From Traditional to Advanced Machine Learning: A Comparative Study of Political Tweet Sentiment Analysis

MSc Research Project
MSc Data Analytics

Vishnunath Nharekkat
Student ID: x22234217

School of Computing
National College of Ireland

Supervisor: **Musfira Jilani**

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Vishnunath Nharekkat
Student ID:	X22234217
Programme:	MSc Data Analytics
Year:	2024
Module:	MSc Research Project
Supervisor:	Musfira Jilani
Submission Due Date:	12/12/2024
Project Title:	From Traditional to Advanced Machine Learning: A Comparative Study of Political Tweet Sentiment Analysis
Word Count:	7378
Page Count:	20

I hereby certify that the information contained in this (my submission) is information about research I conducted for this project. All information other than my contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the referencing standard specified in the report template. To use another author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Vishnunath Nharekkat
Date:	12th December 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☑
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☑
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☑

Assignments that are submitted to the Programme Coordinator's office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

From Traditional to Advanced Machine Learning: A Comparative Study of Political Tweet Sentiment Analysis

Vishnunath Nharekkat
x22234217

Abstract

The growing influence of social media networks on political discourse requires advanced sentiment analysis to recognize better public viewpoints revealed in complicated and diverse textual data. Existing approaches typically struggle to stabilize computational effectiveness with the ability to capture contextual nuances in sentiment classification jobs. In this research, we investigate machine learning and deep learning strategies, for analyzing the sentiments using tweets. To evaluate high-dimensional data with complex linguistic patterns, we preprocessed the data and fine-tuned it for better results. The Outcomes suggest that while SVM attained an accuracy of 85.91% because of its performance in structured data LSTM outmatched a little with an accuracy of 86.34%, succeeding at capturing nuanced linguistic features. From these results, we can understand that LSTM is better suited for sentiment analysis.

1 Introduction

1.1 Research Background

The Social network systems specifically X have changed just how popular opinion is shared as well as recognized making them vital devices for evaluating belief throughout politically substantial occasions like political elections. With its real-time nature plus international reach, X supplies a system for customers to articulate points of view on political leaders, plans as well as occasions, producing a vibrant electronic online forum. In India where political involvement is deeply woven right into the cultural fabric this digital landscape holds tremendous relevance. With a huge plus varied population proactively taking part in politically unsupported claims online assessing tweets uses one-of-a-kind understandings right into citizen belief as well as popular opinion. By understanding trends along with emotional tones, sentiment analysis of politically unsupported claims on X aids reveals the citizen's assumptions plus affects decision-making procedures.

In the Indian context, the obstacles to examining X information are as considerable as the possibilities. Tweets are typically composed in casual, abbreviated language as well as in India regularly extend numerous languages. This variety incorporated with the large quantity of information requires durable logical devices with the ability to deal with complicated linguistic patterns as well as extract significant insights. Conventional approaches to assessing popular opinions, like surveys, cannot match the depth as well as immediacy supplied by social media sites for evaluation. This research study intends to resolve these difficulties by using advanced Natural

Language Processing (NLP) as well as deep understanding designs to assess beliefs in tweets connected to two famous political figures, Narendra Modi and Rahul Gandhi. By discovering machine learning and deep learning algorithms, the research aims to develop a reliable structure for recognizing public belief in the context of Indian political elections.

1.2 Research Question

Q. How do traditional Machine Learning and Deep Learning compare in accuracy, efficiency, and scalability by analyzing the sentiment of X users?

1.3 Proposed Solution

For this area of the report, we evaluated numerous research study documents to check out previous works as well as difficulties related to sentiment analysis on X data in political contexts. Pre-processing of textual data emerged as a crucial aspect of this research, where tasks such as data cleaning (e.g., URLs, hashtags), tokenization, and lemmatization considerably improve the high quality of input data for evaluation. As an example, Khurana, M., Gulati, A., & Singh, S., 2018 highlighted the significance of structured pre-processing pipelines in improving the machine learning algorithm's accuracy for sentiment detection from a raw dataset. Similarly, Sharma, P. & Moh, T.-S., 2016 showed the efficiency of these steps in preparing multilingual data for evaluation, which is specifically appropriate in Indian political elections, offering the variety of languages made use of on social media platforms.

The two papers focused on the methodology of using computational approaches to do sentiment analysis and noted the advantages and disadvantages of each method employed in the field. Some of the machine learning algorithms that have been used are Support Vector Machines (SVM), an employed machine learning algorithms known for their reliability in sentiment classification tasks that have demonstrated strong performance with competitive accuracy rates like 81 %, in certain electoral research studies according to (S. Kumar and M. T. Uddin Haider, 2023). On the other hand, deep learning models like LSTM take advantage of word embeddings to better recognize the contextual subtleties in text data. However, these models need considerable computational sources, and large annotated datasets are difficult to procure. These findings give a durable structure for resolving the obstacles in my research along with establishing an effective technique for political-based sentiment analysis.

1.4 Novelty of the Study

This research study presents a unique strategy for sentiment evaluation by addressing the complexities of political verbalizations on social networks especially concentrating on tweets in the Indian context. The research study inventively integrates advanced NLP strategies with deep learning models to assess sentiment in the direction of two significant political candidates Narendra Modi and Rahul Gandhi leveraging a big, multilingual data source. In this study, SVM is used for efficient classification and to understand the order and context of words in text we use LSTM for that. This research study provides a durable structure for sentiment evaluation in politically related conversations offering useful understandings of real-time public opinion throughout political elections.

1.5 Document Structure

In this section, we discuss the structure of the document, In Section 2 the Literature Review, we discuss previous papers that are related to sentiment analysis and models that are implemented to analyze public opinions. It gives an idea that is helpful for the research to implement the models. Section 3: methodology, details the study structure describing the dataset, pre-processing methods such as cleansing, tokenization, and, together with the advancement of view category versions: SVM and LSTM model. Section 4: Design Specification explains the architecture and key components of the SVM and LSTM models. Section 5, Implementation, in this section we deeply discuss the model implementation and its fine-tuning techniques. Section 6 Results and Evaluation provides the efficiency metrics of the designs contrasting their precision coupled with efficiency in evaluating political views and delves into the analysis of the results. Discusses how they impact sentiment analysis while also suggesting areas, for research to tackle the identified challenges. Finally, section 7 discusses the conclusion and future works, which summarize the findings and suggest directions for further research.

2 Related Work

Sentiment analysis is a section of the Natural Language Processing (NLP) which is incredibly significant in the determination of feelings and attitudes from the given text. The use of sentiment evaluation in politics has now gained much attention, majorly on social media platforms including X where political opinions are expressed fully. Many research studies have discovered various machine learning and deep learning techniques to enhance the accuracy as well as scalability of sentiment classification.

2.1 An Overview of Sentimental Analysis

Sentiment analysis is a method, which is often used in the processes of the determination of people's opinions concerning various aspects of life such as products and services, tourism, educational systems, movies, and political questions. Text mining, which is a process of finding useful data from a large amount of natural language data sets, is a major step in carrying out sentiment analysis (Ramanathan and Meyyappan 2014). Carpenter, T. and Way, T., 2010 define sentiment analysis as an evaluation or an opinion about something; it is emotional rather than rational, making the other name for it – 'opinion mining.' These microblogs can also be regarded as an important means through which users provide comments on events and issues (Pak and Paroubek, 2010).

2.2 X

X is a prevalent micro-blogging site in which users can express their opinions and disseminate information by tweeting messages not exceeding 140 words (Jose, Bhatia, and Krishna, 2010; Lai, 2012; Lohmann et al., 2012). It is an ideal instrument for getting official reactions to different issues (Osimo and Mureddu, 2010; Pak and Paroubek, 2010). Tweets collected from X are widely employed in the studies of opinion mining and natural language processing (Rambocas and Gama, 2013). For instance, Wang et al. used hashtags such as "#happy" and "#sad" in the construction of models that can automatically determine the kind of emotion portrayed in tweets, theEmoji emotions are happiness, sadness, angels, a heart, and fools (Wang et al., 2012). Similarly, Davidov et al. used sentiment analysis tools to analyze opinions which people post through the Tweets (Davidov, Tsur and Rappoport, 2010).

2.3 Machine Learning-Based Approach

Arthur Samuel introduced what is today called machine learning in 1959 as a stochastic process that can design a program that can retrain itself from the data it has collected. That is a type of learning that takes place autonomously and if supplemented with other experiences during attempts at first instances (Zakaryan, 2021). The type of classification that used the target variables in machine learning is supervised and is also appropriate in sentiment analysis. Machine learning has enhanced Natural Language Processing, and more so for sentiment analysis (Sindhu, Kumar, and Noliya, 2023). For example, Rao et al. differentiated political party outcomes using Machine Learning algorithms including Naïve Bayes, Support Vector Machine (SVM), or Random Forest. Among these, SVM has the highest accuracy 78.83%, then Naïve Bayes and Random Forest with an accuracy of 77.38% & 76.53% respectively. For highly dependent features, Domingos and Pazzani (1997) also noted that Naïve Bayes did better in other models.

2.4 Deep Learning Based Approach

Deep learning is advanced to Machine learning pioneered by Alexey Ivankhnenko in the 1960s; it applies artificial neural networks with layers to analyze large data. Twitter sentiment analysis on Urdu text can be seen in Naqvi et al. where authors compared deep learning models such as LSTM & CNN, and the authors found that CNN achieved the best results with around 77.9% accuracy for longer text sentiment detection (Naqvi et al., 2021). Deole, Jagadish, and Shabrez (2023) used Pakistani political parties hashtag #pakistan on Twitter and deployed a Recurrent Neural Network (RNN) with bidirectional LSTM to analyze sentiment expressed in Urdu, Punjabi in IT transliteration and in English. On the test dataset, their proposed model obtained a high level of accuracy of 95.64%.

2.5 Gaps and Limitations

Even though the machine learning and deep learning approaches to sentiment analysis have been moving forward for years, the authors pointed out the great gaps and limitations in the method. Most conventional ML architectures often do not capture the context meaning of the text, they also exhibit problems with respect to sequential dependency, whereas newly proposed DL methodologies, such as transformers are computationally intensive apart from the requirement of quality training data set, which can be a disadvantage for learning on domain-specific data sets. In this work, it is important to note that while the SVM model does well in handling dimensionality and provides good classification, it has no mechanism to learn the temporal characteristics inherent in text data. Further, although LSTM performs representations in a sequential manner with context-based information and flexibility in capturing sequential dependences, it experiences challenges such as slow computation time, and, overfitting when dealing with small data sets. By incorporating these designs, this research intends to reduce the drawbacks of various other techniques. Incorporating SVM's performance with LSTM's contextual understanding offers a well-balanced framework to conquer the limitations of standalone designs, making sure a lot more detailed understandings for political sentiment analysis.

3 Methodology

This research study adheres to an organized technique comprising the actions listed below to attain the goal of identifying X sentiment towards political candidates Narendra Modi and Rahul Gandhi. The methodology follows the CRISP-DM framework, which follows the steps from data collection and preprocessing to model development, evaluation, and model development.

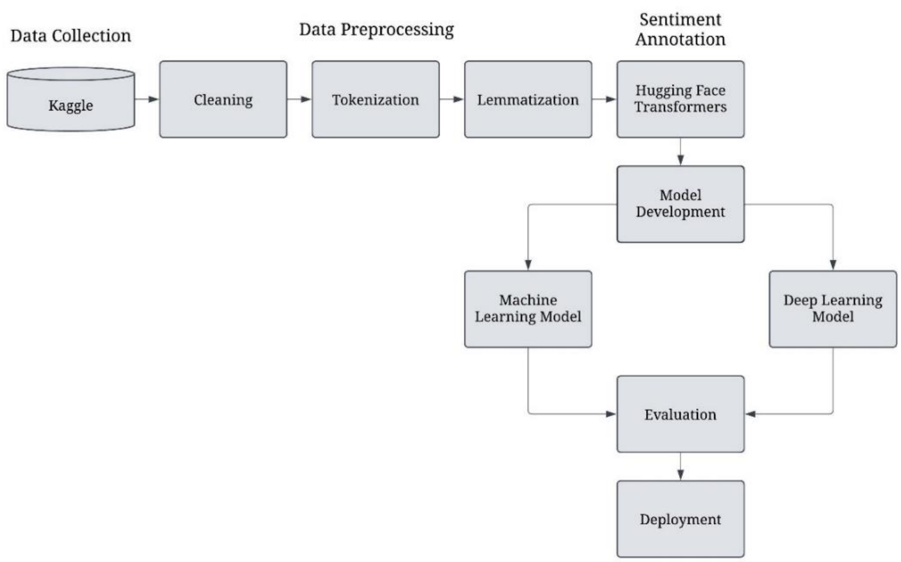

Figure 1: Methodology Diagram

3.1 Data Collection

The dataset for this research study was sourced from Kaggle, which gives publicly available data on tweets relevant to the Indian political elections. This dataset has user-generated tweets, which are examined to identify public opinions throughout the political elections. The dataset is accumulated from a reliable and ethical resource sticking to the ethical standards for the research study.

3.2 Data Understanding and Preprocessing

After loading the dataset in the Jupyter Notebook, the data undertakes the first step to determine unwanted columns plus missing values in the columns. Columns such as " Unnamed: 0,", " Date", together with "User" are eliminated along with the dataset is cleaned up making use of numerous pre-processing strategies. These consist of eliminating URLs, user mentions (e.g., @ username), hashtags, punctuation, numbers, as well as emojis, were removed. The text is additionally transformed to lowercase and stopwords are removed by utilizing the NLTK stopword library.

Finally, lemmatization is used to minimize words to their base form for better understanding. Then we ensure any duplicates were found and removed for better quality data.

3.3 Data Preparation

By understanding the data and preprocessing, we prepared the data for tokenizing the text and analyzing the text using a pre-trained transformer model. Then, we classify each tweet into positive and negative using the Hugging Face pipeline for sentiment analysis. Additionally, we classified the tweets into political party categories (BJL or INC) based on keywords that are related to political parties or candidates. After that, we export the preprocessed and labeled data into a new CSV file for further analysis and model development.

3.4 Modelling

The modeling procedure includes 2 stages; each stage consists of implementing various algorithms to classify and predict the sentiment of tweets. In Stage 1, a Linear Support Vector Machine (SVM) with L1 regularization is used to train the model on the TF-IDF features. Feature selection is carried out making use of SelectFromModel based upon the relevance of features in the trained SVM algorithm. In stage 2, an LSTM (Long Short-Term Memory) model is implemented to analyze the tweets and classify them into sentiment categories. Before using input for LSTM, the text data is tokenized and padded to a fixed length then it is used as input. The model contains 2 LSTM layers, dropout layers for regularization, as well as a dense output layer with a sigmoid activation feature. When training the models, we used early stopping and model checkpoints to prevent overfitting to ensure better model performance.

3.5 Model Evaluations & Predictions

In the model evaluation part, the performance of the models is evaluated using accuracy, classification report including precision, recall, and F1-score, and Confusion matrix display shows the number of correct predictions and model errors. These metrics help us to provide insight into how well the model performs in classifying tweets for sentiment categories and political party labels. The models are trained using training data and tested on the validation set, the best model is selected based on these evaluation criteria. The best model is used for further predictions of unseen new data.

3.6 Deployment & Results

In the deployment stage, the best models are used to understand public opinion toward elections which is helpful for the parties to predict their possibilities to win in the election. The valuable insight of sentiment analysis from the tweets is used for future predictions and they can use it as a tool for similar datasets as well. To evaluate the performance we focus on accuracy, precision, F1-score, recall, and computation performance of the model are used to select the best model.

4 Design Specifications

The design of this research study concentrates on establishing a durable system to evaluate public sentiment along with classifying political associations based on social media data. The system contains numerous essential parts beginning with data pre-processing to ensure the quality along functionality of the textual data. This consists of cleaning, tokenization, and lemmatization, complied with by feature extraction making use of TF-IDF for numerical representation. The design is developed to compare the models of machine learning and deep learning: a Linear Support Vector Machine (SVM) for the baseline sentiment category, and an LSTM (Long Short-Term Memory) network for catching consecutive patterns in text. To ensure the reliability of the model we used hyperparameter optimization to fine-tune each model and its metrics such as accuracy, precision, and recall. To develop the model, we used Python language and its libraries like Pandas, NumPy, Scikit-learn, TensorFlow, and Hugging Face Transformers, to manage a huge dataset and enhance its computing efficiency. The outcome is designed to give actionable insights, with clear visualizations coupled with predictions aligning with the research study goals. This modular as well as scalable design ensures flexibility for future adaptations as well as further evaluations.

4.1 Support Vector Machine (SVM) Architecture

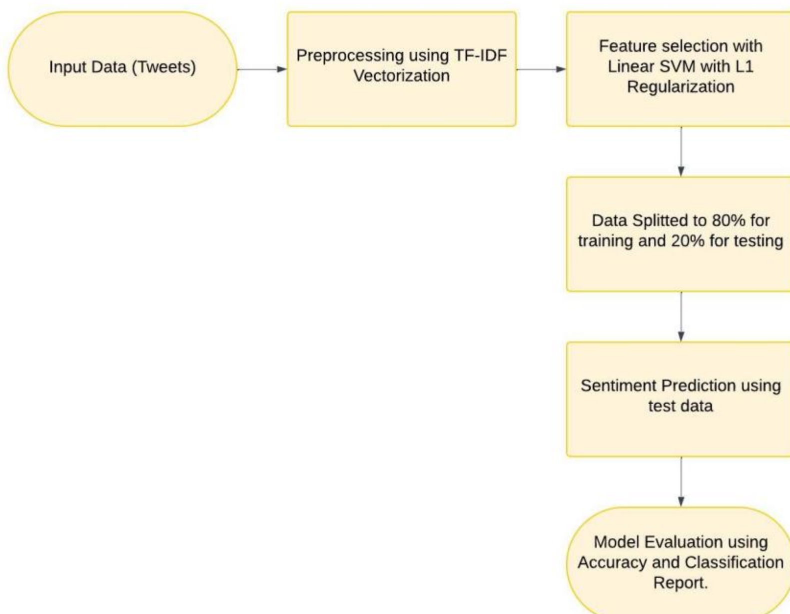

Figure 2: Architecture diagram of Support Vector Machine

The Support Vector Machine (SVM) architecture diagram begins with raw textual data from X, and then the data is preprocessed by removing URLs, mentioned usernames, hashtags, and punctuation. After the preprocessing step, the data is transformed into numerical representations using the TF-IDF vectorization technique, which helps the model capture the importance of words that are relevant to the sentiment based on their frequency. These numerical features are passed as input for the SVM model, which uses a hyperplane to separate the data points into distinct classes. Here, we used feature selection for SVM with L1 regularization, which helps the model to identify the most relevant features for classification. After this process, we split 80% of the data for training and 20% of the data for testing the model, which is evaluated using metrics like accuracy, classification report, and confusion matrix display.

4.2 Long Short-Term Memory (LSTM) Architecture

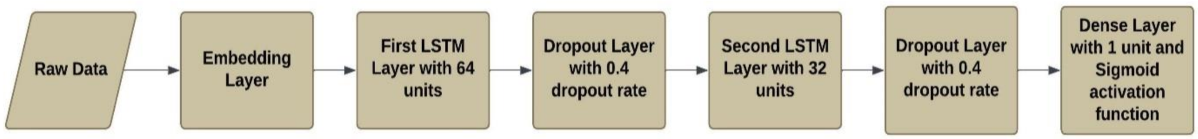

Figure 3: Architecture diagram of Long Short-Term Memory

The Long Short-Term Memory (LSTM) architecture diagram begins with raw textual data from X, and then the data is preprocessed by removing URLs, mentioned usernames, hashtags, and punctuation. After the preprocessing step, the data is tokenized and transformed into a sequence of numbers using a tokenizer. Then the text is padded then passed into an embedding layer. The embedding layer is the layer that changes the words into vectors, which helps the model to capture the meaning of the word. Then, the vectors are passed to multiple LSTM layers which helps the model to understand the order and relationships of the words in the text. To prevent overfitting, we used a dropout layer between the LSTM layers. The output from the LSTM layer is passed to a dense layer with a sigmoid activation function, which helps the model classify the sentiment of the text into positive or negative. Then the model is trained using a binary entropy loss function and evaluated using metrics like accuracy, classification report, and confusion matrix display.

5 Implementation

5.1 Data Understanding

The data understanding stage concentrates on events, discovering, and checking out the data, which functions as the structure of this research study. The data is sourced from Kaggle which includes X blog posts pertinent to both picked political candidates. The dataset contains 1,42,564 rows. Includes various columns that provide specific types of information.

- **Tweet ID (string):** Each tweet is assigned to a Tweet ID, which serves as a code for reference and ensures that there are no duplicate entries, during assessment.
- **Date and Time (string):** This area tapes the specific day and the time at which each tweet was published. Evaluating tweets based on timestamps can be useful for observing patterns specifically throughout political occasions, discussions, or projects as this could mirror changes in public belief.
- **Username (string):** The X username of the person who published the tweet. While this is not always a sign of view evaluating individual interaction patterns as well as persisting with individual messages can give understanding right into individual demographics plus impact.

- **Tweet Text (string):** This is the main message information where the sentiment is extracted. This area is crucial for NLP projects as it contains expressions, viewpoints, and language that will certainly be evaluated to establish a sentiment.

In this step, we performed an initial expedition of the information, examining its total framework together with determining any kind of patterns or abnormalities. Additionally, we assess information's high quality by looking for prospective concerns such as missing out on worths, replicating documents, or unnecessary web content which might require to be attended to in later actions.

5.2 Data Preprocessing

In this study, data preprocessing played an essential function in preparing the raw X data for sentiment analysis and subsequent examination. The dataset is made up of tweets relevant to the 2019 Indian General Elections which were evaluated to recognize public belief in the direction of essential political members and parties. The preprocessing actions were carried out to dispel the noise along with arranging the data for evaluation. These actions are summarized below:

5.2.1 Dataset Loading and Initial Inspection

The data was read into a structure with the aid of Pandas library done by converting the data into a DataFrame, Then the structure and size of the DataFrame was analyzed. Unnecessary columns such as “Unnamed: 0”, “Date”, and “User” were eliminated to concentrate exclusively on the tweet web content.

5.2.2 Data Cleaning

To ensure consistency as well as high-quality data a personalized cleaning feature was established as well as applied to the Tweet column. This feature carried out the following procedures below:

- **Removing URLs:** Using regular expressions, URLs were removed from the tweets to focus on the content only.
- **Removing Mentions:** User mentions (e.g, @username) were removed to anonymize the data.
- **Hashtag Simplification:** Hashtags were simplified by keeping the keyword but removing the # symbol.
- **Punctuation Removal:** Elimination All the punctuation was eliminated by using “string.punctuation” in Python to ensure that the punctuation is eliminated in the right way.
- **Number Removal:** As a measure to eliminate irrelevant noise, numerical values were eliminated.
- **Emoji Conversion:** To translate the emojis we have used an “emoji” that provides more interpretation of the emojis to give a better understanding of the context.
- **Lowercasing:** The text was transformed into lowercase to ensure uniformity.
- **Stopword Removal:** To remove the stopwords from the text we used the NLTK library, which eliminated the words such as “and”, “the”, etc.

- **Lemmatization:** It is another way of ensuring consistency in text representation. In this, we used "WordNetLemmatizer" to lemmatize the words to their base form.

5.2.3 Deduplication and Resetting Indices

To preserve the dataset's originality, duplicate tweets were eliminated, and the DataFrame index was reset to produce a clear sequential order.

5.2.4 Tokenization and Truncation

Tweets were tokenized making use of a pre-trained BERT tokenizer (bert-base-uncased). The number of tokens in each tweet was computed as well as saved in a new column (Token_Count). For tweets going beyond the 512-token limit, truncation was used making use of the tokenizer's encoding technique together with the truncated versions stored in the "Tweets_Truncated" column.

When implementation of these preprocessing steps, it becomes obvious that the data is made more standardized as well as simpler to assist the improved version in analyzing the sentiment required currently.

5.3 Sentiment Annotation

This sentiment annotation was done to positive and negative tweets which gave an insight of the general public feeling towards the political prospects. These steps employed an off-the-shelf sentiment evaluation model, as well as a keyword-based classification for political association. The procedure is explained listed below:

5.3.1 Sentiment Analysis

In using the Hugging Face's Transformers library, a sentiment analysis pipeline was created. For each tweet a sentiment analysis was performed resulting in the respective tweet being classified as either favorable or adverse, capturing the basic sentiment of the tweet.

- Truncation and Padding are to deal with lengthy tweets as well as to ensure compatibility with the model, truncation was put on lower tweets to 512 tokens or less, as well as padding was made use of for uniformity throughout processing.
- For every tweet, there's a corresponding column text labelled with a sentiment which is as follows.

5.3.2 Political Party Classification

In order to link the political association that has been mentioned in each of the given tweets, a keyword search technique was used. Before the start of the study, a basic list of keyword phrases connected with the BJP as well as those connected with INC events was developed.

- Tweets consisting of search phrases connected to BJP (e.g., modi narendra modi bjp) or INC (e.g., rahul gandhi, congress, inc) were identified as necessary.
- Tweets that did not match any type of keywords were classified as Unknown. The political party affiliation was saved in the Political_party column.

To facilitate comparative analysis of public sentiment in the direction of leaders and parties, the dataset was further formatted to incorporate sentiment annotation in parallel to political party classification.

5.4 Model Development

The section on model development describes the machine learning and deep learning techniques employed in this research study for sentiment analysis of tweets. Two unique methods were executed: SVM and LSTM were identified as algorithms that can be relevant to the present study. In the following section, we outlined the model calibration, training processes, and essential characteristics employed.

5.4.1 Support Vector Machine (SVM)

The Support Vector Machine (SVM), which has been introduced in 1992, is one of the most widely applied techniques in the field of learning algorithms, both to classification and regression. Its appeal comes from its solid academic structure as well as practical performance. SVM runs by recognizing a hyperplane that divides data points right into unique classes taking full advantage of the margin in between these classes to enhance decision boundaries. In this research study sentiment classification of tweets is carried out utilizing SVM. The input data, represented as vectors via a weighting procedure, undertakes training to develop patterns. These patterns are then used throughout testing to designate sentiment labels, effectively classifying tweets right into unique classes (Waspodo at el, 2022).

1. **Feature Extraction**
   - The machine only accepts inputs in binary form; when converting the text data to numerical features, the TF-IDF method was used.
   - To further reduce the size of the vocabulary, for the max_features parameter we set 5000 which restricts the vocabulary size.
   - To do this, the ngram_range= parameter is given a value of (1, 2) This enables it to capture both unigram and bi-gram so as to capture the context in relation to the use of a particular word.

2. **Class Imbalance Handling with SMOTE**

   When we discussed the bar chart, we considered the presence of an imbalance in amount of data, to tackle this problem, we used the Synthetic Minority Over-Sampling Technique (SMOTE). This technique solved the imbalance issue by oversampling the minority class by generating synthetic samples. This technique improves the SVM model's ability to more effectively.

3. **Model Training and Testing**
   - By using "train_test_split" we split the data into 80% for training and 20% for testing.
   - In this study, LinearSVC strategy setting is such as Penalty default as L2 regularization for final classification, we use Random State 42 for reproducibility of trained model.

4. **Evaluation**

   When it comes to the evaluation part, the accuracy score and the classification report were used. Using the classification report, we will be able to get the precision, recall and the F1-score for each of the sentiment class.

5.4.2 Long Short-Term Memory (LSTM)

In the case of choosing the model for tweets processing and analysis, the Long Short-Term Memory (LSTM) network was chosen due to its being a deep learning model. LSTMs are a form of Recurrent Neural Network (RNN) that has been highly made to capture dependency in data streams and hence popular for text-based jobs such as sentiment analysis. This model can preserve significant contextual information in time which serves well when evaluating the relevance of the sentence in tweets, for instance (Aljehane, N.O., 2023).

1. **Preprocessing and Tokenization**

   Before training the LSTM version, the initial tweet message in raw form was transformed to feed into the neural network. Initially, the tweets were transformed right into a series of integers making use of a Tokenizer, where each word was assigned a unique integer value. The criterion num_words=10000 was given to consist of only the 10,000 most often occurring words in the dataset, by ensuring the rare words did not make the model complicated. Following the series were standardized to a set dimension of 100 words utilizing pad_sequences. Short tweets were padded with zeros, while longer tweets were truncated making certain all input information had a consistent form suitable with the LSTM model.

2. **Class Imbalance Handling with SMOTE**

   When we evaluated the bar chart, we analyzed the imbalance of data, to resolve this issue we applied the Synthetic Minority Over-Sampling Technique (SMOTE). This technique solved the imbalance issue by oversampling the minority class by generating synthetic samples. This technique improves the LSTM model's ability to more effectively.

3. **Model Architecture**

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 128)	1,280,000
lstm (LSTM)	(None, 100, 64)	49,408
dropout (Dropout)	(None, 100, 64)	0
lstm_1 (LSTM)	(None, 32)	12,416
dropout_1 (Dropout)	(None, 32)	0
dense (Dense)	(None, 1)	33

Total params: 4,025,573 (15.36 MB)

Trainable params: 1,341,857 (5.12 MB)

Non-trainable params: 0 (0.00 B)

Optimizer params: 2,683,716 (10.24 MB)

Figure 4: Summary of model LSTM

The architecture of the LSTM design was created to efficiently learn and predict the sentiment of tweets. It starts with an embedding layer that changes integer-encoded words right into dense vector representations of dimension 128 permitting the model to capture the meaning as well as connections in between words (parameters: input_dim=10000 for a vocabulary of 10,000 words, output_dim=128 for embedding dimensionality). Two LSTM layers are piled to extract consecutive relationships: the initial LSTM layer, with 64 units plus return_sequences= True, passes a series to the following layer while the second LSTM layer with 32 units, condenses these attributes right into a portable representation. To reduce overfitting, droplet layers with a rate of 0.3 adhere to each LSTM layer by randomly deactivating neurons during training. Finally, a dense output layer with a sigmoid activation feature produces a solitary result value, standing for the possibility of a positive sentiment, with results classified as 0 (negative) or 1 (positive) for this binary classification job.

4. **Training Configuration**

The LSTM model was trained by making use of the binary_crossentropy loss function, which gauges exactly how properly the design anticipates binary sentiment labels. The Adam optimizer with a learning rate of 0.001 was selected to effectively upgrade model weights and accelerate convergence. To boost the training process, two callbacks were applied: EarlyStopping which monitored the validation loss as well as stopped training after four consecutive epochs with no enhancement to stop overfitting as well as ModelCheckpoint, which automatically conserved the best-performing model based upon validation loss, by ensuring optimal performance.

5. **Evaluation**

It’s clear in cross-validation where the dataset was divided into 70% of the training dataset which is used to train the model, 20% of the validation dataset which is used to check its accuracy at the course of training and 30% of the testing dataset which determines the model final efficiency. From observations made above it can be concluded that LSTM model successfully preprocesses the text: The LSTM model proved its effectiveness in the field of sentiment classification since it accurately identifies dependencies between words and the general meaning of the tweets. The accuracy of the test supports the idea of its efficiency and applicability for this purpose concurrently with the proved stability of the algorithm.

5.5 Result Documentation and Finalization

Consequently, a design is evaluated in terms of its effectiveness and using parameters as accuracy, precision, recall and F1 score. In addition to these efficient measures, computational complexity coupled with scalability is considered because they define the behavioral patterns of the developed model for large scale sentiment analysis tasks. The examination stage is essential in identifying which model performs highly balanced between accuracy and computational efficiency.

We carry out a comparative evaluation throughout models to examine their strengths and limitations. Accuracy shows each model’s capability to properly classify the sentiments while precision along with recall provides the model's capability to spot the sentiment classes efficiently. F1-score equilibriums precision along with recall giving a solitary action of category efficiency.

5.6 Deployment

The last stage of this research study entails deploying the best-performing design as an analytical tool for real-time sentiment analysis. This tool is made to classify public views on X relating to the political prospects providing stakeholders with a useful source for keeping an eye on popular opinion highly. The implementation enables practical applications by changing the trained model right into a practical sentiment analytical tool that can continually upgrade as brand-new data is accumulated.

This implementation serves as an obtainable user interface for stakeholders to acquire insights right from political opinions on social media platforms which can aid decision-making and help with strategy planning. This tool allows data-driven understandings that show changes in public opinion in time providing a real-time understanding of public sentiment trends towards the candidates.

6 Results and Evaluations

The result and evaluations section discusses the findings of the performance analysis of models that are used for sentiment analysis. This section includes a detailed evaluation of traditional and advanced machine learning models such as Support Vector Machines (SVM) and Long Short-Term Memory (LSTM). Additionally, the outcomes of the model are visualized through confusion matrices, a ROC curve and a training and validation plot providing a detailed understanding of each model is discussed below.

6.1 Data

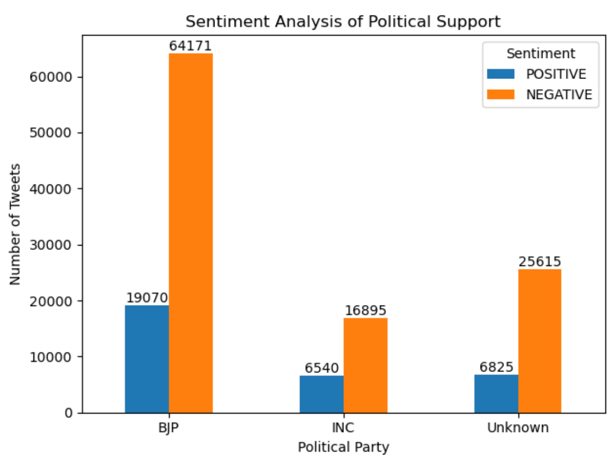

Figure 5: Bart Chart of political support from tweets

From the above bar chart diagram, we can understand political support from tweets by analyzing the sentiment of the tweets from X across different political parties. Evaluating the bar diagram, Bharatiya Janata Party (BJP) has the highest number of tweets compared to the Indian National Congress (INC). The tweets labeled as "Unknown" indicate the tweets that are not relevant to the project, which means it does not support any political parties. Overall, the bar diagram highlights a strong political discussion occurring on social media related to the BJP. From this, we can understand one more thing, the data is imbalanced, to resolve this we used the SMOTE (Synthetic Minority Oversampling Technique) function to balance the data.

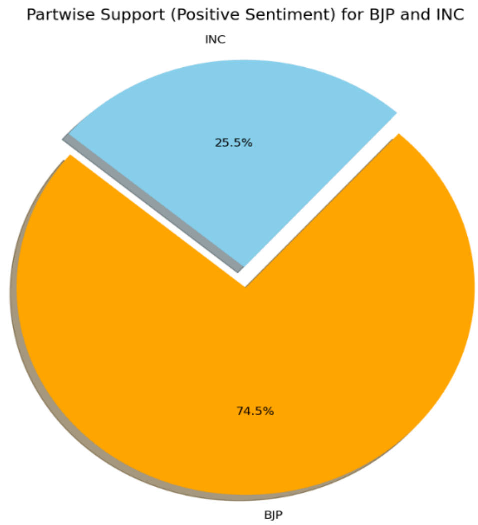

Figure 6: Pie chart of partwise support

From the Pie chart, it shows the part-wise distribution of positive sentiments for the BJP and INC. By evaluating the pie chart, 74.5% of the positive sentiments support the BJP and only 25.5% of positive sentiment gets for INC. So, we can say that almost three times more people support BJL compared to INC. The pie chart highlights that BJL has a stronger foundation for the election.

6.2 Experiment 1: Support Vector Machine

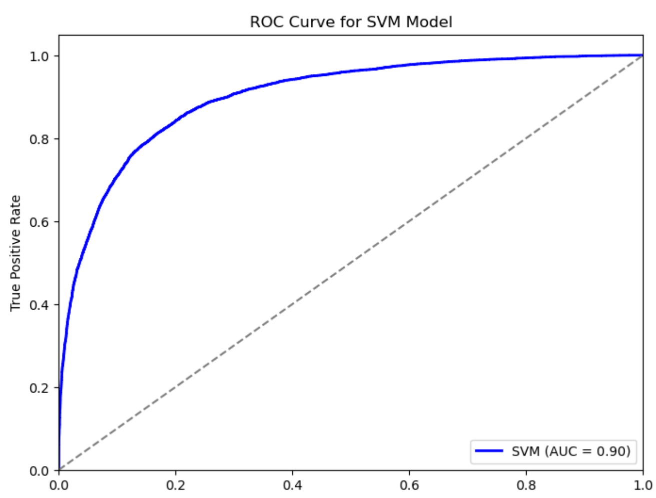

Figure 7: ROC Curve for SVM Model

The above figure shows the Receiver Operating Characteristics (ROC) of the Support Vector Machine model's performance. It compares the True Positive Rate against the False Positive Rate at different thresholds shown. If the True Positive Rate is higher and the False Positive Rate is lower then, we can say that the model is performing well. In the graph, we can see a gray diagonal line, which represents a random guess with an Area Under the Curve (AUC) value of 0.5. From our SVM model, we got an AUC value of 0.90, which represents the model is good at identifying the classes and balancing sensitivity and specificity.

6.3 Experiment 2: Long Short-Term Memory

Figure 8: Accuracy Plot of LSTM

Figure 8 represents the accuracy plot of the LSTM (Long Short-Term Memory) model, which performs on both training and validation data over 5 epochs iterations. The blue line denotes the training accuracy, and the orange line represents the validation accuracy of the model. From the plot, we can see that the training accuracy reached almost 96%, which means that the model learned the data very well. However, the validation accuracy stopped improving after the second epoch and started dropping slightly.

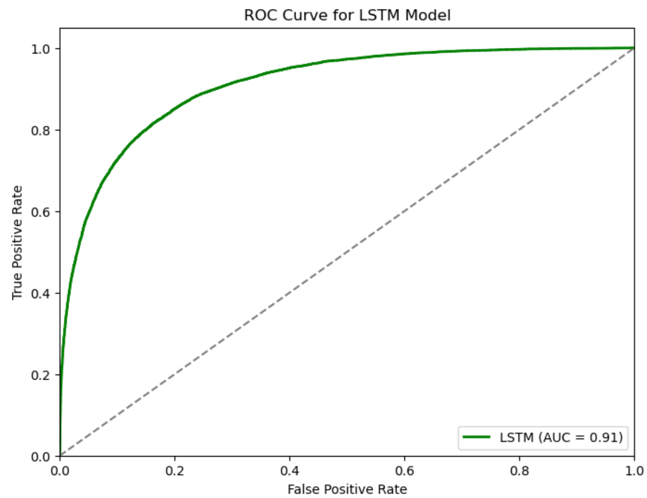

Figure 9: ROC Curve for LSTM Model

The above figure shows the Receiver Operating Characteristics (ROC) of the LSTM model's performance. It compares the True Positive Rate against the False Positive Rate at different thresholds shown. If the True Positive Rate is higher and the False Positive Rate is lower then, we can say that the model is performing well. In the graph, we can see a gray diagonal line, which represents a random guess with an Area Under the Curve (AUC) value of 0.5. From our LSTM model, we got an AUC value of 0.91, which represents the model is good at identifying the classes and balancing sensitivity and specificity.

6.4 Discussion

Metrics	SVM	LSTM
Accuracy	86%	87%
Precision	0.82	0.83
Recall	0.76	0.79
F1-Score	0.78	0.8

Figure 10: Tabular results of SVM and LSTM models

The outcomes of this research highlighted the relative efficiency of analyzing various sentiment models on political tweets. By comparing the traditional machine learning model Support Vector Machine with the deep learning model LSTM for analyzing the sentiments of X users, there are clear distinctions in accuracy, efficiency, and scalability. The LSTM version performs far better than SVM with an accuracy of 87% compared to SVM's 86%. This suggests that LSTM is better at predicting the sentiment in the data. LSTM likewise has greater F1 scores particularly for the

NEGATIVE class, revealing that it is much better at discovering both true positive and true negative instances in the data.

Regarding effectiveness, SVM is faster and utilizes much less computing power than LSTM. SVM is quicker to train as well as it is much less resource-intensive, which can be useful if time, as well as calculation sources, are limited. On the other hand, LSTM needs even more time and computational power to train because of its complexity, but this makes it more effective when managing huge or complex datasets.

When it comes to scalability LSTM is a much better fit for dealing with huge datasets and complicated data because it can understand the patterns of datasets with lots of features or series like text data. By evaluating the SVM model, it is not performing well with huge and complicated datasets.

In conclusion, if the dataset is short and not complex, the SVM model can perform faster and be easy to implement, but in our research, we are working with a huge dataset with complex contents in the text, so by comparing the requirements and performance of the models, LSTM is a good choice for handling complex datasets and for better accuracy.

7 Conclusions and Future Works

As part of this research, we are exploring the Support vector machine (SVM) and Long Short-Term Memory (LSTM) networks which are the traditional and advanced machine learning techniques for sentiment analysis on X data related to the 2019 Indian election. To ensure effective model performance we must concentrate more on the preprocessing of data, the preprocessing steps involve data cleaning and feature selection to improve the model's accuracy and efficiency. From the results, both SVM and LSTM perform very well in understanding the textual public opinion. However, LSTM is slightly performing higher compared to SVM, by proving its ability to understand the temporal relationships in text data and the ability to remember and utilize long-term dependencies in the text.

For future work, this research study can be prolonged by including advanced methods such as transformers or hybrid versions that incorporate numerous deep learning architectures. Moreover, broadening the dataset to include tweets from various periods or various other resources can boost design generalization. Future research studies might likewise check out real-time sentiment evaluation and its application in monitoring continuous political occasions or campaigns. In addition, improving the design with multi-language assistance and including much more advanced NLP techniques might give a much deeper understanding of diverse linguistic and cultural perspectives.

REFERENCES

Khurana, M., Gulati, A., & Singh, S., 2018. *Sentiment Analysis Framework of Twitter Data Using Classification*. In: *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, Solan, India, pp. 459-464.

Sharma, P. & Moh, T.-S., 2016. Prediction of Indian election using sentiment analysis on Hindi Twitter. *IEEE International Conference on Big Data (Big Data)*, Washington, DC, USA, pp. 1966-1971.

S. Kumar and M. T. Uddin Haider, 2023. Sentiment Analysis of Political Party Twitter Data using Ensemble Learning Classifier. *14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India, pp. 1-5.

Lu, B. and Tsou, B.K., 2010. Combining a large sentiment lexicon and machine learning for subjectivity classification. *2010 International Conference on Machine Learning and Cybernetics*, Qingdao, China, pp. 3311-3316.

Aljehane, N.O., 2023. A new approach to sentiment analysis on Twitter data with LSTM. *3rd International Conference on Computing and Information Technology (ICCIT)*, Tabuk, Saudi Arabia, pp. 657–663.

Waspodo, B., Aini, Q., Singgih, F.R., Kusumaningtyas, R.H. & Fetrina, E., 2022. Support Vector Machine and Lexicon-based Sentiment Analysis on Kartu Prakerja (Indonesia Pre-Employment Cards Government Initiatives). *2022 10th International Conference on Cyber and IT Service Management (CITSM)*, Yogyakarta, Indonesia, pp. 1–6.

Ramanathan, V. and Meyyappan, T., 2014. An Exhaustive Exploration on Twitter Sentiment Analysis. *Journal of Computer Science and Applications International Research Publication house*, 6(1), pp.1-10.

Carpenter, T. and Way, T., 2010. Tracking Sentiment Analysis through Twitter. *ACM Computer Survey*. Villanova: Villanova University.

Pak, A. and Paroubek, P., 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. *Special Issue of International Journal of Computer Application*. France: Université de Paris-Sud.

Jose, A.K., Bhatia, N. and Krishna, S., 2010. Twitter Sentiment Analysis. *National Institute of Technology Calicut*.

Lai, P., 2012. Extracting Strong Sentiment Trend from Twitter. *Stanford University*.

Lohmann, S., Burch, M., Schmauder, H. and Weiskopf, D., 2012. Visual Analysis of Microblog Content Using Time-Varying Co-occurrence Highlighting in Tag Clouds. *Annual Conference of VISVISUS*.

Osimo, D. and Mureddu, F., 2010. Research Challenge on Opinion Mining and Sentiment Analysis. *Proceedings of the 12th Conference of FRUCT Association*.

Rambocas, M. and Gama, J., 2013. Marketing Research: The Role of Sentiment Analysis. *The 5th SNA-KDD Workshop*.

Wang, W., Chen, L., Thirunarayan, K. and Sheth, A.P., 2012. Harnessing Twitter's 'big data' for automatic emotion identification. In: *Privacy, Security, Risk and Trust (PASSAT) 2012 International Conference on and 2012 International Conference on Social Computing (SocialCom)*. pp. 587-592.

Davidov, D., Tsur, O. and Rappoport, A., 2010. Enhanced sentiment learning using Twitter hashtags and smileys. *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pp. 241-249.

Sharma, A. and Dey, S., 2012. Performance Investigation of Feature Selection Methods and Sentiment Lexicons for Sentiment Analysis. *Proceedings of the Association for the Advancement of Artificial Intelligence*.

Zakaryan, A., 2021. Application of artificial intelligence (neural networks) in education. *Main Issues of Pedagogy and Psychology*, 19(1), pp. 78-87.

Sindhu, S., Kumar, S. and Noliya, A., 2023. A Review on Sentiment Analysis using Machine Learning. *Proceedings of the 2023 International Conference on Innovative Data Communication Technologies and Application (ICIDCA)*, Uttarakhand, India, pp. 138-142.

Rao, D.R., Usha, S., Krishna, S.S., Ramya, M.S., Charan, G.S. and Jeevan, U., 2020. Result Prediction for Political Parties Using Twitter Sentiment Analysis. *International Journal of Computer Engineering and Technology*, 11(4), pp.1-6.

Domingos, P. and Pazzani, M., 1997. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3), pp.103-130.

Naqvi, U., Majid, A. and Abbas, S.A., 2021. UTSA: Urdu Text Sentiment Analysis Using Deep Learning Methods. *IEEE Access*, 9, pp.114085-114094.

Deole, Y., Jagadish, T., Shabrez, M., P, M. and A, G., 2023. Multi-Lingual Sentiment Analysis of Urdu and English Tweets Using RNN with Bidirectional LSTM. *2023 6th International Conference on Recent Trends in Advanced Computing (ICRTAC)*, Chennai, India, pp. 813-817.