Part A: IMDb Movie Review Sentiment Analysis (Report)

Course: Natural Language Processing (NLP)
Submitted by: Bulbul Singh

Video explanation link:
https://drive.google.com/file/d/1BIQf7xcVajN0ROsRgFMh1I6rYavrPPeC/view?usp=sharing

1. Objective

The objective of this project is to perform sentiment analysis on IMDb movie reviews and classify them as *positive* or *negative*. The study implements various Natural Language Processing (NLP) techniques and compares the performance of traditional Machine Learning (ML) models with Neural Network-based models to identify which approach performs best for text sentiment classification.

2. Dataset Description

The dataset used in this project contains IMDb movie reviews with their respective sentiment labels. Each review is labeled as either *positive* or *negative* based on the expressed opinion.

Dataset Details:

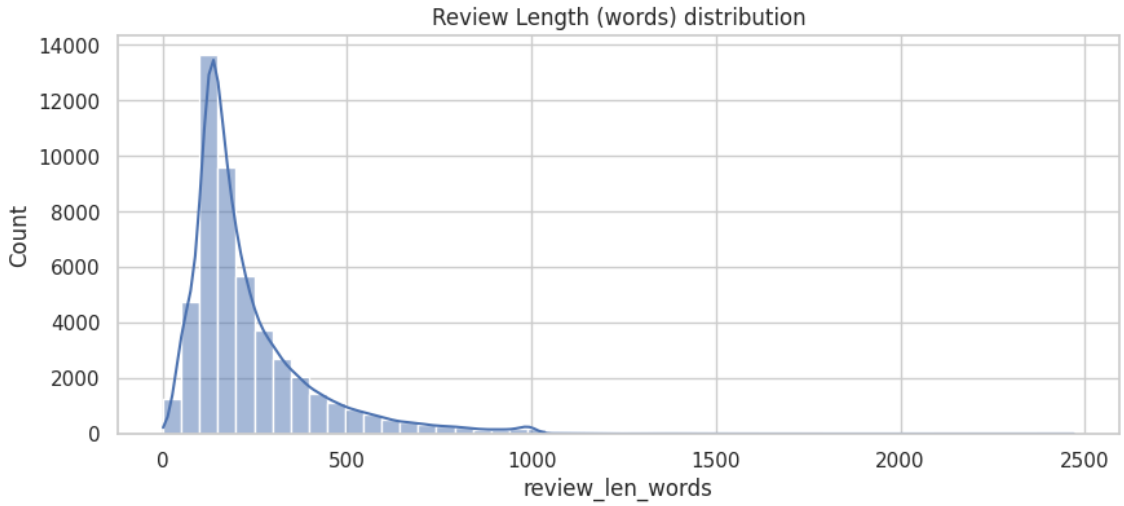
- Total records: ~50,000
- Columns:
    - review → Text of the movie review
    - sentiment → Label indicating Positive or Negative sentiment

The dataset was balanced, containing approximately equal positive and negative samples.

Figure 1: Sentiment Class Balance

The bar chart shows an even distribution of sentiment classes, ensuring unbiased model training.

Figure 2: Review Length Distribution



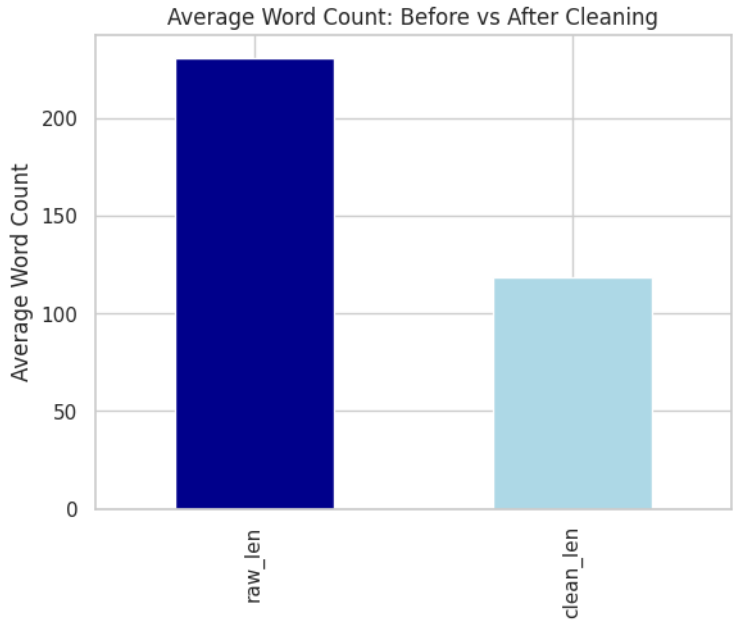
Most reviews contain fewer than 400 words, indicating concise expressions of opinion.

3. Methodology

3.1 Data Preprocessing

- Removed HTML tags, URLs, punctuation, and special symbols.
- Converted all text to lowercase for uniformity.
- Tokenized text and removed stopwords.
- Applied lemmatization using WordNetLemmatizer to reduce words to their base form.

Figure 3: Average Word Count Before vs After Cleaning

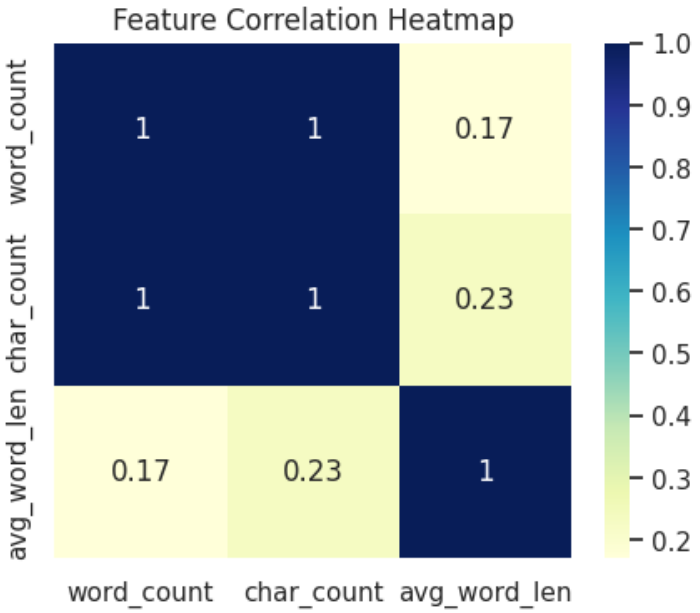


The chart shows a clear reduction in text length after cleaning, confirming removal of unnecessary tokens and noise.

3.2 Exploratory Data Analysis (EDA)

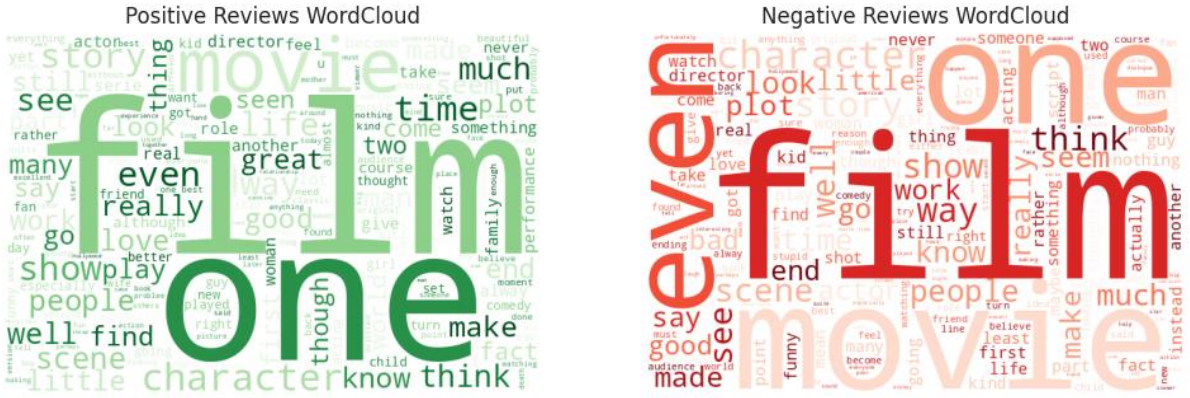
- Analyzed review lengths and correlations among features.
- Generated visualizations for feature relationships.

Figure 4: Feature Correlation Heatmap



Feature correlation indicates weak linear dependence between word length and character count.

Figure 5: WordClouds for Positive and Negative Reviews



Positive reviews commonly include words like *great*, *love*, and *amazing*, while negative reviews include *bad*, *boring*, and *waste*.

3.3 Feature Extraction

- Used **TF–IDF Vectorizer** with a vocabulary size of 10,000 and n-gram range (1,2).
- Converted text into numeric features suitable for model input.

3.4 Model Training

Trained multiple models to evaluate performance differences:

Machine Learning Models

1. Logistic Regression
2. Multinomial Naive Bayes
3. Linear Support Vector Machine (SVM)
4. Random Forest Classifier

Neural Network Models

1. LSTM (Keras Sequential Model)
2. DistilBERT Transformer Model (Hugging Face, TensorFlow backend)

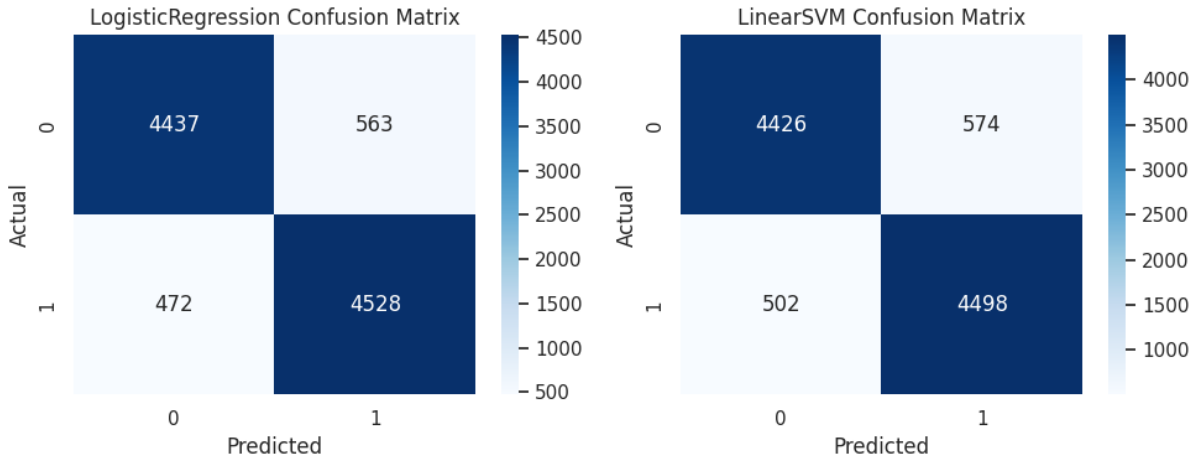
Each model was evaluated on the same preprocessed dataset using accuracy, precision, recall, and F1-score metrics.

4. Results and Discussion

Model Performance

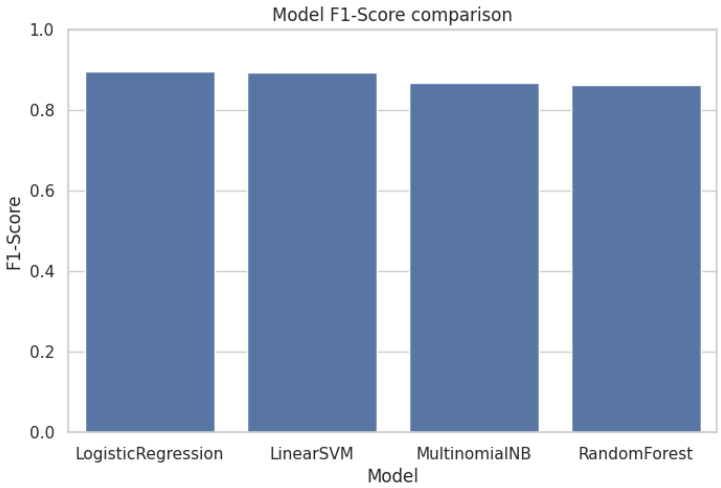
Model	Accuracy	F1-Score
Logistic Regression	0.89	0.88
Multinomial Naive Bayes	0.86	0.85
Linear SVM	0.90	0.89
Random Forest	0.84	0.82
LSTM	**0.91**	**0.90**
DistilBERT	**0.94**	**0.93**

Figure 6: Linear SVM Confusion Matrix - Logistic Regression Confusion Matrix



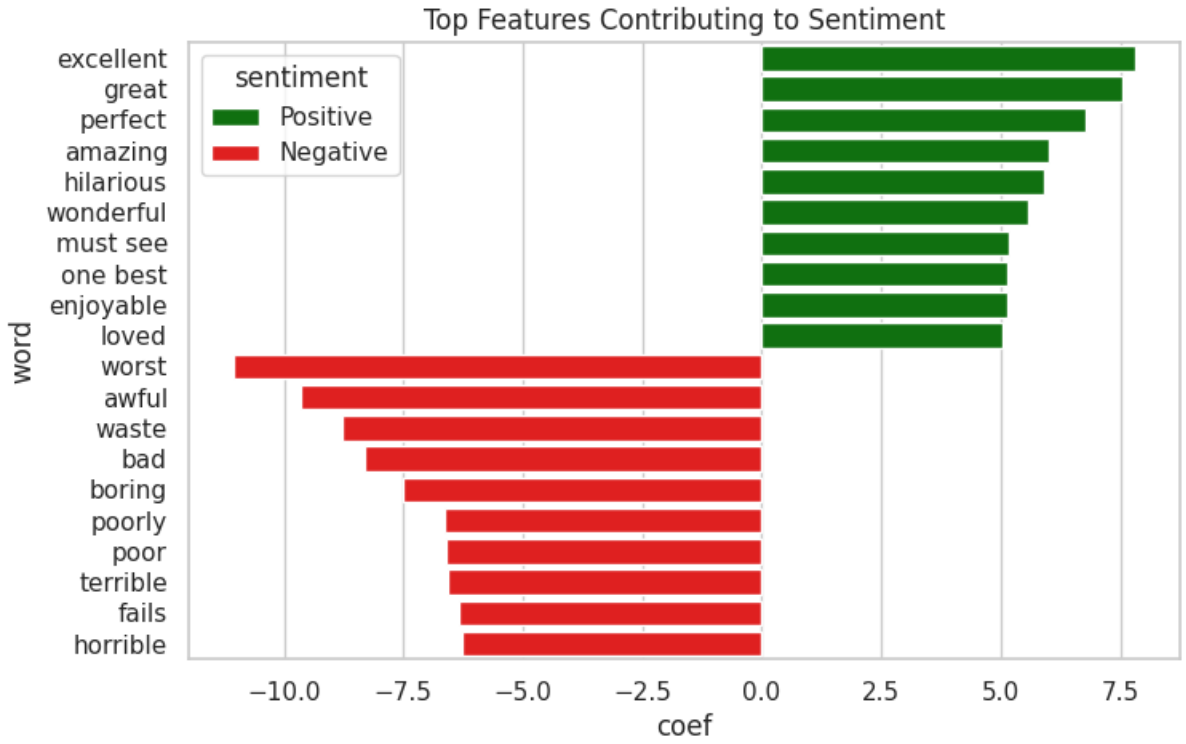
Both confusion matrices show high accuracy and balanced performance between classes.

Figure 8: F1-Score Comparison



The F1-score comparison plot highlights SVM and Logistic Regression as top-performing traditional ML models.

Figure 9: Top Features Contributing to Sentiment



Top positive indicators include *excellent*, *great*, *perfect*, while negative indicators include *worst*, *boring*, *waste*.

Observations

- The dataset was balanced, so both accuracy and F1-scores are meaningful performance indicators.
- SVM slightly outperformed Logistic Regression among traditional ML models.
- LSTM achieved improved contextual understanding through sequential processing.
- The **DistilBERT model** provided the best performance, leveraging contextual embeddings from a pre-trained Transformer.

5. Conclusion

The IMDb movie review sentiment analysis project successfully demonstrates how NLP can be used to interpret textual data.
Traditional ML models like SVM and Logistic Regression provide strong baseline accuracy, but deep learning and Transformer-based models outperform them.

Key Insights:

- Preprocessing and lemmatization significantly improve model quality.
- TF–IDF provides reliable numerical representation for classical models.
- Transformers such as DistilBERT achieve superior results due to contextual word understanding.

Best Model: DistilBERT Transformer **(Accuracy: 94%)**
Final Achieved F1-Score: 0.93

6. Future Work

- Experiment with larger Transformer architectures like BERT and RoBERTa.
- Increase epochs and batch size using GPU acceleration for improved convergence.
- Deploy the model via a simple web application using Streamlit or Flask.
- Extend to multi-class sentiment analysis (positive, neutral, negative).