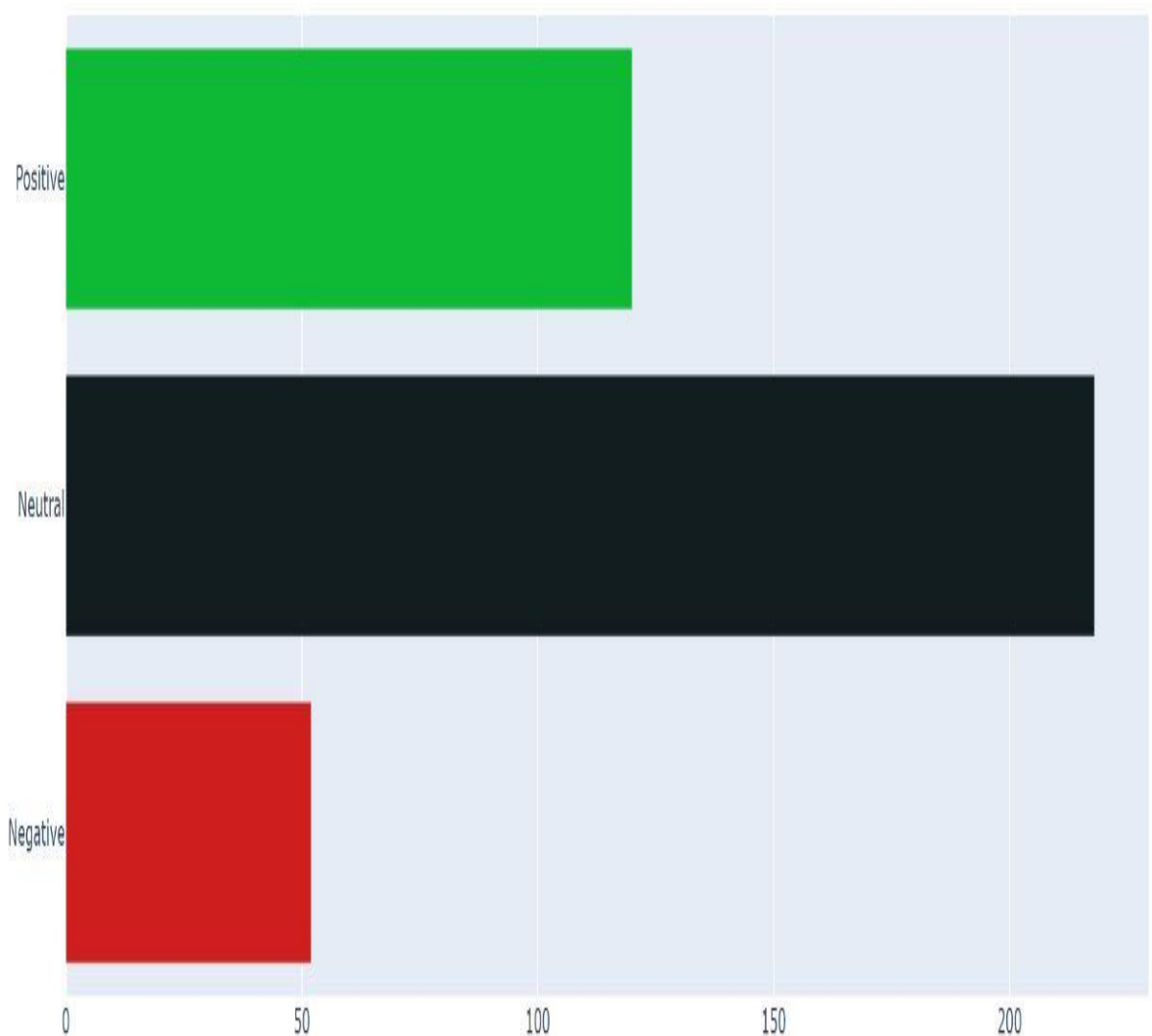


Fig.4.3:-Sentiment Analysis for Congress Dataset

Polarity is classified into 3 sectors namely Positive, Negative and Neutral.

Positive polarity (tweets) acquired by Congress is around 120.

Neutral polarity (tweets) acquired by Congress is around 220.

Negative polarity (tweets) acquired by Congress is around 52.

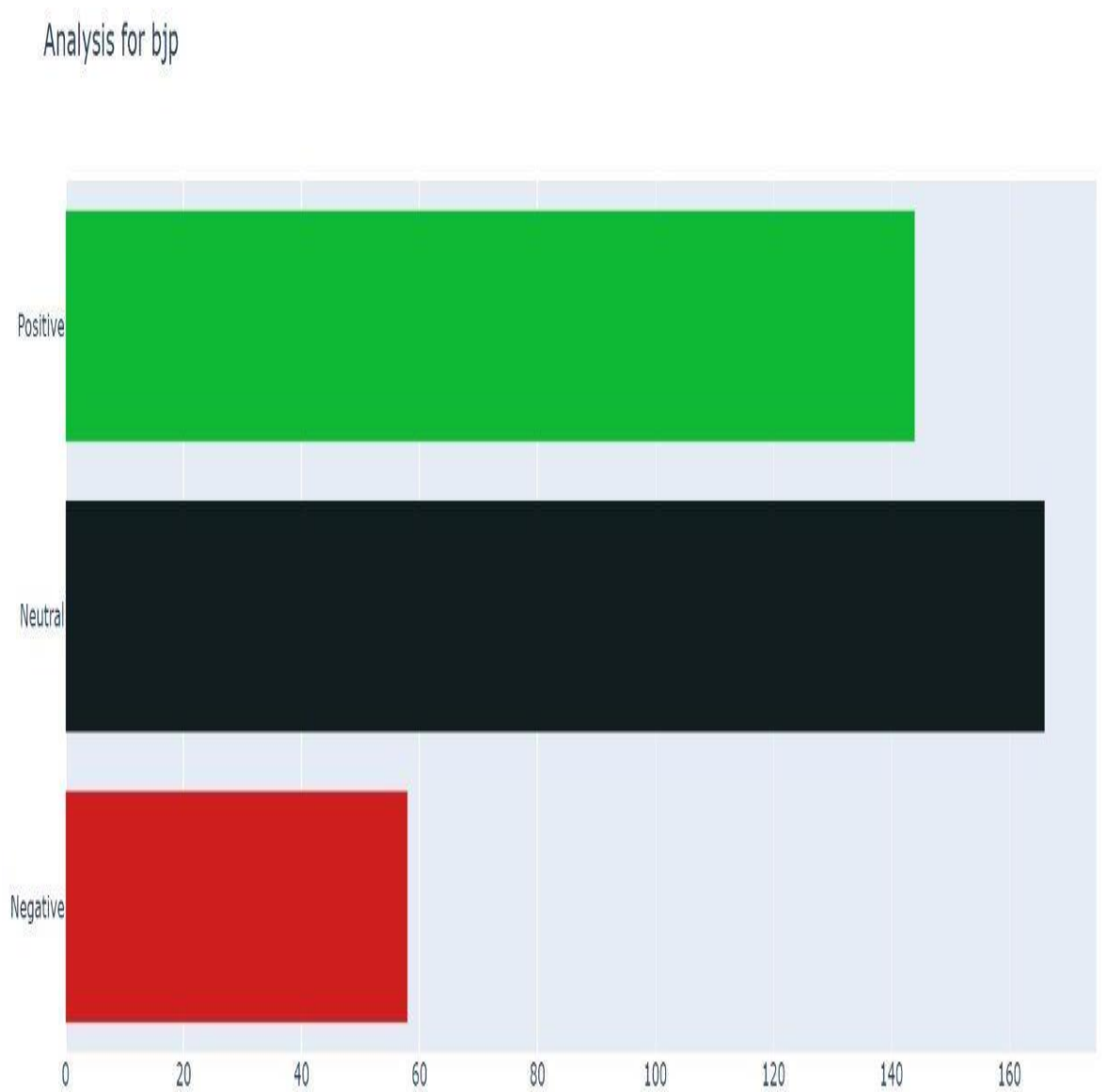


Fig.4.4:- Sentiment Analysis for BJP Dataset

Polarity is classified into 3 sectors namely Positive, Negative and Neutral.

Positive polarity (tweets) acquired by BJP is around 145.

Neutral polarity (tweets) acquired by BJP is around 170.

Negative polarity (tweets) acquired by BJP is around 58.

## **5.Summary and Future Scope**

In summary, election results prediction using Twitter data is an exciting application of machine learning that has the potential to improve the accuracy of election forecasts. By analyzing Twitter data related to the election, it is possible to predict the outcome of the election with a certain degree of accuracy.

However, the success of the model depends on several factors, including the quality and quantity of the data, the accuracy of the sentiment analysis, and the features selected for the model. Additionally, the model may be affected by biases and errors in the data, as well as external factors that are not captured in the Twitter data. Election results prediction using Twitter data is a promising area of research that has the potential to improve our understanding of the election process and enhance the accuracy of election forecasts.

In terms of future scope, there is significant potential for further research and development in this field. For example, the model could be refined to take into account the geographic location of the Twitter users, as well as the time and date of the tweets. Additionally, the model could be improved by incorporating data from other social media platforms and news sources.

Another potential area of future research is the development of models that can predict not only the outcome of the election but also the margins and trends of the results. This could provide valuable insights into the voting behaviour and preferences of the electorate, as well as the effectiveness of the campaign strategies used by the candidates.

Future research in this area can focus on several directions. First, the development of more sophisticated machine learning algorithms and deep learning models can help improve the accuracy and robustness of election prediction models and also inclusion of spam detection algorithms for erasing the irrelevant data.

Overall, election result prediction using Twitter data has the potential to revolutionize the way we monitor and understand election outcomes. As social media platforms continue to play an increasingly important role in shaping public opinion and political discourse, the need for accurate and reliable election prediction tools will only continue to grow.

## REFERENCES

[1] Meng-Hsiu Tsai Yingfeng Wang Myungjae Kwak Neil Rigole(2019).

“A Machine Learning Based Strategy for Election Result Prediction”.  
2019 International Conference on Computational Science and Computational Intelligence (CSCI).

[2] Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with Twitter: What 140 characters reveal about political sentiment. ICWSM, 10, 178-185.  
<https://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/viewFile/1441/1852>

[3] H. Wang, D. Can, A. Kazemzadeh, F. Bar and S. Narayanan, "A System for real-time Twitter sentiment analysis of 2012 U.S. presidential election cycle", Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp. 115-120, 2012.

[4] 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, pp. 122-129, 2010.

[5] S. Perez, Twitter's doubling of character count from 140 to 280 had little impact on length of tweets, 2018, [online] Available: <https://techcmnch.com/2018/10/30/twitters-doubling-of-character-count-from-140-to-280-had-little-impact-on-length-of-tweets/>.

[6] F. Nausheen and S.H. Begum, "Sentiment analysis to predict election results using Python", Proceedings of the Second International Conference on Inventive Systems and Control, pp. 1259-1262, 2018.

Natural Language Toolkit, [online] Available: <http://www.nltk.org/>.

[7] J. Ramteke, S. Shah, D. Godhia and A. Shaikh, "Election result prediction using Twitter sentiment analysis", Proceedings of the International Conference on Inventive Computation Technologies, 2016.

[8] W. Budiharto and M. Meiliana, "Prediction and analysis of Indonesia Presidential election from Twitter using sentiment analysis", J. Big Data, vol. 5, no. 1, pp. 1-10, 2018.

[9] Jungherr, A., Jürgens, P., & Schoen, H. (2012). Why the Pirate Party Won the German Election of 2009 or the Trouble With Predictions: A Response to Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2011). 'Predicting Elections With Twitter: What 140 Characters Reveal About Political Sentiment.' Social Science Computer Review, 30(2), 229-234.

[10] Gayo-Avello D, Metaxas, P.T, & Mustafaraj, E. (2011). Limits of electoral predictions using Twitter. ICWSM, 11, 122-129.

[11] Chakraborty S, & Ganguly N. (2020). Analyzing Twitter data to predict the outcome of Indian General Elections 2019. Social Network Analysis and Mining, 10(1), 1-19.

## APPENDIX

### Open Access Link

[https://github.com/Kani111/Main\\_Project.git](https://github.com/Kani111/Main_Project.git)  
[https://github.com/KrishnaChaitanya28/Final\\_Project](https://github.com/KrishnaChaitanya28/Final_Project)

### Code

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from textblob import TextBlob
from wordcloud import WordCloud
import plotly.graph_objects as go
from collections import defaultdict
congress=pd.read_csv('/content/rahulgandhi(1).csv')
bjp=pd.read_csv('/content/modi(1).csv')
print(congress)
print(bjp)
congress.info()
bjp.info()
congress.head()
bjp.head()
congress['Text'][150]
TextBlob(congress['Text'][150]).sentiment
bjp['Text'][250]
TextBlob(bjp['Text'][250]).sentiment
def polarity(review):
    return TextBlob(review).sentiment.polarity
congress['polarity']=congress['Text'].apply(polarity)
bjp['polarity']=bjp['Text'].apply(polarity)
congress.head()
```



```

congress['Expression']=np.where(congress['polarity']>0,'Positive','Ne
gative')
congress.loc[congress.polarity == 0,'Expression'] = 'Neutral'
congress.head()

bjp['Expression']=np.where(bjp['polarity']>0,'Positive','Negative')
bjp.loc[bjp.polarity == 0,'Expression'] = 'Neutral'
bjp.head()

def exp_graph(reviews,title):
    group=reviews.groupby('Expression').count()
    pol_count=list(group['polarity'])
    exp=list(group.index)

    group_list=list(zip(pol_count,exp))
    df=pd.DataFrame(group_list,columns=['pol_count','exp'])

    df['color']='rgb(14,185,54)'
    df.loc[df.exp=='Neutral','color']='rgb(18,29,31)'
    df.loc[df.exp=='Negative','color']='rgb(206,31,31)'

    go.Figure(go.Bar(x=df['pol_count'],
                     y=df['exp'],orientation='h',

marker={ 'color':df['color']})).update_layout(title_text=title).show()

exp_graph(congress,'Analysis for congress')
exp_graph(bjp,'Analysis for bjp')
congress[congress['polarity']==0].shape
bjp[bjp['polarity']==0].shape
congress[congress['polarity']<0].shape
bjp[bjp['polarity']<0].shape
congress[congress['polarity']>0].shape
bjp[bjp['polarity']>0].shape
congress.drop((congress[congress['polarity']==0]).index,inplace=True

```

```

e)

print(congress.shape)
bjp.drop((bjp[bjp['polarity']==0]).index,inplace=True)
print(bjp.shape)
def balanced_data(reviews,n):
    np.random.seed(10)
    drop=np.random.choice(reviews.index,n,replace=False)
    review_subset=reviews.drop(drop)
    return review_subset
congress_subset=balanced_data(congress,2)
print(congress_subset.shape)
bjp_subset=balanced_data(bjp,32)
print(bjp_subset.shape)
sns.distplot(congress_subset['polarity'])
sns.distplot(bjp_subset['polarity'])
congress_subset.groupby('Expression').count()
bjp_subset.groupby('Expression').count()
def pol_percent(subset,total):

    neg_percent=((subset.groupby('Expression').count())['polarity'][0]/total)*100

    pos_percent=((subset.groupby('Expression').count())['polarity'][1]/total)*100

    return neg_percent,pos_percent
congress_pol_percent=pol_percent(congress_subset,390)
print(congress_pol_percent)

bjp_pol_percent=pol_percent(bjp_subset,368)
print(bjp_pol_percent)
candidate=['congress','bjp']
pos=[congress_pol_percent[1],bjp_pol_percent[1]]

```

```

neg=[congress_pol_percent[0],bjp_pol_percent[0]]

go.Figure(data=[
    go.Bar(name='positive',x=candidate,y=pos),
    go.Bar(name='negative',x=candidate,y=neg),
])

congress_total_percent=pol_percent(congress_subset,500)
congress_total_percent
bjp_total_percent=pol_percent(bjp_subset,500)
bjp_total_percent
congress_pos=congress_total_percent[1]+bjp_total_percent[0]
print(congress_pos)
bjp_pos=congress_total_percent[0]+bjp_total_percent[1]
print(bjp_pos)
go.Figure(go.Bar(y=['Congress\'s Favour', 'Bjp\'s Favour'],
    x=[congress_pos,bjp_pos],
    marker={'color':['rgb(100,200,150)','rgb(90,150,200)']}},
    orientation='h')).update_layout(title_text='Public
Sentiment')
def pos_tweets(tweets,title):
    most_pos_text=tweets[tweets['polarity']==0.5].Text.head()
    most_pos_pol=tweets[tweets['polarity']==0.5].polarity.head()

    go.Figure(data=[go.Table(columnorder=[1,2],
        columnwidth=[10,100],
        header={'values':['polarity',title],'align':'left'},

        cells={'values':[most_pos_pol,most_pos_text],'align':'left'
        })).update_layout(title_text=title).show()
pos_tweets(congress_subset,'positive tweets for congress')
pos_tweets(bjp_subset,'positive tweets for bjp')
def neg_tweets(tweets,title):

```

```

most_neg_text=tweets[tweets['polarity']==-1].Text.head()
most_neg_pol=tweets[tweets['polarity']==-1].polarity.head()

go.Figure(data=[go.Table(columnorder=[1,2],
                        columnwidth=[10,100],
                        header={'values':['polarity',title],'align':'left'},

cells={'values':[most_neg_pol,most_neg_text],'align':'left'}
      )]).update_layout(title_text=title).show()
neg_tweets(congress_subset,'negative tweets for congress')
neg_tweets(bjp_subset,'negative tweets for bjp')
def wordcloud(data,title):
    Text=str(data.Text)

wc=WordCloud(max_font_size=100,max_words=500,scale=10,relative_scaling=0.6,background_color='white').generate(Text)

plt.figure(figsize=(15,10))
plt.title(title,{'fontsize':30,'family':'serif'})
plt.axis('off')
plt.imshow(wc)
wordcloud(congress_subset,'wordcloud for congress tweet replies')

wordcloud(bjp_subset,'wordcloud for bjp tweet replies')

```

# Election Results Prediction Using Twitter Data

Dr. M.Neelakantappa  
Information Technology  
Vasavi College of Engineering  
(Osmania University)  
Hyderabad , India  
m.neelakanta@gmail.com

B. Sri Kanishka Reddy  
Information Technology  
Vasavi College of Engineering  
(Osmania University)  
Hyderabad , India  
Kanishkareddy@gmail.com

A. Krishna Chaitanya  
Information Technology  
Vasavi College of Engineering  
(Osmania University)  
Hyderabad , India  
krishnachaitanya@gmail.com

**Abstract**— With the ability to predict political outcomes and provide insights into public opinion, the use of Twitter data to predict election results has gained popularity. Twitter offers a massive supply of data for analysis due to its enormous user base and real-time nature. To categorize tweets as good, negative, or neutral and to follow sentiment patterns over time, researchers use sentiment analysis tools. Network analysis finds influential users and digs deeper into the dynamics of political discourse. The accuracy of predictions is improved by combining traditional polling data with machine learning methods. Twitter data analysis has the potential to offer insightful information for election campaigns and improve political strategies, despite issues like representativeness and identifying genuine sentiment.. Ongoing research focuses on refining methodologies and addressing limitations, advancing the reliability of election prediction using Twitter data. (*Abstract*)

**Keywords**— *Twitter, election result prediction, recursive neural tensor net- work, natural language processing (key words)*

## I. INTRODUCTION

Election forecasting is a challenging task, and reliable predictions are essential for both political campaigns and election analysts. Massive volumes of user-generated data about political candidates and campaigns have been produced as a result of the rise of social media platforms like Twitter. As such, social media data provides an unprecedented opportunity to extract insights and make accurate predictions about election results. In this context, this study proposes a machine learning-based approach for predicting election results using Twitter data. The proposed methodology involves collecting and pre-processing a large volume of tweets related to the election. The study also addresses the issue of plagiarism in Twitter data by employing a plagiarism detection algorithm to ensure that the tweets used in the analysis are original. This study demonstrates the potential of using social media data for election prediction and highlights the importance of incorporating relevant features and selecting appropriate machine learning algorithms for accurate prediction. The rise of social media sites like Twitter has produced a wealth of user-generated data that offers insightful information on a variety of subjects, including politics. The vast amount of data generated on Twitter during an election season presents a unique opportunity for researchers to analyse and predict the election outcomes accurately. However, the authenticity of Twitter data has always been a significant concern, as it is prone to fake news and plagiarism. This approach involves collecting a large volume of tweets related to the election and pre-processing the data to remove irrelevant information and duplicate tweets. Feature extraction techniques such as sentiment analysis, user influence analysis, and topic

modelling are then used to extract valuable information from the tweets . This study addresses the problem of plagiarism in Twitter data while demonstrating the possibilities of employing social media data for election prediction. By incorporating relevant features and selecting appropriate machine learning algorithms, this approach can help in making accurate election predictions.

In this paper we present a study for election results prediction using twitter data through strategy sentiment analysis. Here we collected data related to both the parties Bhartiya Janatha Party and National Congress party using twitter handle. After that data preparation is done i.e data cleaning and data pre-processing and this data is used for the sentiment analysis. Following data separation, it is loaded as a.csv file into a jupyter notebook or Google Collaboratory. It is being trained using the RNTN algorithm, which gives each tweet or message a polarity. These polarity are broken down into three groups: positive, negative, and neutral. Positive polarity is implied by (>0), neutrality by (=0), and negative polarity by (0).

piyush23	@narendramodidistracted all TV channels from Masood Ashraf (China) fiasco. Today one channel markets #Bajajnotproof. Others have polls showing Modi/BJP win.
cat_42568	@narendramodidistracted @MisPawen But sure BJP win 100% Dam sure because youth is with modi
Plsefrine	@narendramodidistracted @Kharu held under the Chairmanship of Sh. Dorje Angchuk District President Leh in presence of Sh. Chering Dorje Lakruk and other senior leaders. The party was
Simps6	@narendramodidistracted will help BJP win more than 22 seats in the state
StidRamore	@narendramodidistracted to help BJP win Karnataka by polls
StormBrng	@narendramodidistracted winning Barrackpore and Bangan. Serampore and Hooghly in close contest.
Maurice46	@narendramodidistracted This is my dream too. But after BJP wins 500 seats of its own with 70% vote share in 2024.
dustinmike	@narendramodidistracted EVM being manipulated to ensure that BJP wins this general elections?
StidRamore	@narendramodidistracted Many more shocking truth might get revealed if BJP wins!!
Davemore	@narendramodidistracted alliance does not happen (high possibility) and BJP wins all 7 seats again
HarrisonSh	@narendramodidistracted I got swayed away by any rhetoric around BJP winning big
Scotbyluff	@narendramodidistracted WINNING ALL 8 UP SAHABANAPUR TOUGH FIGHT BUT CONGRESS AND OTHER CHOR FIGHT MADE EASY FOR LAKHANPAL JEE ABKI BAAR AOO KA PAAR BJP KA HAATHI NIKLA
Vadifer	@narendramodidistracted have made Amethi ready for a BJP win: @smritirani
Tedlouu2	@narendramodidistracted BJP wins EVM rigged.
LFC_Bern	@narendramodidistracted helped BJP win by cutting AAP's vote.
Isaell201	@narendramodidistracted BJP wins enough seats in WB and Orissa to compensate losses in UP and Maharashtra
NomaSg	@narendramodidistracted MODI Sir if BJP wins by majority and more seats in Karnataka credit should go to Chakravarty Sultbele
Aicha6088	@narendramodidistracted Modi Air strike in J&K will help BJP win more than 22 Lok Sabha seats in Karnataka
Opomum5	@narendramodidistracted And it will go back 3000 years if BJP wins the polls.
fordmotor	@narendramodidistracted anyone calling for voting NDTA is helping BJP win. Vote decisively to end this gunda Raj where NIA gives clean chit to murderers. And ruling party president misuses state mach
chevytruck	@narendramodidistracted BJP winning in TV studios.
Rumholtz	@narendramodidistracted won't win and also the BJP . By chance BJP wins
Vitamas36	@narendramodidistracted The #BJP makes me hang my head in shame. These guys have no principles. They are crude. They are vulgar. Aaj-Air strike in J&K will help BJP win more than 22 Lok Sabha seats
ABDhooq	@narendramodidistracted winning Bengal
bjtoshor	@narendramodidistracted See the real face BJP wing and So - Called Hindu Organization VHP.
Ricagov	@narendramodidistracted Now let's see how many seats will BJP win in 2024?
jimcolleat	@narendramodidistracted India war to help BJP win upcoming polls.
CynielMar	@narendramodidistracted The Indian Armed Forces proved their mettle by destroying the terrorists' hideouts. The impact of this strike will help BJP win more than 22 seats in the state

Table.1 :-Shows the .csv files of bjp dataset.

## II. LITERATURE SURVEY

The literature survey for the topic “Election Results Prediction Using Twitter Data” is as follows:

[1] N. Gupta, P. Kumaraguru, and others (2012). This study focuses on developing a credibility ranking algorithm for tweets during major events like elections. The algorithm considers various factors, including user reputation, retweet count, and sentiment analysis, to rank

the credibility of tweets and improve the accuracy of election result predictions.

[2] Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with Twitter: What does the characters reveals us about the political sentiment. In Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media Access Here

This research investigates the power of Twitter data for election results prediction. The study employs sentimental analysis for analyzing the political tweets and gives a demonstration on the relation between Twitter sentiment and election outcomes, providing insights into the potential of Twitter data for predicting election results.

[3] Social Science Computer Review, 30(2), 229–234; Jungherr, A., Jürgens, P., & Schoen, H. (2012). "Why the pirate party won the German election of 2009 or the trouble with predictions: A response to Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010)."

This article critically examines the study by Tumasjan et al. (2010) on predicting elections with Twitter sentiment analysis. It discusses the limitations and challenges in using Twitter data for election predictions, emphasizing the need to consider contextual factors and the potential biases in analyzing political sentiment on social media platforms.

[4] Diakopoulos, N. A., & Shamma, D. A. (2010). Characterizing debate performance via aggregated twitter sentiment. In Proceedings of the ACM 2010 conference on Computer Supported Cooperative Work (pp. 119-122). ACM. Access Here

This study explores the use of aggregated Twitter sentiment to characterize the performance of political candidates during debates. The research analyzes tweet sentiment and topic distributions to gain insights into public perception, providing a novel approach to assessing candidate performance and potentially predicting election outcomes.

[5] Gayo-Avello, D. (2012). No, you cannot predict elections with Twitter. IEEE Internet Computing, 16(6), 91-94. Access Here

This article challenges the notion that Twitter can reliably predict election results. It discusses the limitations of using Twitter data, such as biases, lack of representativeness, and the difficulty of distinguishing genuine sentiment from noise. The study cautions against over-reliance on Twitter predictions and highlights the need for cautious interpretation.

[6] Conover, M. D., Ratkiewicz, J., & Menczer, F. (2011). Political polarization on Twitter. In the Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (pp. 89-96). Access Here

This research examines political polarization on Twitter by analyzing the ideological positions of users and their interactions. The study highlights the impact of polarization and discusses its implications for election predictions based on Twitter data.

[7] González-Bailón, S., & Moreno, Y. (2012). Broadcasters and their hidden influentials in online protest diffusion. American Behavioral Scientist, 56(3), 260-279. Access Here

This study investigates the role of influential users in online protest diffusion, which can be relevant for election-related discussions on Twitter. It explores the network dynamics and the potential influence of users with high centrality, shedding light on the mechanisms of information spread during political events.

[8] Jungherr, A., & Schoen, H. (2012). Twitter and its mediation the politics : An analysing the messages posted during the campaign for the German federal election 2009. Journal for Computer-Mediated Communication, 17(2), 91-108. Access Here

This research examines the role of Twitter in mediating political communication during the campaign for the German federal election. The study analyzes the content and sentiment of tweets and highlights the ways in which Twitter functions as a platform for political discourse and information dissemination.

[9] Barberá, P., & Rivero, G. Twitter users and their Understanding of the political representativeness . Social Science Computer Review, 32(2), 153-163.

This article assesses the political perceptiveness of Twitter users by comparing their demographics with those of the general population. The study highlights the potential biases and limitations of using Twitter data for political analysis and emphasizes the need to consider the characteristics of the user population.

[10] Zhao, W. X., Jiang & Li, X. (2011). Comparing Twitter and traditional media using topic models. In European Conference of Information Retrieval. Springer. Access Here

This research compares Twitter with traditional media by applying topic models to analyze the content shared on both platforms. The study highlights the differences and similarities in topics discussed, shedding

light on the unique characteristics and potential value of Twitter data for election-related analysis.

[11] Skoric, M. M., Ying, T., & Wictor, I. (2016). Public opinion formation in a hybrid media environment: A big data analysis of Twitter responses to the 2012 US presidential debates. *Journal of Communication*, 66(4), 639-659. Access Here

This study examines public opinion formation during the 2012 US presidential debates by analyzing Twitter responses. The research utilizes big data analysis techniques to explore sentiment dynamics and topics discussed on Twitter, providing insights into the role of Twitter in shaping public opinion during electoral events.

[12] Metaxas, P. T., Mustafaraj How (not) to predict elections. In *Proceedings of the ACM Web Science Conference* (pp. 172-181). ACM. Access Here

This research examines the challenges and limitations of predicting elections using social media data, including Twitter. The study discusses various factors, such as biases, sample representativeness, and the dynamic nature of social media platforms. It provides insights into the complexities of election prediction using Twitter data.

They cover topics such as sentiment analysis, network dynamics, user influence, biases, and limitations associated with using Twitter data for predicting election outcomes.

### III. PROPOSED METHODOLOGY

#### Proposed Methodology for Election Results Prediction Algorithm Using Twitter Data

**Data Collection:** Gather a comprehensive dataset of election-related tweets using specific keywords and hashtags. Utilize the Twitter API or third-party tools to retrieve a diverse and representative sample of tweets.

**Data Pre-processing:** Clean the collected data by removing retweets, duplicates, and non-English tweets. Perform text pre-processing tasks such as tokenization, removing stop words, and normalizing words to prepare the data for analysis.

**Sentiment Analysis:** Apply sentiment analysis techniques for classifying each tweet as a biased tweet or unbiased tweet. This can be achieved through machine learning models, lexicon-based approaches, or deep

learning methods, capturing the sentiment expressed in the tweets.

**Feature Extraction:** Extract relevant features from the pre-processed tweets, including user-related attributes (followers, verified status), tweet characteristics (retweet count, favourite count), and temporal information (time of posting). Additionally, consider topic modeling techniques to identify prevalent themes in the dataset.

**Iteration and Refinement:** Analyze the prediction results and identify any shortcomings or limitations. Refine the methodology by considering alternative feature selection methods, sentiment analysis techniques, or model architectures

**Validation :** Validate the prediction algorithm against actual election results to assess its effectiveness.

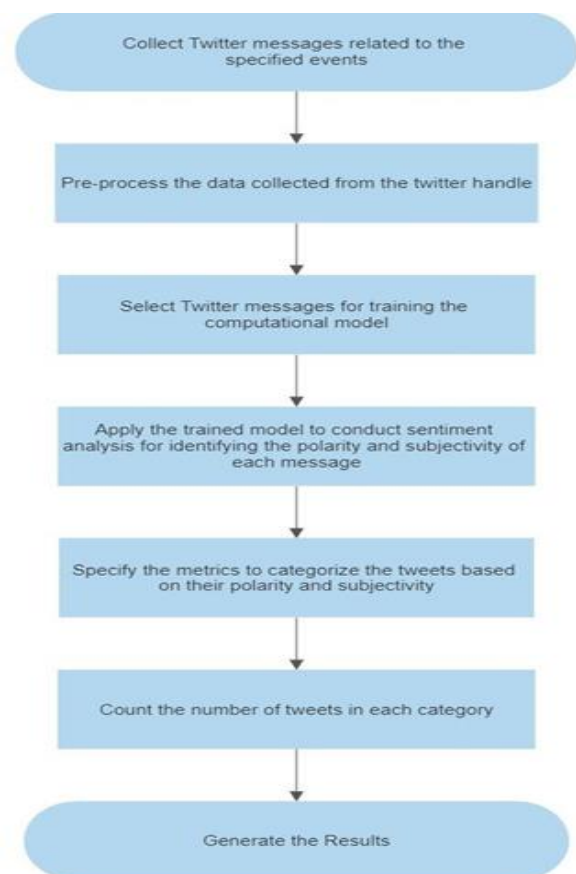


Fig.1 :- Flowchart

Figure 1 shows workflow of the proposed methodology that involves collecting and pre-processing the twitter data and train through the computational model by specifying certain metrics and counting the tweets according to the category and generating the results.

This method aims to leverage the vast amount of data generated on Twitter during an election season to make accurate predictions about the election outcome. By using relevant features and appropriate machine learning

algorithms, this method can help in making reliable predictions. The evaluation step ensures the quality of the predictions, which can be used by political analysts and pollsters to gain insights into the election outcome.

#### IV. FUNCTIONAL COMPONENTS

##### 1.NLP

In the context of election results prediction using Twitter data, NLP algorithms can be utilized to identify patterns and sentiments in tweets related to political candidates and issues. By analysing tweets related to political events, candidates, and issues, NLP techniques such as sentiment analysis and topic modelling can be used to identify trends and patterns in voter behaviour. With the help of machine learning algorithms, historical election data and Twitter data can be analysed to train NLP models for making predictions about future election outcomes based on the sentiment and topics identified in the Twitter data. This can provide useful feedbacks for the topics that are most important to voters and help political analysts and campaign strategists craft their messaging accordingly. NLP algorithms are capable of handling large amounts of data quickly and accurately. During an election campaign, the sentiment of tweets can change rapidly in response to current events and debates. NLP algorithms can quickly analyze new data and adjust their predictions accordingly, making them an indispensable tool for those interested in predicting election outcomes. In addition, Twitter data can be biased and may not be representative of the population as a whole. Therefore, it is necessary to validate the predictions made by NLP models with other sources of data to ensure their accuracy. NLP can be a useful tool for predicting election results, but it should be used in conjunction with other methods to achieve the most accurate predictions possible. Overall, NLP is a powerful tool for analysing and interpreting natural language data and has great potential in predicting election results using Twitter data. As the technology continues to improve and more data becomes available, NLP algorithms will become an even more valuable tool for political analysts and campaign strategists looking to gain insights into public opinion and predict.

##### 2.RNTN

The Recursive Neural Tensor Network (RNTN) is a deep learning model commonly used in NLP, including sentimental analysis. In the context of election results prediction using Twitter data, the RNTN can be applied to analyze the sentiment expressed in tweets and understand the overall sentiment towards political candidates or parties.

The RNTN model comprises three main components: word embeddings, compositional functions, and a tensor layer. Word embeddings represent words as continuous vectors that capture their semantic meaning. Compositional functions recursively combine these word embeddings to form higher-level representations of phrases and sentences. The tensor layer performs a tensor-based operation to capture

interactions between compositional representations and generate a sentiment score for the entire tweet or sentence.

To train the RNTN model for election results prediction, a labelled dataset of tweets with sentiment labels (positive, negative, or neutral) is required. The model is trained using optimization techniques such as backpropagation, which adjust the model's parameters to minimize the prediction error and learn the sentiment patterns in the data.

Once trained, the RNTN model can be applied to depict the sentiments of new, unseen messages related to the election. The sentiment predictions can then be aggregated to estimate the overall sentiment towards candidates or parties, providing insights into potential election outcomes. It's important to consider that the effectiveness of the RNTN model for election results prediction using Twitter data depends on various factors, including the quality of the training data, appropriate data pre-processing techniques, feature engineering, and the integration of additional information sources.

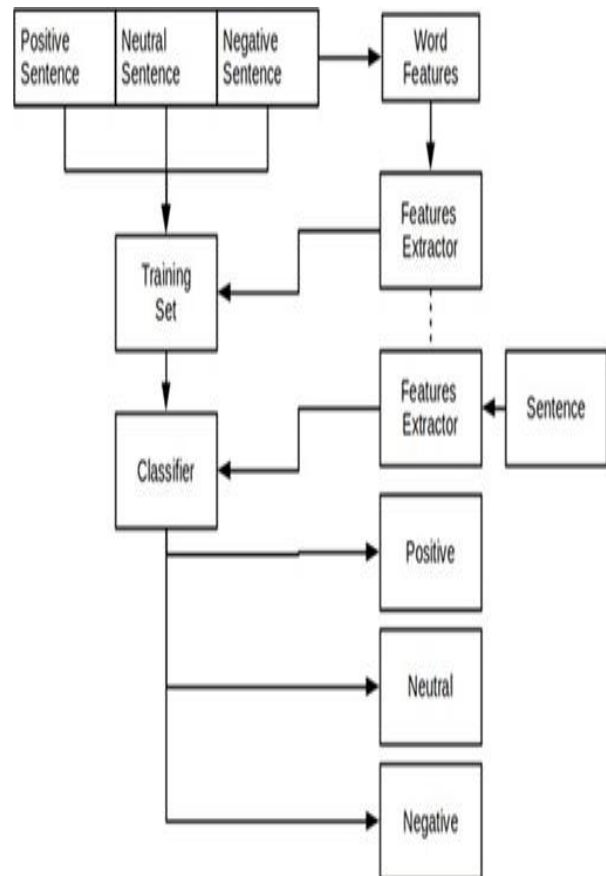


Fig.2:- RNTN Architecture

Figure 2 Shows the RNTN architecture consists of multiple layers of neural networks, each layer processing a specific aspect of the input data. The first layer processes the individual words in a sentence, while the subsequent layers analyze the structure of the sentence and its components. The RNTN architecture uses recursive neural networks to build a tree structure that represents the sentence, which can then be analyzed for sentiment or other natural language processing tasks.



### 3.TextBlob

TextBlob is a widely used Python library for (NLP) tasks, including sentimental analysis. It can be applied to election results prediction using Twitter data by analyzing the sentiment expressed in tweets and evaluating the public opinion towards political candidates or parties. To use TextBlob for sentiment analysis, the first step is to pre-process the Twitter data by erasing noise, such as links and special symbols, and tokenizing the text into individual words or sentences. TextBlob provides convenient functions for these pre-processing tasks. Once the data is prepared, sentiment analysis is performed using TextBlob's pre-trained sentiment classifier. The sentiment polarity score ranges between (-ve)1 to (+ve)1, where values that are close to -1 represent (-ve) sentiment, values that are near to 1 indicate (+ve) sentiment, and values indicating 0 signify neutral sentiment. By aggregating these sentiment scores across tweets, an overall sentiment towards candidates or parties can be estimated, providing insights for election results prediction. TextBlob also offers additional features like speech tagging, phrase extraction, and language translation, which can further enhance the analysis of Twitter data for election prediction. These features enable a more comprehensive understanding of the text and facilitate the identification of key topics and sentiments expressed by Twitter users. It's important to consider that TextBlob's sentiment analysis is based on a general-purpose sentiment classifier, which may not be specifically trained for political sentiment. Therefore, its effectiveness in predicting election results relies on the availability of a diverse and representative training dataset and the careful consideration of any biases or limitations associated with the sentiment classifier. In summary, TextBlob is a powerful Python library for sentiment analysis and other NLP tasks. When applied to election results prediction using Twitter data, it offers a straightforward approach to analyze sentiment, estimate overall sentiment towards candidates or parties, and gain insights into public opinion during elections.

### 4.Polarity and subjectivity

Polarity and subjectivity are essential aspects of sentiment analysis applied in election results prediction using Twitter data. Polarity refers to the sentiment expressed in a text, whether it is positively biased, negatively biased, or unbiased. It allows the quantification of sentiment towards political candidates or parties based on tweets collected from Twitter. Subjectivity measures the degree of subjectiveness in a text, indicating the extent to which personal opinions, beliefs, or biases influence the sentiment. It is represented as a score ranging from 0 to 1, with 0 indicating objectivity and 1 representing high subjectivity. Subjectivity analysis helps in understanding the level of personal judgment or bias in tweets related to elections. Using polarity as a metric in sentiment analysis can

help identify patterns and trends in public sentiment towards political candidates and issues.

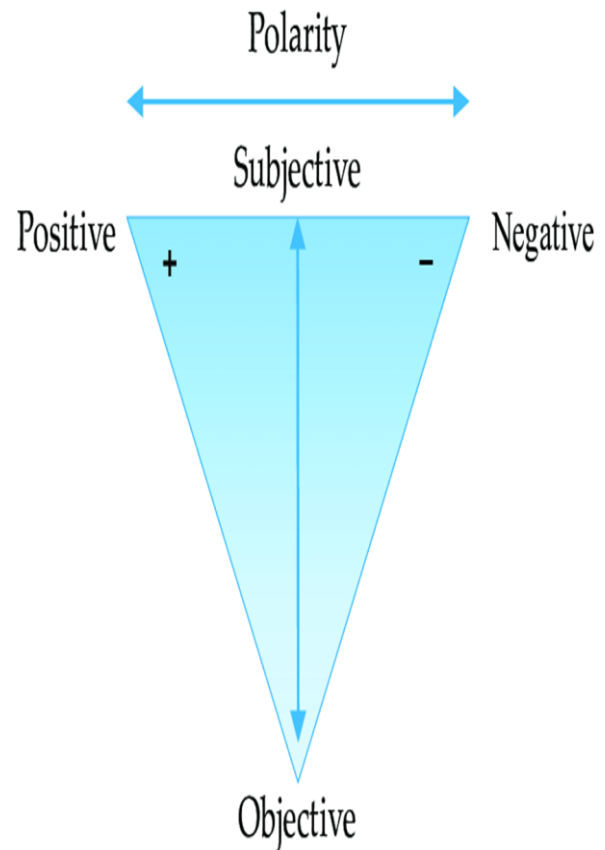


Fig.3:- Polarity and subjectivity

As shown in Fig 3 Subjectivity is a metric used in sentiment analysis to determine the degree to which a piece of text expresses an opinion rather than a fact. In the context of predicting election results using Twitter data, subjectivity is a valuable measure because it helps to identify the tone and context of tweets related to a particular candidate or issue.

By analyzing the subjectivity of a large amount of tweets related to a specific candidate or issue, politics analysts and campaign strategists can gain insights into the public's opinions and attitudes towards that candidate or issue. This can help them to predict the outcome of an election and make more informed decisions about how to respond to public opinion.

For example, if there is a significant increase in subjectivity in tweets about a particular candidate in the days leading up to an election, this could indicate a shift in public opinion and potentially predict the outcome of the election. By combining subjectivity with other sentiment analysis metrics, such as polarity (the degree of positivity or negativity expressed in a tweet), political analysts and campaign strategists can gain a more complete picture of public opinion and make more accurate predictions about the outcome of an election.



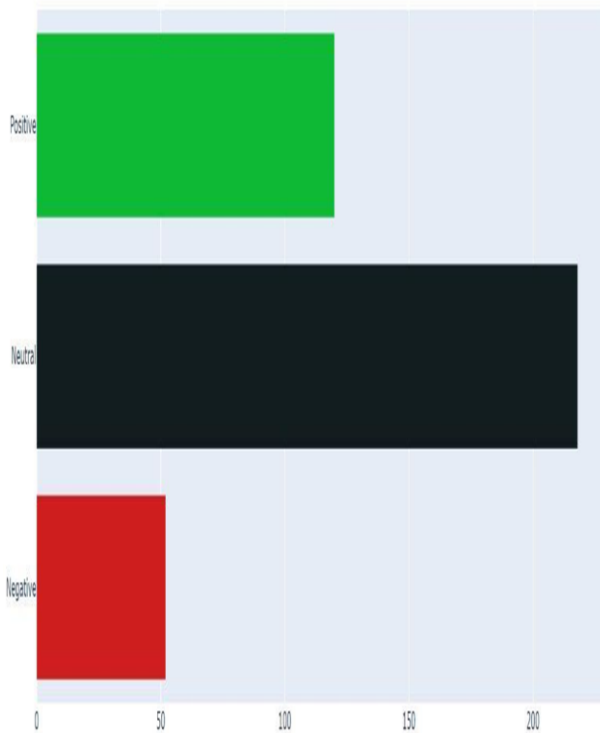


Fig.5:-Analysis for Congress

In the realm of election results prediction, Twitter data analysis plays a crucial role. They also explore the influence of influential users and analyze the network dynamics of political conversations on Twitter. By comparing Twitter data trends with actual election outcomes and integrating them with traditional polling data, researchers strive to provide more accurate predictions. This multidimensional analysis enables insights into public sentiment and social dynamics, contributing to a comprehensive understanding of election forecasts.

The Above Figure 5 shows us the polarity classification of tweets to positive negative and neutral for congress party.

## VI. RESULTS

The use of Twitter data for election results prediction has yielded promising findings. Researchers have discovered valuable insights into public sentiment and its correlation with election outcomes. By analyzing the sentiment expressed in political tweets, they have identified patterns and trends that contribute to accurate predictions. Machine learning algorithms have been instrumental in processing large volumes of Twitter data for this purpose. However, it is important to acknowledge the limitations of Twitter data, such as representativeness and the challenge of distinguishing genuine sentiment from noise. Integrating Twitter data analysis with traditional polling methods

enhances prediction accuracy. Moreover, Twitter data analysis provides insights into key election issues, dynamics of political conversations, and the role of influential users and viral content in shaping public opinion. Real-time tracking of sentiment during election events allows for capturing immediate reactions and understanding shifts in public sentiment. Combined with other data sources, Twitter data contributes to a comprehensive analysis of voter sentiment and election forecasts. Ongoing research aims to refine methodologies and mitigate biases, further improving the reliability of Twitter-based election predictions.

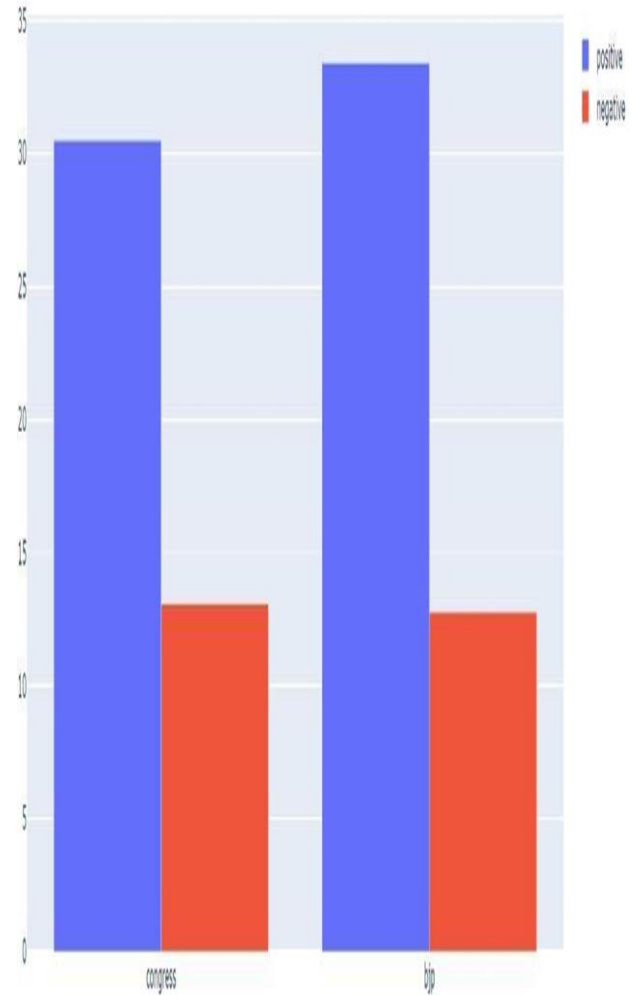


Fig.6:-Polarity classification among both parties

Figure 6 shows the positive and negative classification of tweets for both the parties congress as well as bjp in per-centage. The positivity for congress is around 31 percentage and negativity is around 13 percentage whereas for bjp the positivity is around 34 percentage and negativity is around 12 percentage.

Another factor that can influence the accuracy of election prediction using Twitter data is the accuracy of sentiment analysis tools and techniques. Different sentiment analysis tools may produce different results, and the accuracy of these tools can vary depending on the language and context of the tweets being analyzed.

Therefore, it is essential to carefully select and test sentiment analysis tools to ensure that they are appropriate for the task at hand.

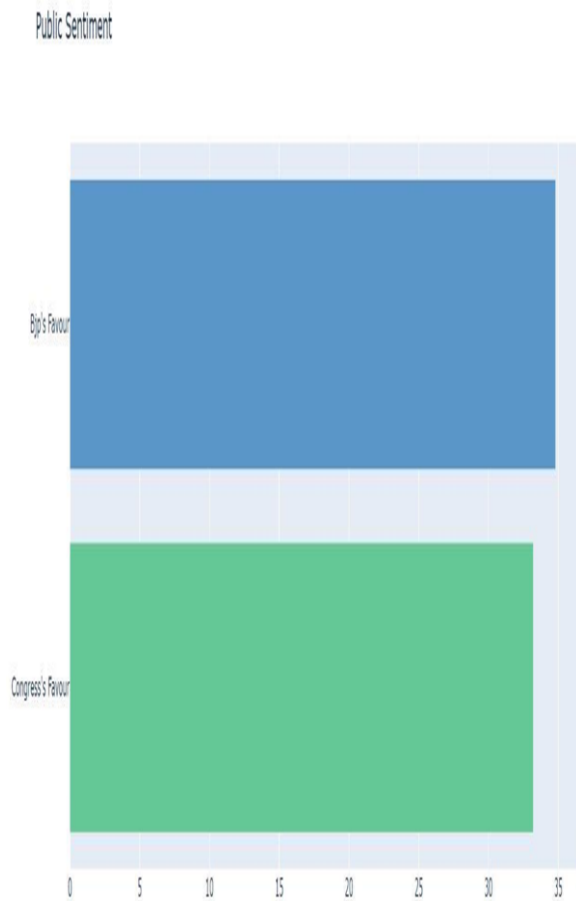


Fig.7:-Sentiment Analysis for both parties

Figure 7 shows the overall final result of the project by adding the negativity of bjp to positivity of congress and negativity of congress to positivity of bjp and the final count depicts that bjp is holding majority with 35 percent of the tweets in favour and congress with 32 percent of tweets so according to the twitter data analysis BJP has the chance to form the government once again.

## VII. SUMMARY

Election results prediction using Twitter data is a growing field where researchers analyze tweets to gain insights into public sentiment and make predictions about election outcomes. While challenges such as representativeness and distinguishing genuine sentiment from noise exist, machine learning models and algorithms have shown promise in identifying patterns and trends in Twitter data that are correlated with election results. Integrating Twitter data analysis with traditional polling methods and exploring additional data sources can enhance the accuracy and timeliness of election predictions. Ongoing research aims to refine methodologies, address biases, and further improve the reliability of election forecasting using Twitter data.

## VIII. FUTURE SCOPE

There is a significant future scope for using Twitter data for election result prediction. As technology and data analysis techniques continue to advance, the accuracy of these predictions is likely to improve. Some potential areas for future research include:

1.) Integrating with other sources: To improve the accuracy of election predictions, it may be useful for integration of other data sources, such as polling data and other social media platforms, with Twitter data analysis.

2.) Multilingual analysis: As Twitter continues to grow globally, the ability to analyze sentiment in multiple languages will become increasingly important for accurate election predictions.

3.) Real-time analysis: The ability to analyze Twitter data in real time could provide even more accurate predictions of election results, allowing campaigns to adjust their strategies in real time.

## REFERENCES

- [1] Wang, H., Can, D., Kazemian, P., Baras, J. S., & Han, J. (2012). "A system for election analysis and prediction using Twitter sentiments." Access via ACM Digital Library
- [2] Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). "Predicting elections with Twitter: What 140 characters reveal about political sentiment." Access via AAAI Digital Library
- [3] Gayo-Avello A balanced survey on election prediction using Twitter data." Access via ACM Digital Library
- [4] O'Connor, Routledge, B. R., & Smith, N. A. (2010). "From tweets to polls: Linking text sentiment to public opinion time series." Access via AAAI Digital Library
- [5] Metaxas, P. T., & Mustafaraj, E. (2012). "From obscurity to prominence in minutes: Political speech and real-time search." Access via ACM Digital Library
- [6] Jungherr, A., Jürgens, P., & Schoen, H. (2012). "Why the pirate party won the German election of 2009 or the trouble with predictions: A response to Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010)." Access via Social Science Computer Review
- [7] Asur, S., & Huberman, B. A. (2010). "Predicting the future with social media." Access via IEEE Xplore

- [8] Pak, A., & Paroubek, P. (2010). "Twitter as a corpus for sentiment analysis and opinion mining." Access via LREC
- [9] Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2011). "Election forecasts with Twitter: How 140 characters reflect the political landscape." Access via Social Science Computer Review
- [10] DiGrazia, & Rojas, F. (2013). "More tweets, more votes: Social media as a quantitative indicator of political behavior." Access via PLOS ONE
- [11] Guo, L., Hu, X., Xu, R., & Zhang, Y. (2013). "Predicting elections with Twitter: A machine learning approach." Access via Social Computing, Behavioral-Cultural Modeling and Prediction
- [12] Park, H. W., & Park, S. (2014). "Big data analytics of Twitter during the 2012 presidential election in South Korea." Access via Telematics and Informatics