Final Report: Sentiment Analysis

Contributors:

Aayush Raju Bade (B23CS1084), Dhruv Kumar Tiwari (B23CM1014)
Vaibhav Singh (B23EE1101), Shyam Charan (B23EE1012)
Yateen (B23CS1082), Gnani Prakash Y. (B23CS1080)
Srinivas (B23CS1010)
April 14, 2025

1. Introduction

Sentiment Analysis is the process of applying Machine Learning (ML) techniques to determine the emotional tone behind textual content—typically classifying it as positive, negative, or neutral. In this project, we developed a sentiment analysis model aimed at classifying text-based data accurately. Our focus was on improving classification accuracy using various ML models, preprocessing techniques, and embeddings.

2. Objectives

The primary objectives of our project were:
- To preprocess and analyze textual data for sentiment classification.
- To compare different machine learning models based on performance metrics.
- To evaluate the models using accuracy, precision, recall, and F1-score.
- To identify key challenges and implement solutions to improve sentiment classification.

3. Methodology

3.1 Dataset Selection

We utilized a dataset provided by **Prof. Anand Mishra**, complemented by additional datasets to better understand the problem space:
- IMDB Movie Reviews Dataset
- Twitter Sentiment Analysis Dataset
- Custom datasets sourced from online platforms

3.2 Data Preprocessing

To prepare the textual data for analysis, we applied the following preprocessing techniques:
- Tokenization
- Stopword Removal
- Lemmatization and Stemming
- Vectorization techniques: TF-IDF, Word2Vec, and BERT embeddings

3.3 Model Selection

We experimented with multiple traditional machine learning models:
- Naïve Bayes
- Logistic Regression

- Support Vector Machine (SVM)
- Decision Tree

3.4 Evaluation Metrics

To evaluate model performance, we used the following metrics:
- Accuracy
- Precision, Recall, and F1-score
- Confusion Matrix

---

4. Progress and Results

During the project, we:
- Conducted research on various datasets to understand sentiment analysis use cases.
- Explored and benchmarked existing models.
- Developed a preprocessing pipeline for text cleaning and vectorization.
- Implemented and evaluated multiple ML models for sentiment classification.

Benchmarked Model Accuracies (from literature):
- Logistic Regression – 90%
- SVM – 90%
- Decision Tree – 75–86%

Final Training Accuracies (our models):
- **Logistic Regression:** 69.89%
- **Support Vector Machine (SVM):** 55.44%
- **Decision Tree:** 71.71%

These results reflect the challenges of working with real-world data and the limitations of traditional ML models compared to deep learning architectures.

---

5. Challenges and Roadblocks

While working on this project, we encountered the following challenges:
- Difficulty understanding some ML and NLP concepts in depth.
- Choosing the right model architecture for our specific dataset.
- Managing computational constraints when using advanced embeddings or large datasets.

We addressed these through team discussions, faculty guidance, and leveraging online resources and open-source tools.

---

6. Conclusion and Future Work

This project provided us with hands-on experience in building sentiment analysis models from scratch. While our results demonstrate moderate accuracy with traditional ML models, there is significant scope for improvement using deep learning models like LSTMs or Transformers in the future.

In future iterations, we aim to:
- Integrate deep learning models such as Bi-LSTM or BERT-based classifiers.
- Perform hyperparameter tuning for improved performance.

- Enhance visualization of results and sentiment trends.
- Deploy the model via a simple web interface or API for real-time predictions.

7. References

- IMDB Movie Reviews Dataset
- Twitter Sentiment Analysis Dataset
- Concepts and tutorials on word embeddings (TF-IDF, Word2Vec, BERT)
- Scikit-learn documentation and ML guides
- Academic papers on sentiment classification models