Project 1: Sentiment Analysis on Movie Reviews

Table of Contents

1. Introduction

Sentiment analysis is a type of natural language processing (NLP) that has great potential for understanding the sentiments and views that are hidden in textual data. Sentiment analysis has become an essential tool for understanding user ideas, feedback from customers, and public sentiment across multiple platforms because of the rapid growth of digital content. In this report we explore sentiment analysis in this research, with a particular emphasis on movie reviews.

This project aims to classify movie reviews into positive and negative sentiments by utilizing deep learning and machine learning techniques. Our goal in doing this is to provide filmmakers, critics, and other industry stakeholders a better understanding of how the public feels about movies, so they can evaluate the response from the public and make wise choices.

The initial step of our study is an in-depth analysis of the dataset, which consists of a wide range of movie reviews collected by multiple sources. We can learn more about the dataset's structure, potential patterns within the reviews, and the sentiment distribution by performing data exploration.

After doing data exploration, the raw text data is cleaned and made ready for modeling during the preprocessing phase. In order to prepare the data for machine learning and deep learning model training, this involves doing tasks like noise removal, lowercase conversion, and word tokenization.

Our project's core is the data modeling stage, when we create sentiment classification models using a variety of frameworks and methods. We analyze various methods for precisely modeling the complex sentiments expressed in movie reviews, including conventional machine learning

algorithms like Logistic Regression and Linear SVC to advanced deep learning models like Convolutional Neural Networks (CNNs) and fully-connected Neural Networks (MLP).

Lastly, we evaluate our models' performance using relevant metrics and methods to provide insight into their F1-score, accuracy, precision, and recall. Our goal is to determine the best method for sentiment analysis on movie reviews by comparing and examining the results.

With this project, we hope to show off the use of sentiment analysis in the context of movie reviews as well as the capabilities of deep learning and machine learning methods in analyzing human emotions and views as displayed in textual data. In the end, we hope to make a helpful contribution to the field of sentiment analysis by providing entertainment industry stakeholders with insightful knowledge collected from audience opinions.

2. Data Description and Exploration

Our goal in the "Data Description and Exploration" phase is to learn more about the features and structure of the movie review dataset. Here's a more detailed explanation of each:

- **Loading the Dataset:**
  - Using the relevant data loading function offered by the pandas library, we start by loading the dataset containing movie reviews. The dataset contained two primary columns: "review" and "sentiment," and it was saved in a structured format, typically in a spreadsheet or CSV file.

- **Fundamental Information about the Dataset:**
  - Using the `info()` method provided by pandas DataFrame, we analyzed the dataset's basic details. The dataset's structure, including the number of entries (rows) and the types of data in each column, was made clearer to us in this step.

- **Displaying the First few Rows:**
  - I used the `head()` method to show the first few rows of the dataset so that you could get an idea of the actual information.
  - This gave us the opportunity to analyze the structure, concepts, and opinions shared in the movie reviews.

- **Check Missing Values :**
  - Missing values may affect the quality of analysis and modeling and are often discovered in real-world datasets.
  - To find any null or missing values in the data, we used the {isnull().sum()} method to verify the dataset for missing values.
  - Making sure there are no missing values is important for maintaining the dataset's integrity and avoiding mistakes during analysis.

- **Exploring the Sentiments Distribution :**

- We visualized the distribution of sentiments in the dataset using a bar plot.
- This provided insights into the balance between positive and negative sentiments in the movie reviews.
- A balanced distribution of sentiments ensures that the dataset is representative and suitable for training machine learning models without bias towards any particular sentiment class.

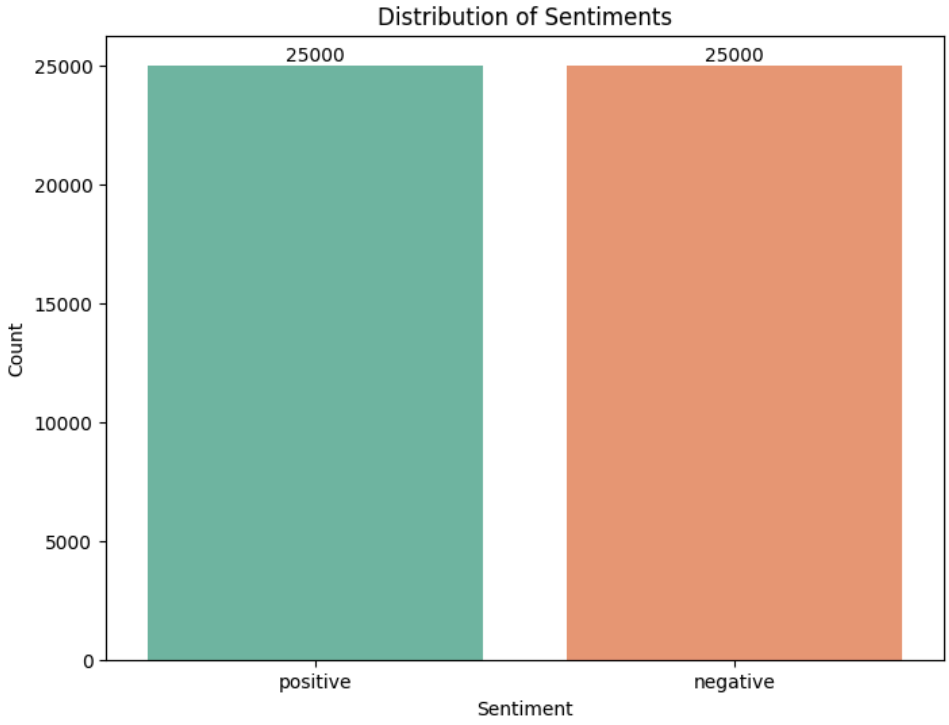

Fig 2.1 Distribution of Sentiments

Overall, the "Data Description and Exploration" phase prepared the basis for additional analysis and modeling by offering insight into the distribution, and structure of the dataset. Following preprocessing, modeling, and evaluation phases of the sentiment analysis project was determined by the findings of this exploratory analysis.

3. Data Preprocessing

During the data preprocessing stage, we performed multiple essential steps to get the unprocessed text data ready for modeling. An explanation of each step is provided below:

- **Removing Noise and Special Characters:**
  - Text data often contains noise and special characters that are unrelated to sentiment analysis and may affect model performance.

- To eliminate these undesired characters from the text, such as HTML tags, punctuation marks, and other non-alphanumeric symbols, we used regular expressions.
- Removing noise guarantees clean text data with only the necessary details required for analysis.

- **Transformation to Lowercase:**
  - When analyzing text data, inconsistencies could occur due to the combination of uppercase and lowercase letters.
  - We converted every word in the text to lowercase to maintain consistency and prevent processing the same word differently based on case variations. Tokenization and feature extraction, among other processing steps, work better and more effectively when the text data is preprocessed.

- **Word Tokenization:**
  - This method involves splitting the text into unique words, or tokens, which function as the main element of analysis.
  - To make the text data easier to process and analyze, we used word tokenization to divide it into meaningful tokens.
  - For tasks including natural language processing, tokenization is an essential preprocessing step since it separates the text into distinct parts that can be analyzed further.

- **Removal of Stopwords and Stemming (Not Used):**
  - Common words like "the," "is," and "and," that keep coming regularly in the text but usually have no semantic importance, are known as stopwords.
  - While stemming involves removing prefixes and suffixes to return words to their basic form. For instance, "running" turns into "run."
  - Despite being widely used preprocessing techniques, stopword removal and stemming were not used in this project. They may, however, be helpful in lowering noise and increasing the effectiveness of text processing algorithms.

- **Dataset Splitting:**
  - We divided the dataset into training and testing sets in order to evaluate how well machine learning and deep learning models performed.
  - 75% of the data was utilized as the training set, which was then used to train the models; the remaining 25% was used as the testing set, which was used to evaluate the performance of the models.
  - To make sure that the training and testing sets were equally representative of the whole data set and had a balanced distribution of both positive and negative reviews, random sampling was used.

The overall goal of the data preprocessing phase was to standardize, clean, and organize the unprocessed text data in order to make analysis and modeling more efficient. We prepared the

dataset for training deep learning and machine learning models for sentiment analysis on movie reviews by performing the preprocessing steps.

4. Data Modelling

In the data modeling phase, we applied various machine learning and deep learning techniques to classify movie reviews into positive or negative sentiments. Here's an elaboration on the models used:

- **Logistic Regression:**
  - Logistic Regression is a simple yet powerful linear classification algorithm widely used for binary classification tasks.
  - We utilized a logistic regression model with CountVectorizer for feature extraction, converting text data into numerical features.
  - The model achieved an accuracy of 89.16% on the test data, demonstrating its effectiveness in classifying movie reviews based on sentiment.
  - Logistic Regression provides interpretable coefficients, allowing us to understand the impact of each feature on the classification decision.

- **Linear SVC (Support Vector Classifier):**
  - Linear SVC is a linear classification model that aims to find the hyperplane that best separates the classes in the feature space.
  - Similar to logistic regression, we employed CountVectorizer for feature extraction and trained a Linear SVC model.
  - The model achieved an accuracy of 87.39% on the test data, indicating its capability in classifying movie reviews into positive or negative sentiments.
  - Linear SVC is known for its ability to handle high-dimensional data efficiently and is suitable for text classification tasks.

- **K Neighbors Classifier:**
  - The K Neighbors Classifier is a non-parametric classification algorithm that classifies samples based on the majority class among their k nearest neighbors.
  - We utilized CountVectorizer in combination with the K Neighbors Classifier to classify movie reviews.
  - However, the model achieved a relatively lower accuracy of 64.25%, suggesting that it may not be the best choice for sentiment analysis on movie reviews.
  - K Neighbors Classifier tends to be computationally expensive and may not perform well on high-dimensional data.

- **MLPClassifier (Fully-connected Neural Network):**
  - MLPClassifier is a multi-layer perceptron model, which is a type of fully-connected neural network.
  - We simplified the model architecture due to computational constraints, using a single hidden layer with 50 neurons and the ReLU activation function.

- The model achieved an accuracy of 84.13% on the test data, demonstrating competitive performance compared to traditional machine learning models.
- Fully-connected neural networks are capable of capturing complex patterns in data but may require extensive tuning of hyperparameters for optimal performance.

- **Convolutional Neural Network (CNN):**
  - CNNs are deep learning models commonly used for image classification but also applicable to sequential data such as text.
  - We implemented a CNN architecture for sentiment analysis, comprising embedding, convolutional, max-pooling, and fully-connected layers.
  - The CNN achieved an accuracy of 89.10%, comparable to logistic regression, indicating its effectiveness in extracting features from text data and capturing sentiment information.
  - CNNs are known for their ability to automatically learn hierarchical features, making them suitable for tasks involving structured data like text.

Overall, the data modeling phase involved experimenting with a variety of machine learning and deep learning models to identify the most effective approach for sentiment analysis on movie reviews. The results demonstrated the strengths and weaknesses of each model and provided insights into their performance characteristics and suitability for the task at hand.

5. Findings

The analysis of various sentiment classification models revealed interesting results.

Upon examining the model performance, the Convolutional Neural Network (CNN) appeared as the most accurate with an accuracy of 89.54%, closely followed by Logistic Regression, which attained an accuracy of 89.16%. Moreover, Linear SVC did well, with an accuracy of 87.34%. With an accuracy of 64.25%, the K Neighbors Classifier, on the other hand, lags behind, suggesting that it is relatively less good at categorizing movie reviews.

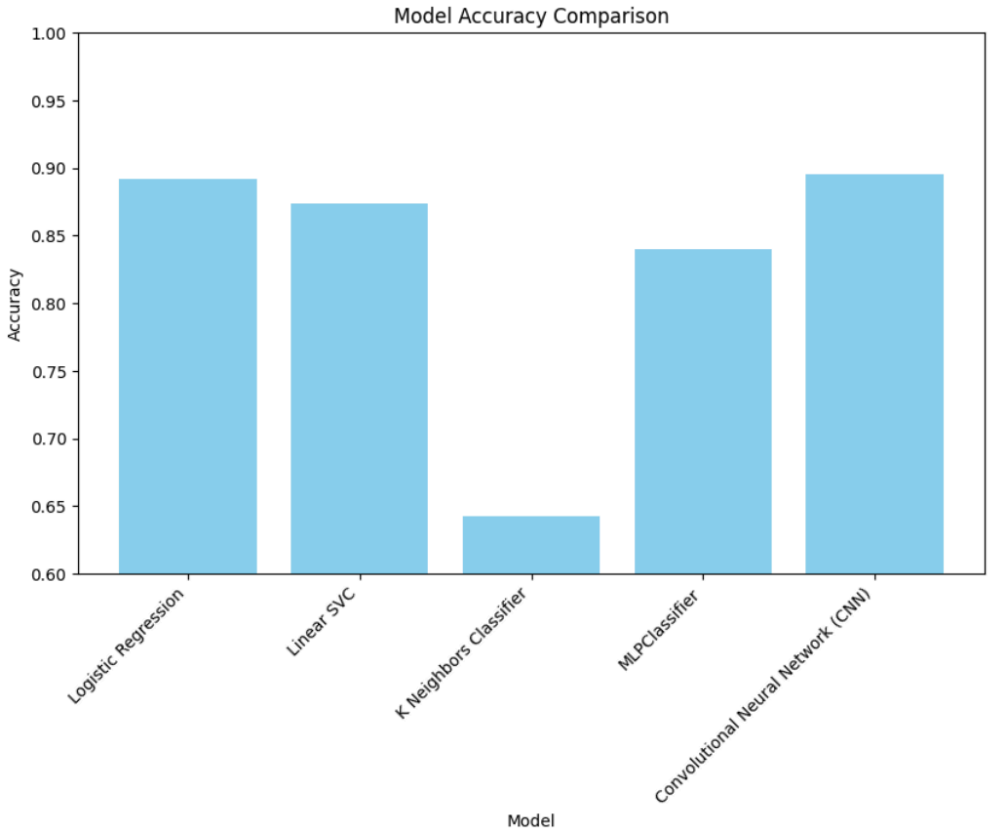

Fig 5.1 Model Accuracy Comparison

The confusion matrix for the Logistic Regression model provided a visual representation of how well the model predicted positive and negative sentiments. It showed the number of true positives, true negatives, false positives, and false negatives, offering insights into the model's performance.

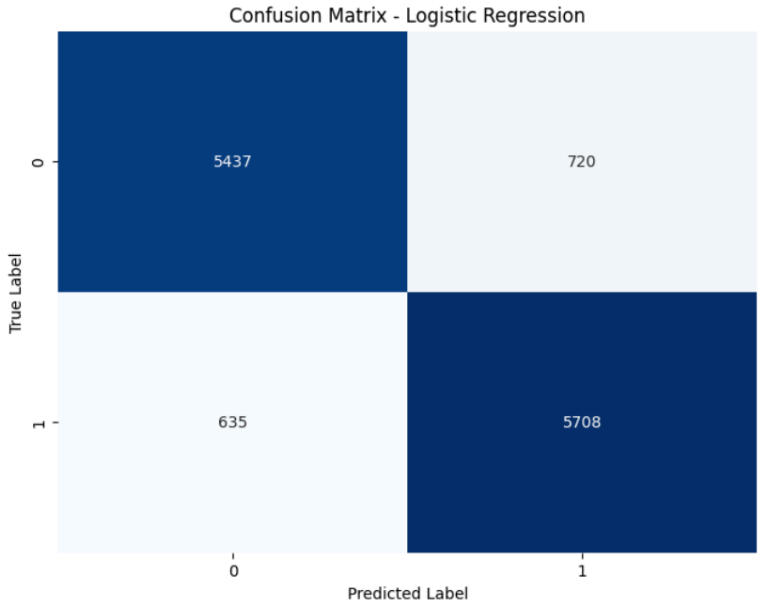

Fig 5.2 Confusion matrix Logistic Regression

Furthermore, the trade-off between true positive rate and false positive rate was demonstrated for all models using the Receiver Operating Characteristic (ROC) curves. The model's capacity to differentiate between positive and negative reviews was shown by the area under the ROC curve (AUC).

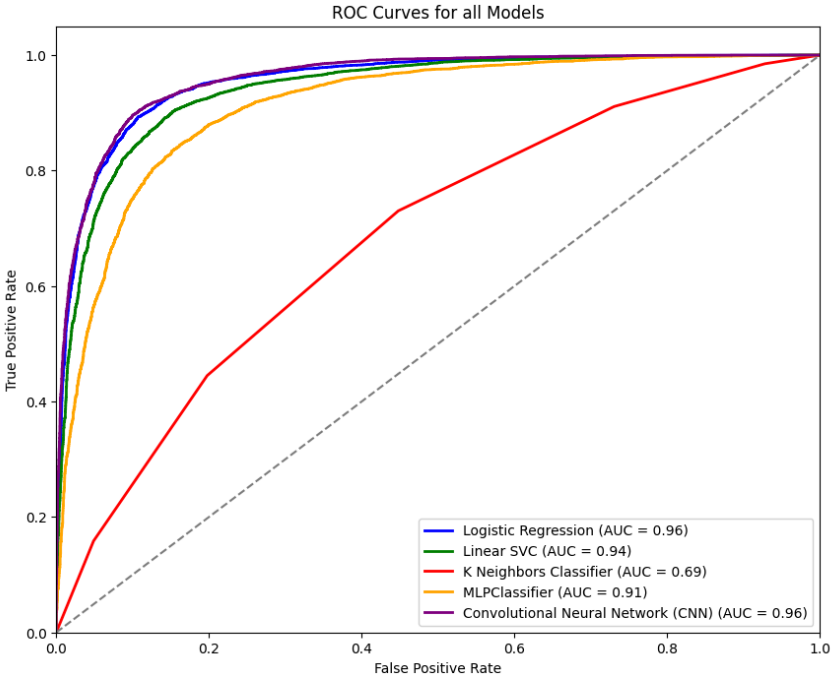

Fig 5.3 ROC curves for all models

Furthermore, each model's precision-recall balance was shown by the Precision-Recall curves. Better precision and recall performance was shown by a bigger area under the curve (AUC). Once more, the CNN model outperformed the other models, achieving the highest AUC of 0.96, closely followed by Logistic Regression at 0.95.

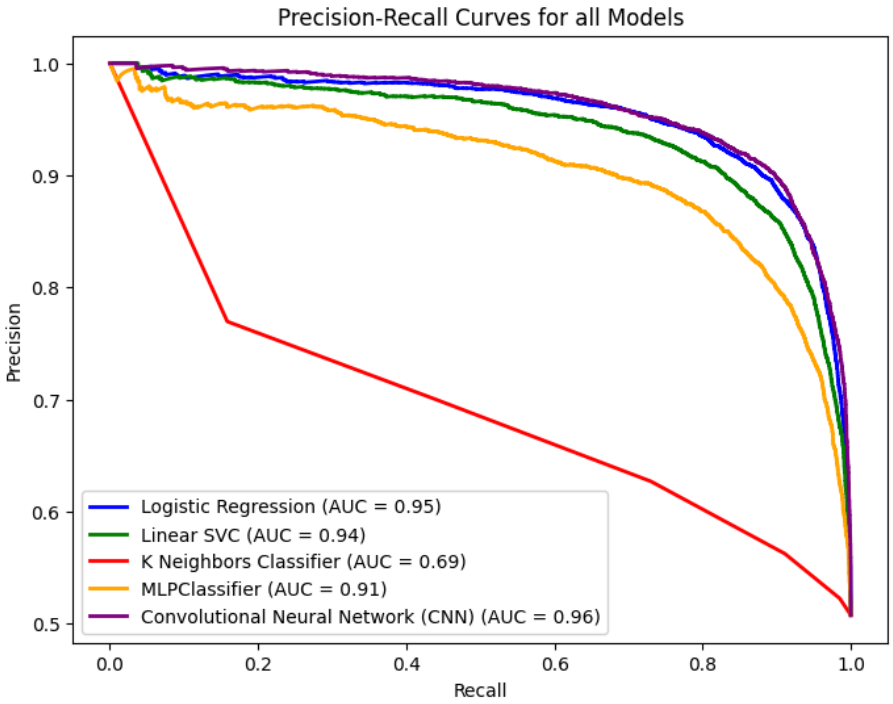

Fig 5.4 Precision Recall Curve for all models

These visualizations provide a comprehensive overview of the performance of each sentiment classification model, enabling stakeholders to make informed decisions based on the analysis of movie reviews.

6. Conclusion

We found a balanced sentiment distribution in the data, with approximately equal amounts of positive and negative movie reviews. This balanced distribution helps avoid biases towards any certain sentiment class, which is essential for efficiently training machine learning models.

We discovered that the text data had noise and special characters during the data exploration phase, which might have an impact on how well sentiment analysis models performed. Therefore, in order to make sure the data was consistent and clean for modeling, we performed preprocessing steps such as noise removal, text conversion to lowercase, and tokenization.

In addition, we divided the dataset into testing and training sets in order to evaluate how well different classification models performed. In order to prevent overfitting and evaluate how effectively the models generalize to new data, this phase is essential.

In terms of modeling, we experimented with several algorithms, including Logistic Regression, Linear SVC, K Neighbors Classifier, MLPClassifier (Fully-connected Neural Network), and Convolutional Neural Network (CNN). Each model demonstrated different levels of accuracy and performance in classifying movie reviews into positive or negative sentiments.

Overall, we found that Logistic Regression and CNN performed exceptionally well, achieving accuracies above 89%. These models showed they were really good at spotting patterns in the reviews and accurately guessing if the reviews were positive or negative. Additionally, we observed that the K Neighbors Classifier showed relatively lower accuracy compared to other models, suggesting that it might not be the most suitable choice for this task.

In summary, our analysis revealed that sentiment analysis on movie reviews can be effectively performed using machine learning and deep learning models, with Logistic Regression and CNN emerging as top-performing models in this context. These findings highlight the importance of data preprocessing, model selection, and evaluation in building accurate sentiment analysis systems for movie reviews.