**Sentiment Analysis on movie reviews using NLP**

**1. Introduction**

**Objective:** This project aims to develop a sentiment analysis pipeline for movie reviews on IMDB. This involves classifying sentiment into positive and negative categories using Natural Language Processing (NLP) techniques and machine learning models. The project aims to build an efficient model for sentiment classification, compare the performance of traditional methods, and provide insights into the effectiveness of these methods.

**2. Data Description**

**Dataset Source:**

* IMDB Movie Ratings Sentiment Analysis
* [Dataset on Kaggle](https://www.kaggle.com/datasets/yasserh/imdb-movie-ratings-sentiment-analysis)

**Data Overview:**

* **Number of Instances:** 39,723
* **Features:**
  + text: The review text.
  + label: Sentiment label (0 for negative, 1 for positive).

**Data Splits:**

* **Train + Validation Set:** 85%
* **Test Set:** 15%
* **Validation Set:** 15%

**3. Baseline Experiments**

**Initial Steps:**

1. **Data Import and Inspection:**
   * Loaded data from CSV file.
   * Checked for null values and data types.
2. **Data Exploration (EDA):**
   * Verified the shape and distribution of data.
   * Checked for duplicates and removed them.
   * Analyzed class imbalance and word count distribution.

**Baseline Analysis Results:**

* **Class Distribution:** Balanced (almost equal number of positive and negative reviews).

A blue and orange pie chart

Description automatically generated

* **Word Count Distribution:** Right-skewed with few anomalies.

A graph of a number of columns

Description automatically generated

**4. Advanced Experiments**

**4.1 Data Preprocessing**

**Text Cleaning:**

* Converted text to lowercase and removed punctuation.

**Tokenization and Lemmatization:**

* Used NLTK (Natural Language Tool Kit) for tokenization and lemmatization.
* Implemented part-of-speech tagging for accurate lemmatization.

**Stop Words Removal:**

* Removed common stop words using NLTK’s stop words list.

**Feature Extraction:**

* Used TF-IDF vectorizer to transform text data into feature vectors.

**Text before and after applying analysis:**

|  |
| --- |
| Even though I have great interest in Biblical movies, I was bored to death every minute of the movie. Everything is bad. The movie is too long, the acting is most of the time a Joke and the script is horrible. I did not get the point in mixing the story about Abraham and Noah together. So if you value your time and sanity stay away from this horror. |
| even though great interest biblical movie bore death every minute movie everything bad movie long acting time joke script horrible get point mix story abraham noah together value time sanity stay away horror |

**4.2 Model Development and Tuning**

**Train-Test Split:**

* Split the dataset into training, validation, and test sets.

**Models Developed:**

1. **Logistic Regression:**
   * **Parameter Tuning:** Used GridSearchCV to find the best parameters.
   * **Best Parameters:** C values of [0.01, 0.1, 1, 10, 100] and solvers 'liblinear' and 'saga'.
   * **Validation Accuracy:** 88.67%
   * **Test Accuracy:** 89.31%
2. **Naive Bayes:**
   * **Parameter Tuning:** GridSearchCV was used to find the best alpha values.
   * **Best Parameters:** alpha values of [0.01, 0.1, 1, 10, 100].
   * **Validation Accuracy:** 86.04%
   * **Test Accuracy:** 86.02%

**Performance Comparison:**

**Comparative Analysis:**

Compared performance of Logistic Regression and Naive Bayes using metrics such as accuracy, confusion matrix, and F1 score.

* **Classification Report:**





* **Confusion Matrix Comparison:**

A close-up of a graph

Description automatically generated

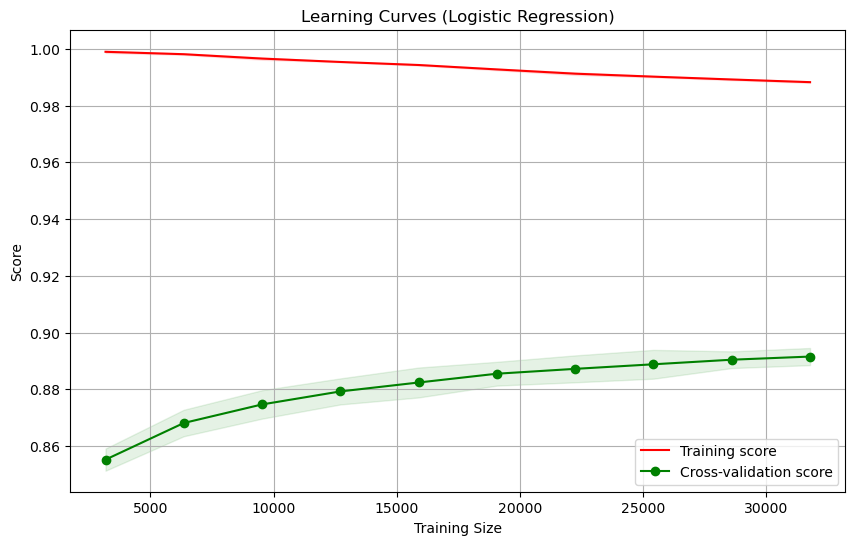
**5. Overall Conclusion**

**Best Model:**

* Logistic Regression is identified as the best-performing model with the highest accuracy and ROC AUC score.

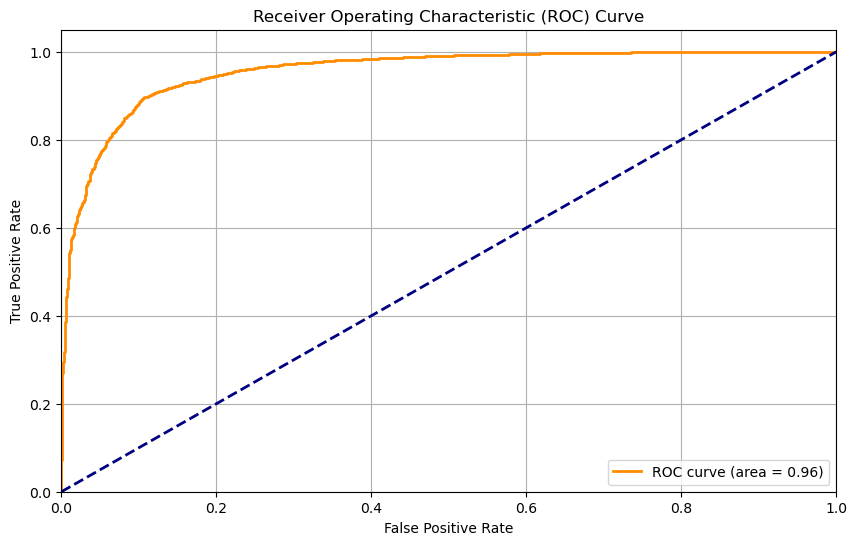
**Learning Curves:**

* Generated learning curves for Logistic Regression to analyze model performance with varying training sizes.



**ROC Curve:**

* **ROC AUC Score:** 0.95
* **ROC Curve Plot:** Displays the trade-off between true positive rate and false positive rate for the Logistic Regression model



**Insights:**

* Logistic Regression outperforms Naive Bayes in both validation and test accuracy.
* Learning curves indicate that the model benefits from more training data.
* The ROC curve confirms that the Logistic Regression model provides an excellent trade-off between sensitivity and specificity.

**Additional Information**

**Tools and Libraries Used:**

* Python, pandas, numpy, matplotlib, seaborn, NLTK, scikit-learn.

**External Resources:**

* Kaggle dataset.
* NLTK for text processing and stop words.

**Reflection Questions**

1. **Biggest Challenge:**
   * Implementing effective text preprocessing and parameter tuning that significantly impacts model performance.
2. **Insights Gained:**
   * Gained a deeper understanding of how different NLP techniques and machine learning models can be applied to sentiment analysis and the importance of parameter tuning for improving model performance.