**A logo with text and numbers

Description automatically generated**

**Lebanese American University**

**Machine Learning Course Project**

**Project Title**

Sentiment Analysis on Opinions and Feedback

**Submitted by:**

* **Student Name(s)**: [Abdullah Jrad]
* **Student ID(s)**: [202207733]
* **Course Name**: [Introduction to Machine Learning]
* **Instructor’s Name**: [Seifedine Kadry]
* **Submission Date**: [6/Nov/2024]

**Abstract**

This project aims to create a machine learning model(s) that classifies text as positive, or negative. The dataset used for this study is the IMDB Movie Reviews Dataset, which contains 50,000 labeled reviews. After preprocessing the data by cleaning and converting text into numerical features, a Logistic Regression model is trained to predict sentiments. The project focuses on how machine learning can be used to analyze text data efficiently.

The source codes with deployment of flask app can be found on github through this link: [CODE](https://github.com/Abedsay/Machine-Learning-in-Sentiment-Analysis)

**Keywords**

Text Classification - Sentiment Analysis - Language Processing

**Section 1: Project Proposal**

**Objective:**

The goal of this project is to build and compare multiple machine learning models (Logistic Regression and Random Forest) to classify text data into positive or negative sentiments. By analyzing the sentiments of text reviews, this project aims to demonstrate the power of machine learning in understanding public opinion, which can be useful in various industries such as e-commerce, entertainment, and social media analysis.

**Dataset:**

Source: IMDB Dataset on Kaggle.

Size: The dataset contains 50,000 text reviews labeled as positive or negative, making it suitable for classification.

The dataset is clean, well-labeled, and widely used for sentiment analysis tasks, which ensures consistency in evaluation.

**Approach:**

- Preprocessing:

The text reviews will be converted to lowercase, and punctuation will be removed to clean the data. The sentiment labels (positive, negative) will be encoded into binary values: 1 for positive and 0 for negative.

- Feature Extraction:

TF-IDF (Term Frequency-Inverse Document Frequency) will be used to convert text data into numerical features.

The vocabulary size will be limited to 5,000 words to simplify the analysis and reduce computation.

- Model:

Logistic Regression will be used as the classification model, which performs well for text data.

In addition to Logistic Regression, Random Forest will be used to explore the benefits of ensemble learning for sentiment classification. Random Forest’s ability to handle feature interactions and reduce overfitting makes it a valuable addition for comparison.

**Section 2: Data Exploration and Preprocessing**

**Data Cleaning:**

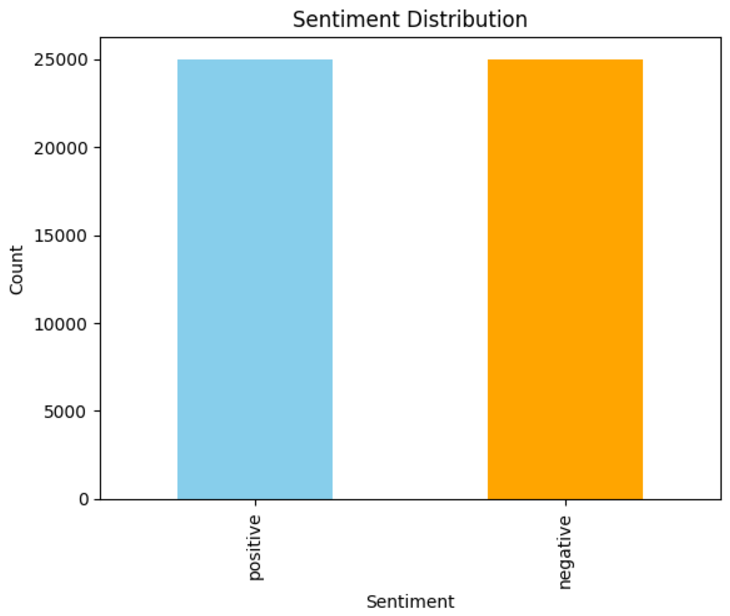
Removed punctuation: The reviews were stripped of all punctuation to standardize the text format.

Converted text to lowercase: Ensured uniformity across the dataset to avoid mismatches in text processing.

Encoded sentiment labels: Converted the target variable (sentiment) into binary values (positive → 1, negative → 0) for machine learning compatibility.

**Exploratory Data Analysis (EDA):**

Class Distribution: The dataset contained an equal number of positive and negative reviews.



**Feature Engineering:**

Applied TF-IDF to extract numerical features from the text reviews. I limited it to the top 5,000 most frequent words for efficiency without sacrificing patterns.

**Data Splitting:**

The dataset was divided into 70% training and 30% testing sets, 35,000 samples for training and 15,000 for testing.

Training set: 35,000 reviews (17,500 positive, 17,500 negative).

Testing set: 15,000 reviews (7,500 positive, 7,500 negative).

**Section 3: Model Selection and Training**

**Model Choice:**

For this project I selected Logistic Regression as the primary model for sentiment classification. It is a good algorithm for binary classification problems. It is efficient, making it a suitable choice for understanding the features (word frequencies) on the predicted sentiment. Random Forest was selected as a secondary model for comparison due to its ability to handle feature interactions, overfitting, and interpretability via feature importance.

Also, the dataset’s size and simplicity of the feature set created using the TF-IDF supports this choice, due to its ability to prioritize important terms over common ones.

**Training:**

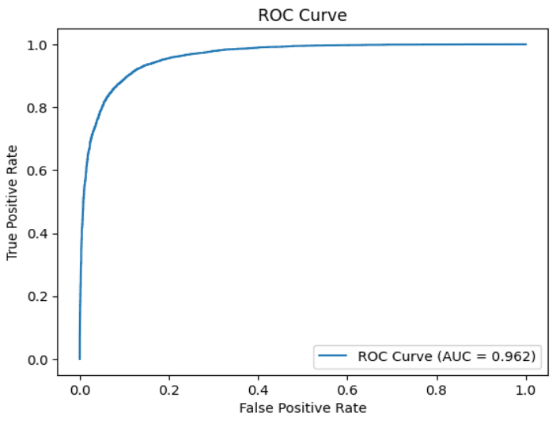
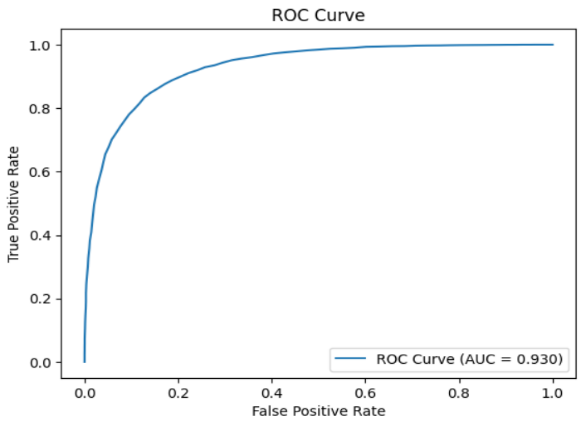
* Loading preprocessed data.
* Splitting the data into training (70%) and testing (30%) sets.
* Training the model using the training data (X\_train and y\_train).
* The hyperparameter n\_estimators was set to 100 and random\_state to 42.

**Hyperparameter Tuning:**

Optimize Random Forest hyperparameters (e.g., max\_depth, min\_samples\_split) to further enhance its accuracy and robustness.

**Evaluation Metrics:**

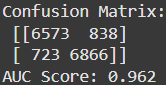
The models were evaluated using the following metrics:

1. Accuracy: The percentage of correctly predicted reviews.
2. Confusion Matrix: A matrix showing the model's true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).
3. AUC\_Score: The model’s ability to distinguish between positive and negative.

**Section 4: Evaluation and Analysis**

**Performance Evaluation (LR):**

- Accuracy:

The accuracy of the model on the test set was 0.90. This indicates that the model correctly classified sentiments for approximately 90% of the reviews.

- Confusion Matrix:

The confusion matrix revealed this:

TP: 6,866 (positive reviews correctly classified as positive)

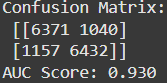
TN: 6,573 (negative reviews correctly classified as negative)

FP: 838 (negative reviews misclassified as positive)

FN: 723 (positive reviews misclassified as negative)

- AUC Score:

The score of 0.962 shows the model’s excellent ability to separate positive and negative sentiments. The closer to 1 the better

**Performance Evaluation (RF):**

- Accuracy: 85.0%.

- Confusion Matrix:

True Positives (TP): 6,432 (correctly predicted positive reviews)

True Negatives (TN): 6,371 (correctly predicted negative reviews)

False Positives (FP): 1,040 (negative reviews misclassified as positive)

False Negatives (FN): 1,157 (positive reviews misclassified as negative)

- AUC Score: 0.930.

**Model Comparison:** Logistic Regression achieved slightly higher accuracy and AUC Score compared to Random Forest. Also, execution in terms of execution times where LR took 1m18s and RF took 3m.

**Comparison Table**

|  |  |  |
| --- | --- | --- |
| **Metric** | **Logistic Regression** | **Random Forest** |
| **Accuracy** | 90.0% | 85.0% |
| **AUC Score** | 0.962 | 0.930 |
| **True Positives (TP)** | 6,866 | 6,432 |
| **True Negatives (TN)** | 6,573 | 6,371 |
| **False Positives (FP)** | 838 | 1,040 |
| **False Negatives (FN)** | 723 | 1,157 |
| **Execution Time** | ~3m | ~1m 20s |

**Error Analysis**

- Misclassifications:

Despite the improvements from TF-IDF, misclassifications occurred due to:

Complex Vocabulary: Words or phrases with multiple meanings (e.g. "not bad").

Mixed Sentiments: Reviews containing both positive and negative sentiments (e.g., "This part was great, but that part was terrible").

- Insights from these errors: (**Section 5: Interpretability and Insights**)

The model still struggles with sarcasm or tone, and Context Relationships: TF-IDF focuses on individual terms or bigrams but does not fully understand the relationships between phrases (e.g., "It wasn’t bad at all" being positive).

**Section 6: Conclusion and Future Work**

**Summary of Findings:**

This project successfully developed a Logistic Regression model and Random Forest model to classify IMDB movie reviews into positive or negative sentiments. The models demonstrated strong performance with:

LR:

* Accuracy: 90%
* AUC Score: 0.962, excellent ability to distinguish between positive and negative sentiments.
* The confusion matrix highlighted balanced classification, with 6,866 true positives and 6,573 true negatives, alongside manageable false positive and false negative rates.

RF:

* Accuracy: 85.0%
* AUC Score: 0.930, excellent ability to distinguish between positive and negative sentiments.
* The confusion matrix highlighted balanced classification, with 6,432 true positives and 6,371 true negatives, alongside manageable false positive and false negative rates.

These results validate the effectiveness of machine learning models, particularly Logistic Regression with TF-IDF, for sentiment analysis tasks. Logistic Regression achieved higher accuracy and AUC scores, and better execution time, making it the better choice.

**Limitations:**

While the model performed well, several limitations were identified:

* Context Understanding: Despite the inclusion of TF-IDF, the models still lack a deep understanding of tone, sarcasm, or relationships between words.
* Misclassifications: Reviews with mixed sentiments resulted in some false positives and false negatives.
* Although effective for binary classification, this might struggle with more complex tasks, such as multi-language datasets.

**Future Work:**

To improve and extend the project, the following can be done:

* Advanced Models: Explore deep learning architectures such as Recurrent Neural Networks (RNNs)
* Context and Sentiment: Addressing mixed sentiments and sarcasm by using deep learning designed for sequential data
* Dataset Expansion: Using datasets with more diverse sources (like Amazon reviews) to generalize the model