Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

A Project Report

on

"Sentiment Analysis: Nepal's Political Scenario"

[Code No: COMP 313]

(For partial fulfillment of 3rd Year 2nd Semester in Computer Science)

Submitted by

Sujan Ghimire (15)
Bigyan Kumar Piya (41)
Manasvi Sharma (52)
Sewan Uprety (60)

Submitted to

Suman Shrestha

Department of Computer Science and Engineering

13th January, 2025

Bona fide Certificate

This project work on

“Sentiment Analysis: Nepal’s Political Scenario”

is the bona fide work of

Sujan Ghimire

Bigyan Kumar Piya

Manasvi Sharma

Sewan Uprety

who carried out the project work under my supervision.

Project Supervisor

____

Dr. Bal Krishna Bal

Professor

Department of Computer Science and Engineering

Date: 13th January, 2025

Abstract

This project aims to develop a sentiment analysis system specifically designed to detect and classify emotions expressed in political discourse from sources such as social media, news articles, and public forums. Sentiment analysis plays a crucial role in understanding public opinion, enabling governments, policymakers, and organizations to gauge the political climate and address public concerns effectively. The dataset consists of comments directed at major political figures both before and after the election. The primary goal is to evaluate changes in public perception between these two timeframes. By leveraging the BERT and RoBERTa models, we have analyzed both sentiments and emotions. This solution is capable of processing large-scale datasets and delivering real-time sentiment insights, making it valuable for tracking political trends, analyzing public reactions to policies, and predicting electoral outcomes.

Keywords: Sentiment Analysis, Emotion detection, Natural Language Processing (NLP), Text Classification, Market Research, Political Sentiment Monitoring, Nepal

Table of Contents

List of Figures

List of Tables

Acronyms/Abbreviations

NLP	Natural Language Processing
LSTM	Long Short-term Memory
DBMS	Database Management System
URL	Uniform Resource Locator
GPU	Graphics Processing Unit
TF-IDF	Term Frequency-Inverse Document Frequency
SVM	Support Vector Machine
NLTK	Natural Language Tool Kit
BoW	Bag of Words
VADER	Valence Aware Dictionary and Sentiment Reasoner
CNN	Convolutional Neural Network

Chapter 1 Introduction

1.1 Background

With the proliferation of social media, these platforms have become essential venues for individuals to share their political views, opinions, and ideologies. The resulting surge in user-generated content has underscored the need to analyze the sentiments embedded within extensive volumes of political text. Political sentiment analysis primarily aims to classify public discourse as expressing positive, negative, or neutral sentiments toward political parties, candidates, or policies. By examining social media posts, news comments, and other forms of online political expression, researchers and analysts can gain valuable insights into public reactions to political events and decisions. This, in turn, enables politicians and campaign teams to refine their strategies effectively.

Advancements in machine learning, particularly deep learning, have greatly enhanced the precision of political sentiment analysis. Methods such as word embeddings (e.g., Word2Vec, GloVe) and transformer-based models like BERT and GPT have facilitated more nuanced sentiment detection by capturing the subtleties and contextual nuances of political language. Notably, one study, *Sentiment Analysis of Political Communication*, proposed a novel approach by integrating dictionary-based methods with crowd coding. While dictionary-based methods enable rapid processing of large datasets, they often face challenges in interpreting political context, sarcasm, and subtle language nuances. Similarly, the paper *A Preliminary Investigation into Sentiment Analysis of Informal Political Discourse* highlighted that traditional sentiment analysis techniques perform well with structured and formal texts but struggle with the complexities of informal communication commonly found on social media and online forums.

1.2 Objectives

- Develop a sentiment analysis system to detect and classify public sentiment towards political entities and policies from online sources.
- Improve analysis of informal language, including slang and casual expressions.
- Extend the system's capabilities to analyze political sentiment in multiple languages, enhancing its applicability across different regions.
- Facilitate timely analysis by processing large volumes of data and delivering immediate insights into public sentiment.

1.3 Motivation and Significance

We chose this topic to address the critical need for understanding public sentiment in political contexts, given the influence of online platforms on political discourse. Our project aims to improve sentiment analysis by employing advanced machine learning techniques that better capture nuances and variations in political opinions. This system focuses on real-time analysis and deeper insights into how people's attitudes and inclinations shift in response to political events and situations, offering more precise and actionable information for political strategies and decision-making.

1.4 Expected Outcomes

To a certain degree, we expect our project to correctly analyze the generalized sentiment of the public towards the current political scenario. The system is designed to provide real-time analysis of large volumes of political text, delivering timely and relevant insights into public sentiment. By offering a deeper understanding of how public attitudes and inclinations shift in response to political events, candidates, and policies, the project aims to furnish valuable information that can guide political campaigns, policymaking, and strategic decision-making. This will ultimately help stakeholders respond more effectively to public opinion and improve their engagement with political audiences.

Chapter 2 Related Works/ Existing Works

2.1 Sentiment Analysis of Political Tweets: Towards an Accurate Classifier by Akshat Bakliwal, Jennifer Foster, Jennifer van der Puil, Ron O'Brien, Lamia Tounsi and Mark Hughes

The paper titled *"Sentiment Analysis of Political Tweets: Towards an Accurate Classifier"* explores sentiment analysis of political tweets, focusing on those shared during the 2011 Irish General Election. The study aimed to classify tweets as positive, negative, or neutral toward political parties or leaders, while excluding sarcastic tweets to simplify the classification task. The researchers compared supervised machine learning techniques with unsupervised, lexicon-based methods, achieving their highest classification accuracy of 61.6%. The dataset consisted of 2,624 manually annotated tweets, with sentiment classifications provided by experts. The study employed a combination of Twitter-specific features, such as emoticons and hashtags, alongside traditional bag-of-words models and sentiment lexicons like the Subjectivity Lexicon and SentiWordNet. Lexicon-based approaches alone achieved an accuracy of 58.9%, while integrating machine learning models and handcrafted features improved the accuracy to 61.6%. However, the study faced challenges due to the complexity of microtext (tweets) and the prevalence of sarcastic language, which proved difficult to classify accurately. The authors suggest that future work should focus on improving sarcasm detection and exploring additional features to enhance classification performance. Overall, while machine learning showed promising results, political sentiment analysis in tweets remains a complex task that requires further refinement to overcome issues such as sarcasm and the brevity of the text.

2.2 A Preliminary Investigation into Sentiment Analysis of Informal Political Discourse by Tony Mullen and Robert Malouf

The paper titled "A Preliminary Investigation into Sentiment Analysis of Informal Political Discourse" by Mullen and Malouf explores sentiment analysis within the domain of informal political discussions, particularly in online forums. The study focuses on posts made in political discussion groups, aiming to determine whether traditional text classification techniques can effectively identify the sentiment expressed. The authors employed a dataset consisting of posts from an online American political forum and conducted various statistical tests to understand the interaction between users' sentiments and political viewpoints.

One key finding was that posts made in response to others frequently exhibited opposing political views. This highlights the need for sentiment analysis methods that account for dialogue structure rather than relying purely on word-based classification models. The authors concluded that traditional methods, such as Naive Bayes classifiers, struggled to achieve high accuracy in this domain due to the informal and unstructured nature of the text, where rhetorical devices like sarcasm and pragmatic expressions were prevalent.

The researchers achieved a modest classification accuracy of 60.37% when using text-based machine learning techniques, suggesting that incorporating social interaction features, such as quoting behavior between users, might enhance performance. For instance, the study found that users frequently quoted individuals from opposing political perspectives, suggesting that sentiment classification could be improved by considering the relationship between interacting users. The study also encountered challenges specific to informal political discourse, such as rampant spelling errors, informal grammar, and the use of politically charged jargon. The authors propose future work to integrate better linguistic models, including named entity recognition and spelling correction, to improve the classifier's ability to handle these complexities.

In summary, the paper shows that while traditional sentiment analysis approaches have some utility in the political domain, they are not well-suited for capturing the nuances of informal political discourse. Future research should focus on exploiting dialogue structure and refining text preprocessing techniques to improve classification accuracy.

2.3 Sentiment analysis of political communication: combining a dictionary approach with crowdcoding by Martin Haselmayer Marcelo Jenny

The study titled *"Sentiment analysis of political communication: combining a dictionary approach with crowdcoding"* focuses on analyzing sentiment in political discourse, particularly in Austrian elections. The researchers aimed to create a German-language sentiment dictionary tailored to political communication. They employed a hybrid approach, combining crowdcoding—a method of collecting sentiment scores from online annotators—with a lexicon-based approach, allowing for large-scale analysis of political texts such as party press releases and media reports. The use of crowdcoding helped overcome challenges related to manual coding, such as time and cost, by employing non-expert coders to evaluate sentiment strength on a 5-point scale. The researchers tested their sentiment dictionary against expert-coded data and achieved a high correlation between crowdcoded and expert ratings, suggesting the effectiveness of their approach. However, the study faced limitations, such as the complexity of accurately capturing sentiment in contexts like sarcasm or rhetorical questions. Despite these challenges, the study demonstrated the potential of customized, domain-specific sentiment dictionaries for political communication, providing insights into phenomena such as negative campaigning and media negativity. The authors recommend further research to refine these methods, especially in handling figurative language and expanding the dictionary for broader applications.

Chapter 3 Design and Implementation

3.1 Workflow of program

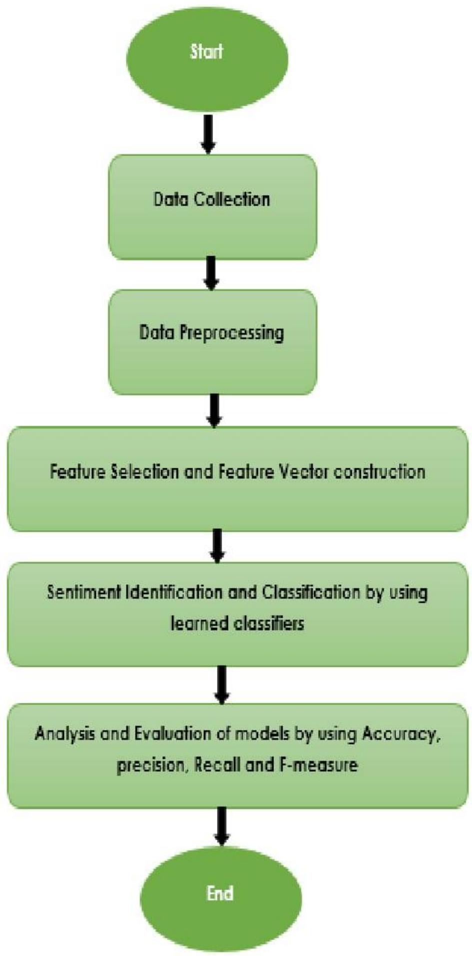


Figure 3.1.1 Workflow of the program

This flowchart outlines the workflow for our group's sentiment analysis project. We begin with data collection from sources like social media, followed by data preprocessing to clean and prepare the data. Next, we focus on tokenization and normalization of these data for further processing. Using machine learning models, we then perform sentiment identification and classification, categorizing text as positive, negative, or neutral. Finally, we analyze and evaluate the model's performance with metrics like accuracy, precision, recall, and F-measure to refine and improve the system.

3.2 Data Collection

Firstly, we decided to collect data on one 4 main political figures: KP Sharma Oli, Sher Bahadur Deuba, Pushpa Kamal Dahal (Prachanda) and Balen Shah. We scraped the comments made about these individuals on platforms such as YouTube and Twitter via their APIs. We have divided our datasets into two categories:

i. Before the election
ii. One year after the election

These two datasets enabled us to perform a comparative analysis of public opinions during these two-time frames.

Figure 3.2.1 Web scraping

3.3 Data Preprocessing

- **Language detection and transliteration:**

We had our datasets in 3 different scripts, English, Romanized Nepali and Devanagari, so we segregated our datasets into these 3 categories. The identification of these 3 scripts was facilitated by langdetect library.

Furthermore, we transliterated Romanized Nepali to Devanagari via deep_translator. We later merged the transliterated Romanized Nepali and preliminary Devanagari datasets into one single dataset.

ma khusi chuu → *मा खुसी छु*

- **Text Cleaning**:
    - Remove irrelevant symbols, special characters, and URLs.
    - Normalize case (convert to lowercase) and handle stopwords. In the case of Devanagari script, we have predefined stopwords later filtered from the dataset.
- **Tokenization**: Use tokenization techniques (Word Tokenizer for Nepali) to split text into individual tokens or words.
- **Lemmatization/Stemming**: Reduce words to their base forms using language-specific tools such as nltk.stem and spacy.

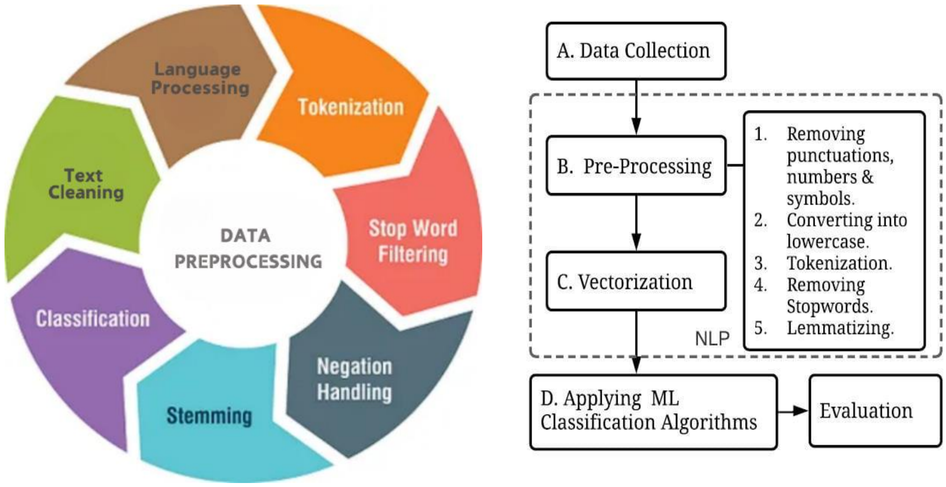
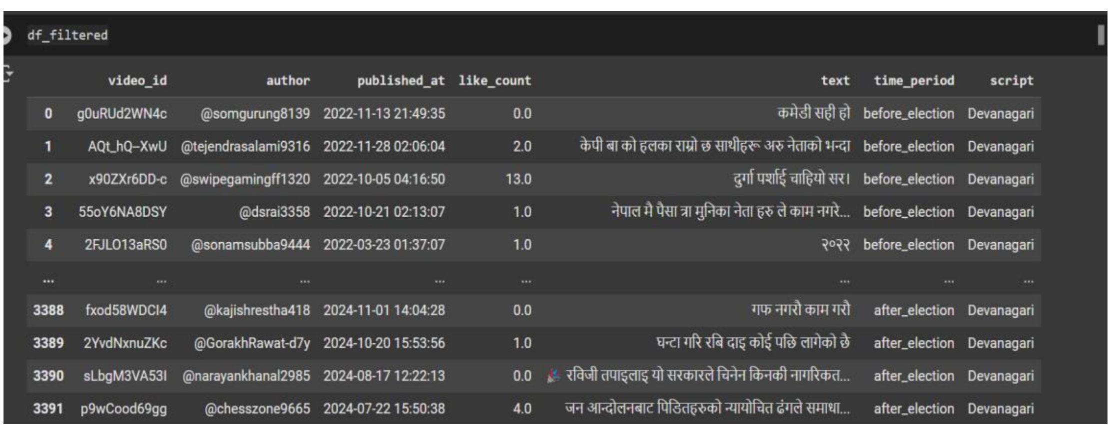


Figure 3.3.1 Pre-processing of data overview (NLP)

3.4 EDA (Exploratory Data Analysis):

EDA allowed us to explore the size, structure, and distribution of documents and sentences, helping us understand the nature of the text data. By analyzing word frequencies, EDA helped us identify the most common themes, words, and phrases in the text, giving us a better understanding of what the data represents.



```
df_filtered
```

	video_id	author	published_at	like_count	text	time_period	script
0	g0uRUd2WN4c	@somgurung8139	2022-11-13 21:49:35	0.0	कमेडी सही हो	before_election	Devanagari
1	AQt_hQ-XwU	@tejendrasalami9316	2022-11-28 02:06:04	2.0	केपी बा को हलका राम्रो छ साथीहरू अरु नेताको भन्दा	before_election	Devanagari
2	x90ZXr6DD-c	@swipegamingff1320	2022-10-05 04:16:50	13.0	दुर्गा पर्शाई चाहियो सर।	before_election	Devanagari
3	55oY6NA8DSY	@dsrai3358	2022-10-21 02:13:07	1.0	नेपाल मै पैसा त्रा मुनिका नेता हरु ले काम नगरे...	before_election	Devanagari
4	2FJLO13aRS0	@sonamsubba9444	2022-03-23 01:37:07	1.0	२०२२	before_election	Devanagari
...	...	...	...	...	...	...	...
3388	fxod58WDCI4	@kajishrestha418	2024-11-01 14:04:28	0.0	गफ नगरौ काम गरौ	after_election	Devanagari
3389	2YvdNxnuZKc	@GorakhRawat-d7y	2024-10-20 15:53:56	1.0	घन्टा गरि रबि दाइ कोई पछि लागेको छै	after_election	Devanagari
3390	sLbgM3VA53I	@narayankhanal2985	2024-08-17 12:22:13	0.0	🎉 रविजी तपाइलाइ यो सरकारले चिनेन किनकी नागरिकत...	after_election	Devanagari
3391	p9wCood69gg	@chesszone9665	2024-07-22 15:50:38	4.0	जन आन्दोलनबाट पिडितहरुको न्यायोचित ढंगले समाधा...	after_election	Devanagari

Figure 3.4.1 Devanagari dataset

- **Word clouds:**

  We used **word clouds** to evaluate the most prominent terms in the text data before and after the election. The word cloud comparison between before and after the election showed a clear shift in focus from campaigning and political strategies to post-election analysis, results, and future implications. This visualization helped us understand the changing nature of public discourse and provided insights into the themes that dominated the conversation before and after the election. Below are the word clouds for the 4 political figures:

  i. **KP Sharma Oli:**

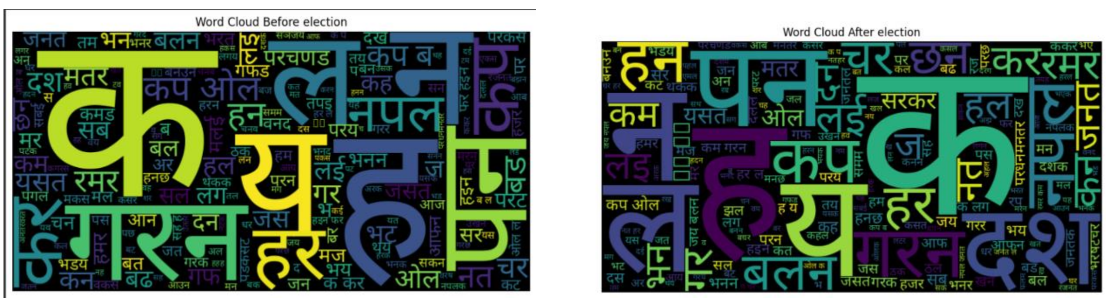


Figure 3.4.2 Word cloud for KP Oli (Devanagari)

Figure 3.4.3 Word cloud for KP Oli (English)

ii. Sher Bahadur Deuba:

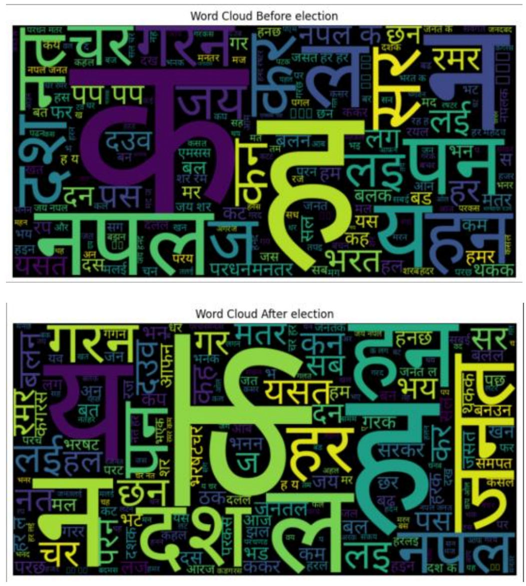


Figure 3.4.4 Word cloud for Sher B Deuba (Devanagari)

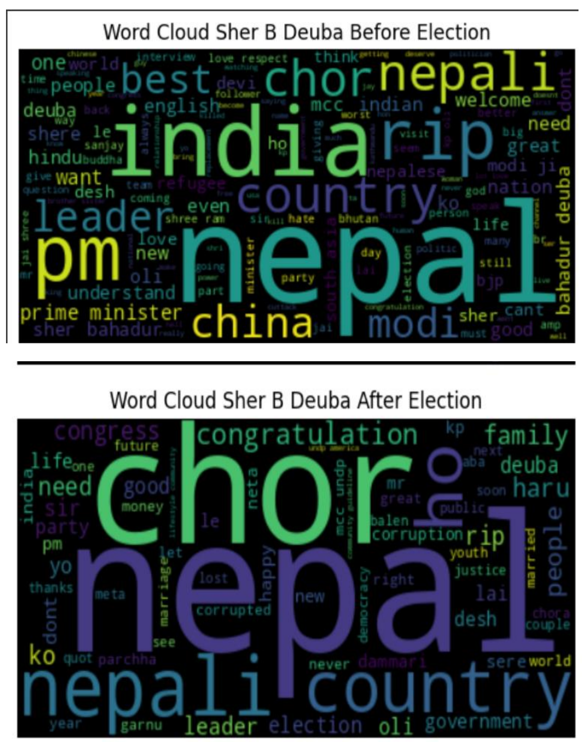


Figure 3.4.5 Word cloud for Sher B Deuba (English)

iii. Balen Shah:

Figure 3.4.6 Word cloud for Balen Shah (Devanagari)

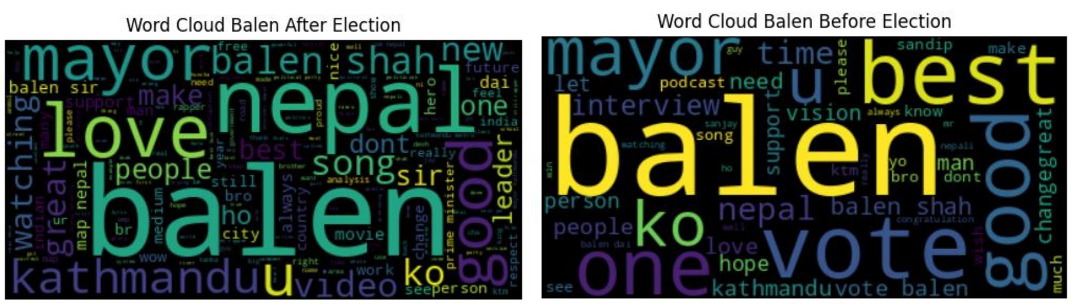


Figure 3.4.7 Word cloud for Balen Shah (English)

iv. Pushpa Kamal Dahal:

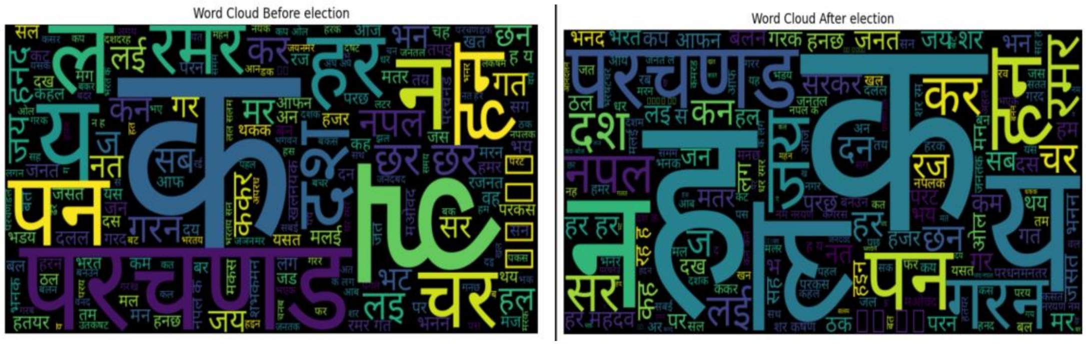


Figure 3.4.8 Word cloud for Pushpa Kamal Dahal (Devanagari)

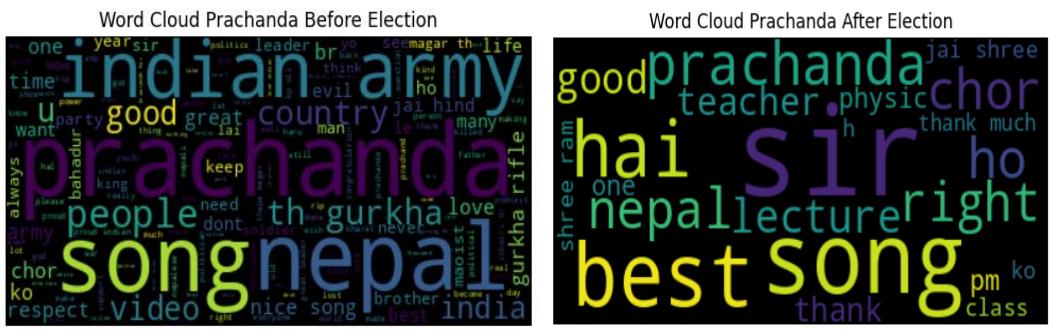


Figure 3.4.9 Word cloud for Pushpa Kamal Dahal (English)

- **Bar graphs for top words:**
  In addition to the **word clouds**, we also generated **bar graphs** to show the top words before and after the election, providing a more precise comparison of the most frequently mentioned terms in the text data. This visual representation helped quantify the evolution of key themes and provided a more detailed, numerical insight into the shift in public and media attention.

  a. KP Sharma Oli:

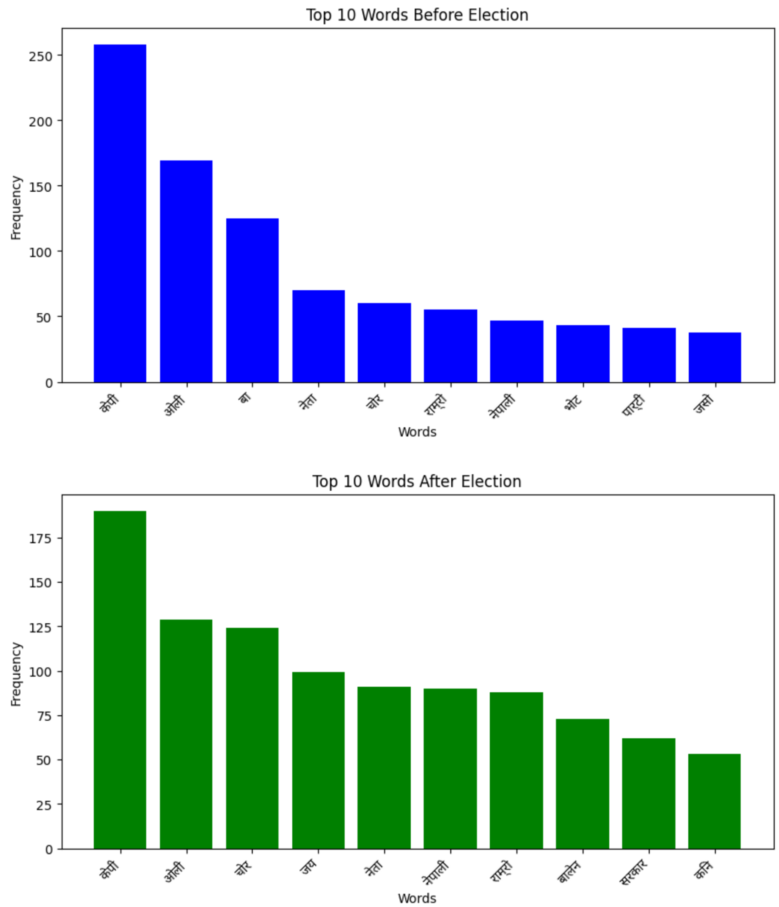
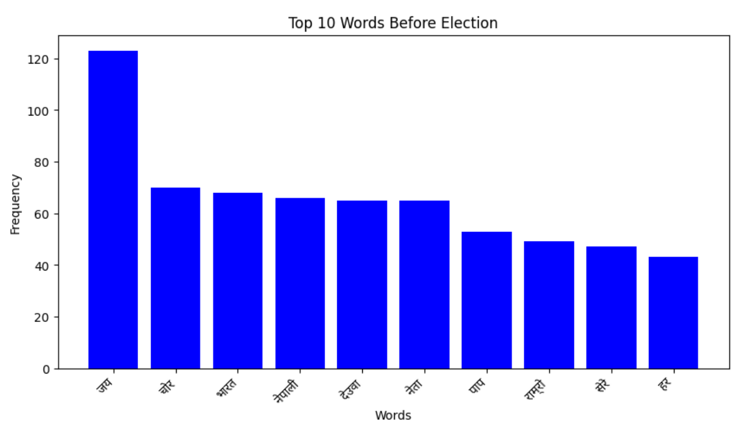


Figure 3.4.10 Bar graph showing top words for KP Oli (Devanagari)

b. Sher Bahadur Deuba:



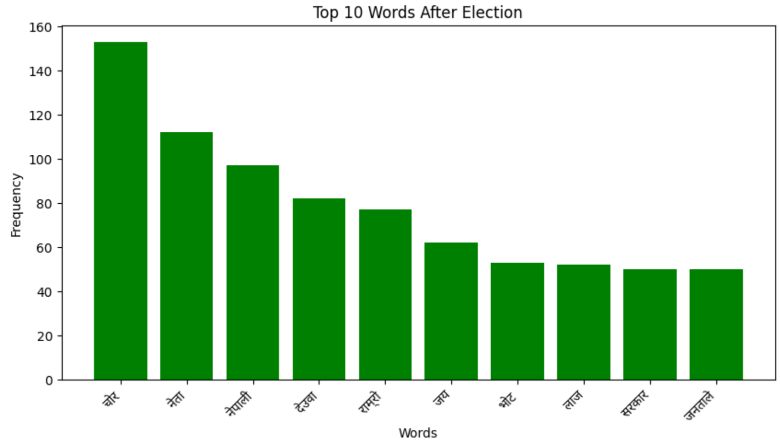


Figure 3.4.11 Bar graph showing top words for Sher B Deuba (Devanagari)

c. Balen Shah:

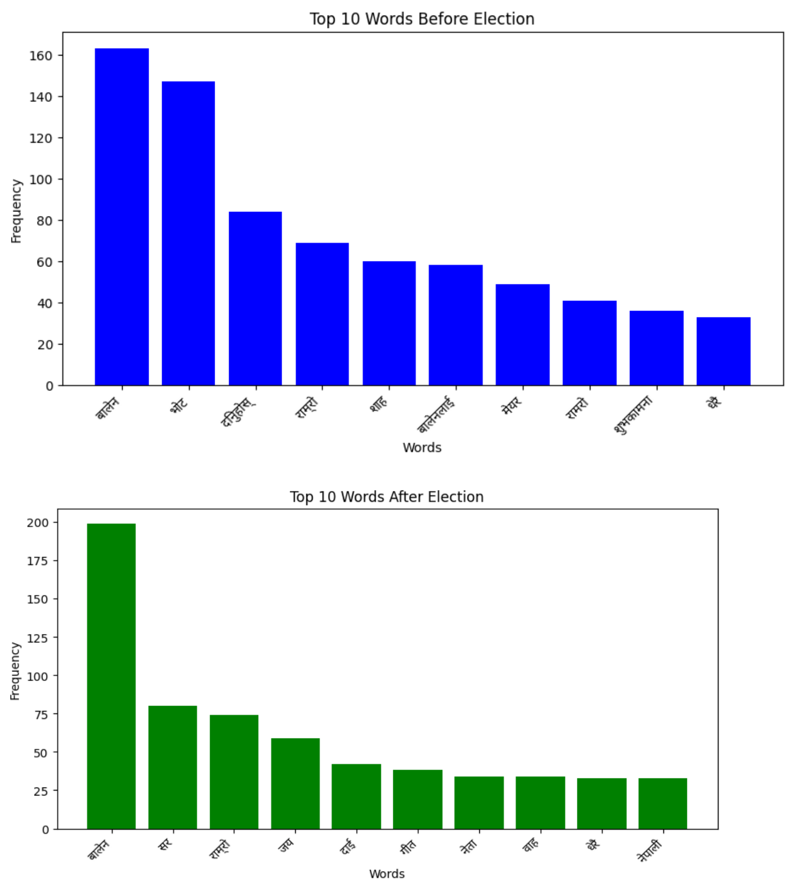


Figure 3.4.12 Bar graph showing top words for Balen Shah (Devanagari)

d. Pushpa Kamal Dahal:

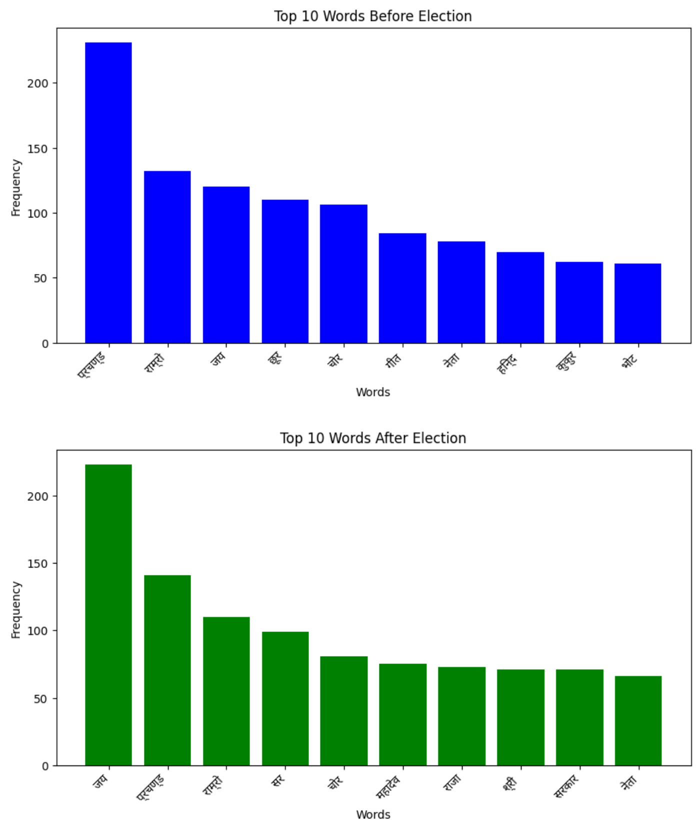


Figure 3.4.13 Bar graph showing top words for Pushpa K Dahal (Devanagari)

3.5 Sentimental analysis using pre trained models:

- For English script:

  Here, we performed sentiment analysis as well as emotion detection by using the comments addressed towards these political figures. For the English script, sentiments are only classified as positive and neutral. In addition to it, emotions are classified as joy, anger, disgust, fear, neutrality, sadness and surprise.

➢ Models:

i. DistilBERT model for sentiment analysis:

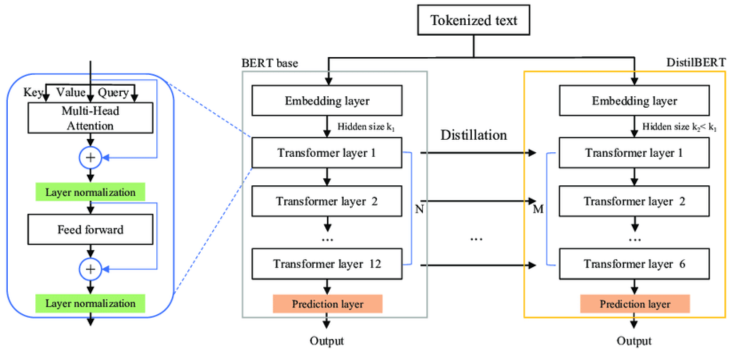


Figure 3.5.1 Architecture of DistilBERT model

DistilBERT is a smaller, faster, and more efficient version of the **BERT (Bidirectional Encoder Representations from Transformers)** model. It is trained using a technique called **knowledge distillation**, where a smaller model is trained to replicate the behavior of a larger, more complex model, like BERT. As a result, DistilBERT retains much of BERT's performance while being more computationally efficient. In our case, the model has been fine-tuned on the **SST-2 dataset**, a sentiment analysis dataset labeled positive and negative movie reviews.

Model architecture:

- **Base Model**: DistilBERT is a transformer-based model that uses the same architecture as BERT but with fewer layers.
- **Number of Layers**: DistilBERT has **6 transformer layers** (compared to BERT's 12 layers).
- **Hidden Size**: The hidden size (embedding size) is **768**.
- **Attention Heads**: DistilBERT uses **12 attention heads** in each transformer layer.
- **Parameters**: It has around **66 million parameters**, which is 40% fewer than the 110 million parameters in BERT.

ii. **DistilRoBERTa** model:

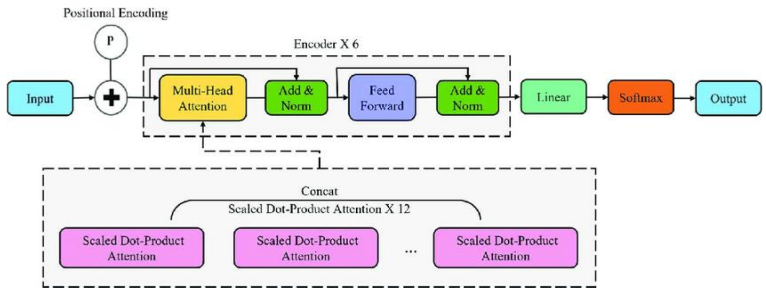


Figure 3.5.2 Architecture of DistilRoBERTa model

The **emotion-english-distilroberta-base** model is a **DistilRoBERTa** model fine-tuned for **emotion detection** in text. It can identify a range of emotions such as **joy**, **anger**, **fear**, **sadness**, **surprise**, and **disgust**. DistilRoBERTa is a lighter, faster version of **RoBERTa**, offering similar performance but with reduced computational requirements. This model is ideal for real-time applications like analyzing social media, customer feedback, and other textual data where detecting emotional tone is important.

Model Architecture:

- **Base Model**: **DistilRoBERTa** is a distilled version of **RoBERTa** (a variant of BERT). It's smaller, faster, and more efficient while retaining much of RoBERTa's performance.
- **Number of Layers**: DistilRoBERTa uses **6 transformer layers** (RoBERTa uses 12).
- **Hidden Size**: The hidden size is **768**.
- **Attention Heads**: It has **12 attention heads**.

- For Devanagari script:

  For this script, we used an already existing BERT model trained on the Nepali datasets. The model achieves an accuracy of 99.75% on the test dataset. It leverages a text-classification pipeline to classify text into sentiment categories, such as positive, negative, or possibly neutral, depending on the task-specific fine-tuning.

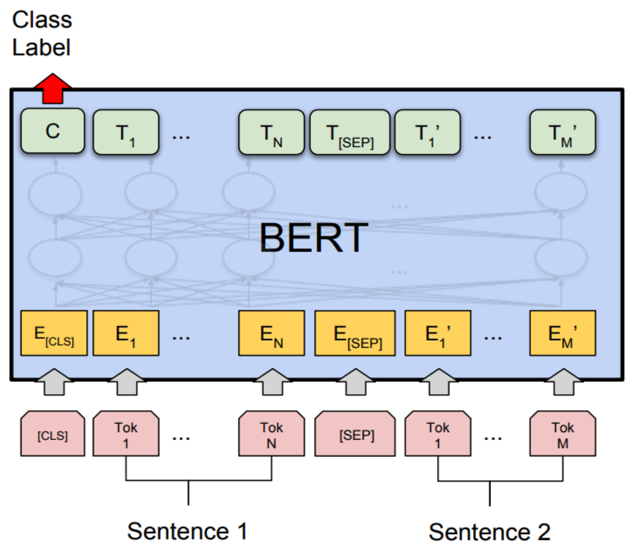


Figure 3.5.3 Architecture of BERT model

3.6 Software Specifications

3.6.1 Programming Languages and Libraries

Utilize Python programming language for data analysis, machine learning modeling, and visualization. Libraries such as pandas, numpy, matplotlib, NLTK (Natural Language Toolkit), SpaCy, TextBlob (for text pre-processing).

3.6.2 Machine Learning Tool

NLTK (Natural Language Toolkit)**:** For text preprocessing, tokenization, and stemming/lemmatization.

TextBlob**:** A simple library for basic NLP tasks including sentiment analysis.

Hugging Face Transformers**:** For pre-trained models like BERT, RoBERTa, or GPT-2 that can be fine-tuned for sentiment analysis.

3.6.3 Language detection and transliteration tools:

- langdetect: A lightweight Python library for detecting the language of a given text, based on Google's language detection algorithm
- google translate: A Python library that provides a simple interface to access the Google Translate API, enabling automatic translation of text between multiple languages.

3.7 Hardware Specifications

3.7.1 Computing Resources

CPU: A multi-core processor with sufficient processing power (e.g., Intel Core i5 or higher) for handling computational tasks involved in data analysis and modeling.

RAM: Adequate memory capacity (e.g., 8GB or higher) to accommodate large datasets and model training processes.

Storage: Sufficient storage space (e.g., SSD or HDD) for storing datasets, code files, and project-related documents.

3.7.2 Graphics Processing Unit (GPU)

GPU: Utilize a dedicated GPU (NVIDIA GeForce or AMD Radeon) for accelerating computation-intensive tasks, particularly for deep learning models and large-scale data processing.

CUDA or OpenCL support: Ensure compatibility with GPU-accelerated libraries and frameworks for machine learning and deep learning tasks.

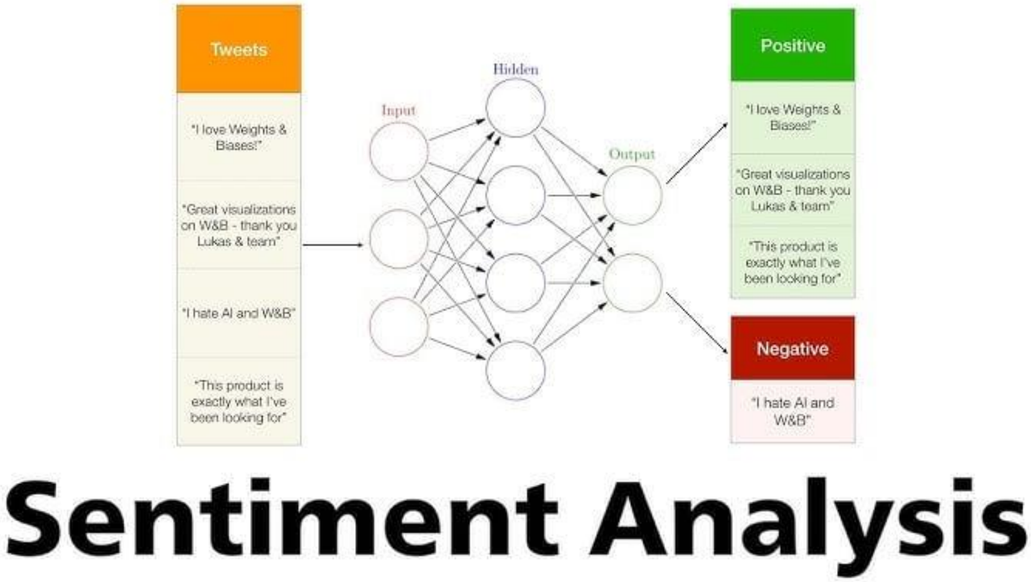


Figure 3.7.1 Sentiment Analysis

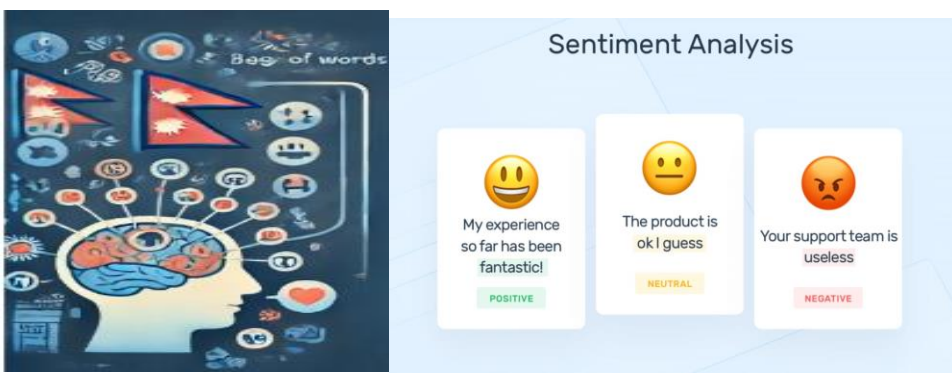


Figure 3.7.2 Sentiment Analysis classification

Chapter 4 Discuss on the Achievements

4.1 Web Scraping and Dataset Creation:

- Successfully web-scraped thousands of YouTube comments related to four prominent Nepali politicians, creating a novel and unique dataset tailored for sentiment and emotion analysis in political discourse.

4.2 Data Preprocessing:

- Translated Romanized Nepali text into Devanagari script, enabling accurate analysis of local language content.
- Cleaned and prepared the dataset, ensuring it was optimized for natural language processing (NLP) tasks.

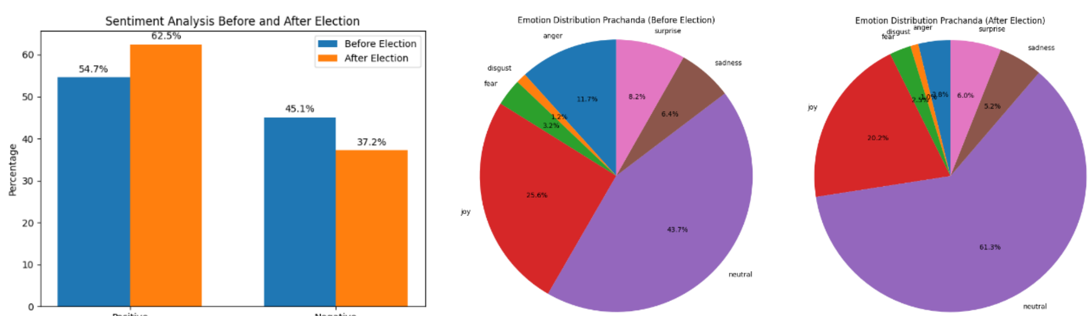
4.3 Application of Pretrained Models:

- Applied a pretrained BERT model for Devanagari text to analyze sentiments in Nepali comments.
- Used an English BERT model for sentiment classification in English comments.
- Performed emotion analysis on the English subset of the dataset using a pretrained RoBERTa model, identifying nuanced emotional expressions.

For Sher B Deuba:

Figure 4.3.1 Sentiment and Emotion analysis of data for Sher Bahadur Deuba

For Puspa Kamal Dahal(prachanda):



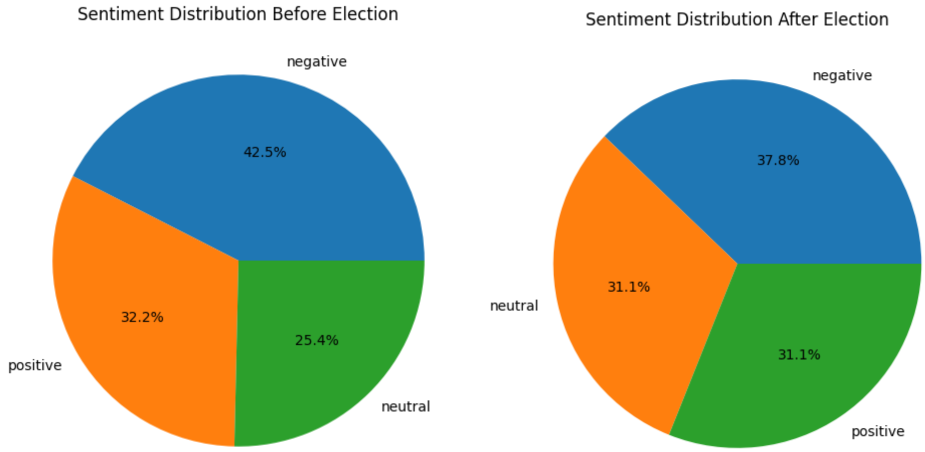
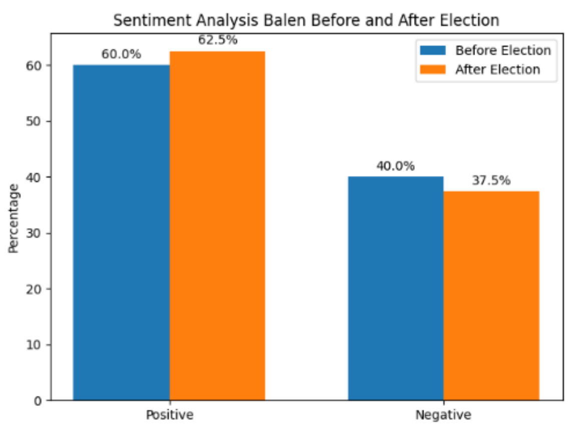
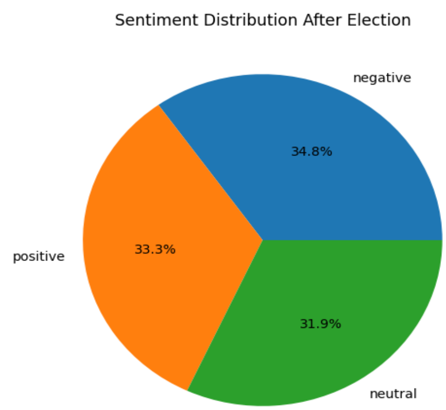
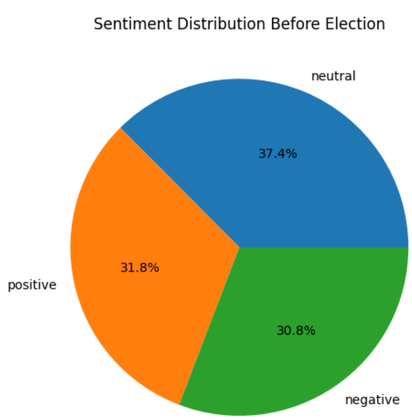


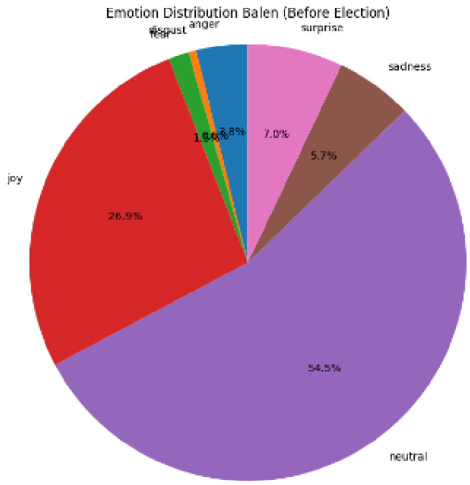
Figure 4.3.2 Sentiment and Emotion analysis of data for Prachanda

For Balen:









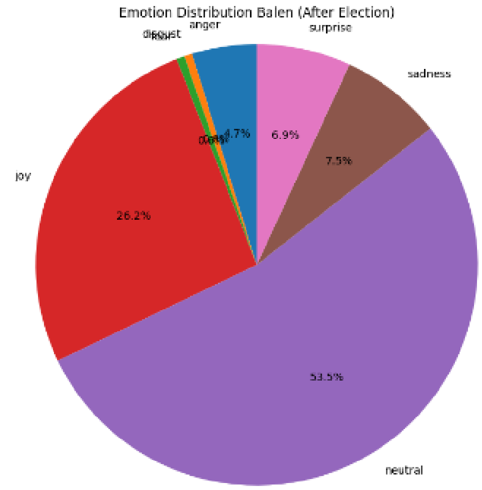
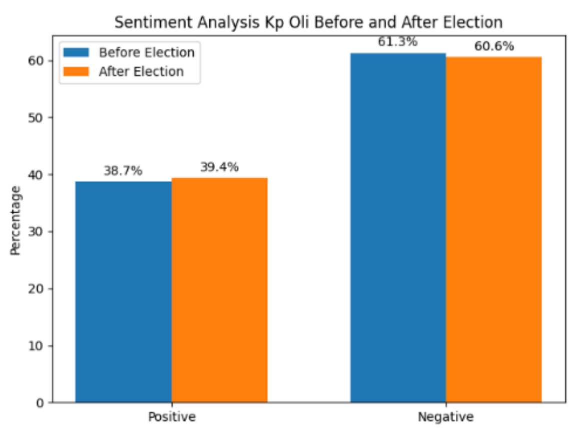
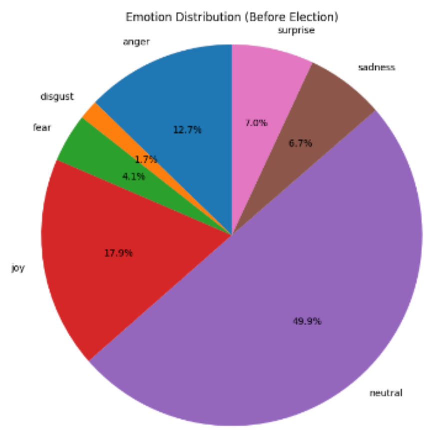
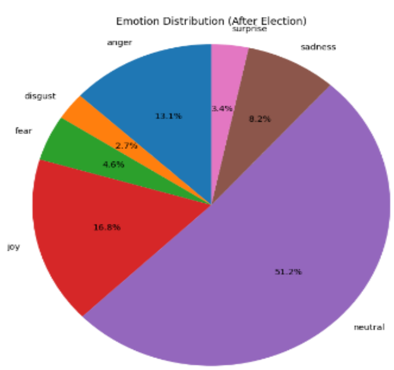


Figure 4.3.3 Sentiment and Emotion analysis of data for Balen

For KP SHARMA OLI:







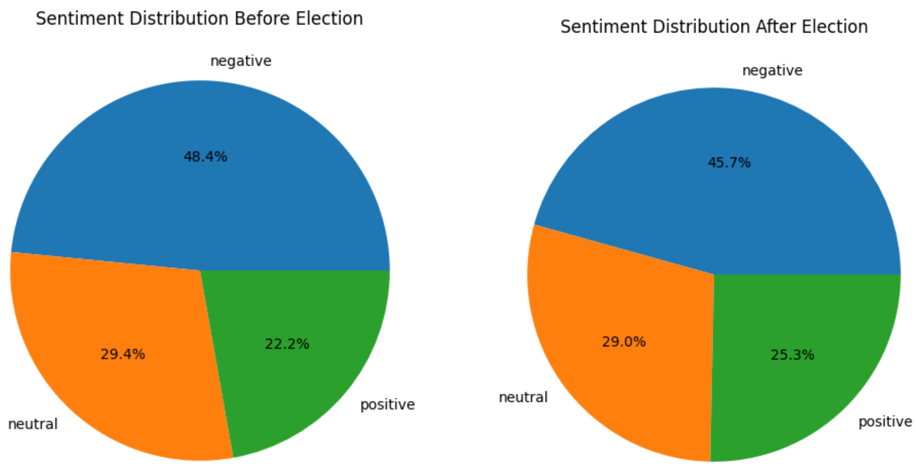


Figure 4.3.4 Sentiment and Emotion analysis of data for KP Sharma Oli

4.4 Exploratory Data Analysis (EDA):

- Conducted extensive EDA to uncover patterns and insights within the dataset, including trends in sentiment and emotional expressions.

- Categorized the dataset into two parts—comments posted before and after the election—and performed comparative EDA to observe shifts in public sentiment and discourse.

Key Insights:

- Found that public sentiment plays a critical role in shaping political discourse in Nepal, with significant variations observed before and after the election.

4.5 Model performance and evaluation:

We have used three models for our project, one of them is used for Devanagari whereas the other two are used for English. Now the performance metrics of these 3 models are given below:

S.N	Model name	Accuracy	Precision	Recall	F1 Score
1.	DistilRoBERTa Model (Emotion detection for English)	0.710	0.749	0.710	0.685
2.	DistilBERT Model (Sentiment analysis for English)	0.913	0.898	0.930	0.914
3.	dpkrm/multilingual-uncased–BERT Model (Sentiment analysis for Devanagari)	0.760	0.810	0.781	0.757

Table 4.5.1 Model performance and evaluation

:

Chapter 5 Conclusion and Recommendation

5.1 Limitations

- **Dataset Constraints**:
    - Lack of labeled Nepali political datasets; challenges with script variations.
    - Imbalanced data leading to biased predictions.
- **Model Limitations**:
    - Difficulty in detecting sarcasm, idioms, and informal grammar.
    - Dependency on pre-trained models not optimized for Nepali political discourse.
- **Scalability and Real-time Processing**:
    - Inefficient handling of large-scale real-time data.
    - High computational demands for deep learning models.
- **Evaluation Metrics**:
    - Limited benchmarks and focus on interpretability.

5.2 Future Enhancements

- **Dataset Collection**:
    - Collaborate for annotated datasets and use data augmentation to address underrepresentation.
- **Model Optimization**:

- Fine-tune domain-specific models and add sarcasm detection capabilities.

- **Real-time Processing**:
  - Employ distributed systems and optimize for low-latency predictions.
- **Visualization**:
  - Develop interactive dashboards and enhance error analysis for better insights.
- **Scalability**:
  - Implement cloud-based, modular pipelines for evolving data and requirements.

References

GeeksforGeeks. "*Python sentiment analysis using VADER*". Retrieved September 14, 2024, from https://www.geeksforgeeks.org/python-sentiment-analysis-using-vader/

Haselmayer, M., & Jenny, M. " Sentiment analysis of political communication: Combining a dictionary approach with crowdcoding". Retrieved September 12, 2024 https://doi.org/10.1007/s11135-016-0412-4

Bakliwal, A., Foster, J., van der Puil, J., O'Brien, R., Tounsi, L., & Hughes, M. "Sentiment analysis of political tweets: Towards an accurate classifier. *DORAS*." Retrieved September 12, 2024, from https://doras.dcu.ie/19962/

Mullen, T., & Malouf, R. "A preliminary investigation into sentiment analysis of informal political discourse". Retrieved September 12, 2024, from https://cdn.aaai.org/Symposia/Spring/2006/SS-06-03/SS06-03-031.pdf

Rahman Jim, J., Talukder, M. A. R., Malakar, P., Kabir, M. M., Nur, K., & Mridha, M. F. "Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review". *ScienceDirect*. Retrieved September 13, 2024, from

https://www.sciencedirect.com/science/article/pii/S2949719124000074

IBM. (2024). "Semantic analysis". *IBM*. Retrieved September 13, 2024, from

https://www.ibm.com/topics/sentiment-analysis

Gupta, S. (2018, January 7). "Sentiment analysis: Concept, analysis and applications." *Medium*. Retrieved September 14, 2024, from

https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17

Shivanandhan, M. (2020, September 30). "What is sentiment analysis? A complete guide for beginners". *freeCodeCamp*. Retrieved September 14, 2024, from https://www.freecodecamp.org/news/what-is-sentiment-analysis-a-complete-guide-to-for-beginners/

Lamba, M., & Margam, M. (2022, April). Sentiment analysis. *ResearchGate*. Retrieved September 15, 2024, from

https://www.researchgate.net/publication/360112442_Sentiment_Analysis

dpkrm. (n.d.). *NepaliSentimentAnalysis* [Hugging Face model]. Retrieved December 5, 2024, from https://huggingface.co/dpkrm/NepaliSentimentAnalysis

Yash22222. (n.d.). *Devanagari-Sentiment-Analysis* [GitHub repository]. Retrieved November 22, 2024, from https://github.com/Yash22222/Devanagari-Sentiment-Analysis

AlexTheAnalyst. (n.d.). *Twitter Scraper V8* [Jupyter Notebook]. Retrieved December 15, 2024, from https://github.com/AlexTheAnalyst/PythonCode/blob/master/Twitter%20Scraper%20V8.ipynb

Alotaibi, S. S., & Alotaibi, B. S. (2024). Sentiment analysis of Arabic tweets using deep learning models. *Journal of King Saud University - Computer and Information Sciences*. Advance online publication. Retrieved November 30, 2024, from https://doi.org/10.1016/j.jksuci.2024.04.001

Hugging Face. (2022, March 2). *BERT 101 - State Of The Art NLP Model Explained*. Retrieved December 10, 2024, from https://huggingface.co/blog/bert-101

Shushant. (n.d.). *nepaliBERT* [Hugging Face model]. Retrieved November 18, 2024, from https://huggingface.co/Shushant/nepaliBERT