Movie Review Sentiment Analysis

By-Aiyan Shamshad (229302265)

Ketan Mathur (229302058)

Prashuk Jain (229302001)

# Objective

* Perform sentiment analysis on movie reviews using traditional machine learning models.
* Classify reviews as positive, negative, or neutral.
* Demonstrate rule-based sentiment analysis using VADER.
* Include data visualization for better insights.
* Benchmarked against SVM, KNN, Logistic Regression and Naïve Bayes.
* Streamlit was used to build an interactive web interface for real-time sentiment prediction. (Add on after mte)
* Explainable AI (XAI) techniques were applied to visualize the contribution of individual words in sentiment classification. (Add on after mte)

# Dataset

* The dataset was collected using the TMDB API.
* Reviews were cleaned, processed, and balanced before storing in the final dataset.
* Final dataset statistics:
  + 1200 Positive reviews
  + 1200 Negative reviews
  + 500 Neutral reviews
* File Used: balanced\_reviews.csv(finalpedata.csv)
* Columns:
  + movie\_name – Name of the movie
  + one\_movie\_review – Actual review
  + sentiment – Target label (Positive / Negative / Neutral)

A graph of positive and negative expressions

AI-generated content may be incorrect.

# Methodology

* **Data Cleaning** involved removing duplicates and handling missing values.
* **Visualizations** were created including sentiment distribution count plots, word clouds, and review length histograms.
* **Text preprocessing** steps included lowercasing, punctuation removal, stopwords removal, tokenization, and lemmatization using WordNetLemmatizer. A clean\_review column was added.
* **TF-IDF Vectorizer** with 5000 features was used for **feature extraction**.
* The dataset was split into **80% training and 20% testing** using train\_test\_split.
* Models used for classification included **Logistic Regression, Naive Bayes, Support Vector Machine (Linear SVC), Random Forest, and K-Nearest Neighbors**.
* **Streamlit** was used to build **an interactive web interface** for real-time sentiment prediction.
* **Explainable AI (XAI) techniques** were applied to visualize the contribution of individual words in sentiment classification.

# VADER Sentiment Demo

* Used nltk.sentiment.vader.SentimentIntensityAnalyzer
* Example sentence: “This movie was super boring and a waste of time”
* Output: Compound score ≈ -0.79 (Negative)
* Note: VADER used for demo purposes, not for model training

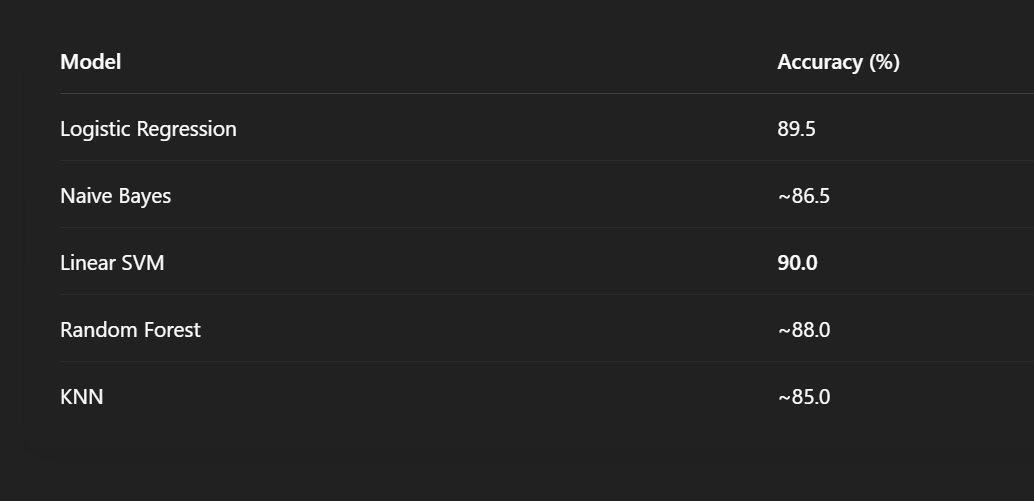
# Evaluation

Metrics used: Accuracy, Confusion Matrix, Classification Report

Linear SVM showed the highest accuracy (~90%)

# Results Summary

* **Linear SVM consistently gave the best performance** across both notebooks.
* Streamlit and Explainable AI enhanced the project’s usability and transparency



# Conclusion

* **Linear SVM outperformed all other models**
* Proper preprocessing significantly improved model accuracy
* Rule-based analysis like **VADER helps in understanding sentiment** but is not as reliable as machine learning models for prediction
* **Streamlit made the model accessible for end-users to test custom reviews**
* **Explainable AI increased model interpretability and trust**

# Tools Used

* Python
* Pandas
* sklearn
* nltk
* matplotlib
* seaborn
* WordCloud
* VADER
* Streamlit
* Explainable AI (e.g., SHAP, LIME, etc.)
* TMDB API