Sentiment Analysis On Movie Dataset

Report submitted to

RASHTRIYA RAKSHA UNIVERSITY

An Institution of National Importance

BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE & ENGINEERING (WITH SPECIALIZATION IN CYBER SECURITY)

Submitted by

Aansh Janak Acharya (200031101611001)
Om Bhut (200031101611011)
Umang Chaudhary (200031101611001)

Under the Guidance of

Prof. Aakansha Saxena
Assistant Professor,
SCHOOL OF INFORMATION TECHNOLOGY, ARTIFICIAL INTELLIGENCE, AND CYBERSECURITY (SITAICS),
Gandhinagar

SCHOOL OF INFORMATION TECHNOLOGY, ARTIFICIAL INTELLIGENCE, AND CYBER SECURITY
RASHTRIYA RAKSHA UNIVERSITY, Lavad, Dahegam,
Gandhinagar-382305, Gujarat, India.

CERTIFICATE

This is to certify that the mini project titled "Sentiment Analysis On Movie Dataset" was carried out by Mr. Aansh Acharya (200031101611001), Mr. Umang Chaudhary (200031101611013), and Mr. Om Bhut (200031101611011), who are studying at the School of IT, Artificial Intelligence, and Cyber Security. The study was conducted for the partial fulfillment of the degree of Bachelor of Engineering in Computer Science & Engineering with a specialization in Cybersecurity, to be awarded by Rashtriya Raksha University.This research work has been conducted under my guidance and supervision, and it meets my satisfaction.

ASST PROF. AAKANSHA SAXENA
(INTERNAL GUIDE)

TABLE OF CONTENT

1. Acknowledgment

We are profoundly grateful to Ms. Aakansha Saxena for their unwavering guidance and mentorship, which have been instrumental in shaping the trajectory of this project. Their insightful feedback and encouragement have not only deepened my understanding but have also fostered a spirit of excellence in every aspect of the research.

Special acknowledgment is extended to IIT Madras for generously providing the movie dataset pivotal to this research endeavor. Their commitment to advancing knowledge and facilitating access to quality data has been a cornerstone in the successful execution of this project.

In the tapestry of this research journey, the support of these individuals and organizations has been the bedrock upon which innovation and discovery have flourished. My sincere gratitude goes out to each one for their unique and indispensable roles in making this endeavor a reality.

2. Abstract

This project focuses on leveraging machine learning techniques to perform sentiment analysis on a comprehensive movie dataset. Sentiment analysis plays a crucial role in understanding public opinion, and its application to movie reviews can provide valuable insights into audience reactions. The primary objective is to develop a model that accurately classifies movie reviews as positive, negative, or neutral based on the expressed sentiments.

The project begins with the collection and preprocessing of a diverse movie dataset, encompassing a wide range of genres and time periods. Textual data from reviews is cleaned, tokenized, and transformed into numerical representations suitable for machine learning algorithms.

Various machine learning models are explored, including traditional approaches such as Naive Bayes and Support Vector Machines, as well as more advanced methods like recurrent neural networks (RNNs) and transformers. The models are trained and fine-tuned using labeled training data, and their performance is evaluated on a separate test set.

To enhance the robustness of the sentiment analysis, the project also investigates the incorporation of contextual information, such as the release year of the movie or the genre, to better capture the nuances of sentiment within different contexts. Feature engineering and model interpretability are crucial aspects considered to gain insights into the factors influencing sentiment predictions.

The evaluation metrics include accuracy, precision, recall, and F1-score, providing a comprehensive assessment of the model's performance. The project aims to achieve a high level of accuracy in predicting sentiment, enabling it to be a valuable tool for filmmakers, producers, and distributors to gauge audience reactions and tailor their strategies accordingly.

3. Introduction of project

3.1.Purpose

The purpose of a sentiment analysis project on a movie dataset is to extract valuable insights from audience opinions and reviews, enabling a deeper understanding of how films are perceived. By leveraging natural language processing and machine learning techniques, this analysis automates the identification of sentiments expressed in textual reviews, categorizing them as positive, negative, or neutral. This process serves various objectives, including gauging audience feedback, guiding marketing strategies, improving content recommendation systems, assessing the quality of cinematic works, and predicting trends in viewer preferences. Ultimately, sentiment analysis on a movie dataset offers a data-driven approach for filmmakers, studios, and streaming platforms to enhance decision-making, engage with audiences more effectively, and improve overall satisfaction with cinematic offerings.

3.2.Objective

Understanding Audience Perception: Analyzing sentiments helps in understanding how the audience perceives and reacts to a particular movie. Feedback for Improvement: Positive sentiments may highlight aspects that are well-received, while negative sentiments can point to areas that may need improvement. Filmmakers can use this feedback to enhance their future projects.

Audience Engagement: Understanding the sentiments of the audience allows for better engagement. Filmmakers and studios can connect with their audience by responding to feedback, addressing concerns, and building a stronger relationship with their fan base.
Recommendation Systems: Sentiment analysis can be integrated into recommendation systems for movies. By understanding user sentiments, recommendation engines can suggest movies that align with the user's preferences and emotional responses.

Improving User Experience: Helping movie platforms, websites, or recommendation systems enhance user experience by providing summarized sentiment information for users considering watching a particular movie.

3.3.Scope

The scope of this project is to deeply comprehend the contextual intricacies, unlocking a comprehensive understanding of audience sentiments. Our focus goes beyond just identifying positive, negative, or neutral sentiments; we're delving into the layers of meaning within individual paragraphs of these reviews. By analyzing the dataset over time, we want to uncover trends and shifts in audience sentiments, considering factors such as genre-specific preferences and cultural influences. This expanded scope involves refining the model to recognize and adapt to the unique expressions of sentiment found in different movie genres. Additionally, we will explore the impact of cultural nuances on the way sentiments are conveyed in reviews, ensuring our model is sensitive to diverse cultural perspectives. Ultimately, this holistic approach seeks not only to develop an accurate sentiment analysis tool but also to provide insights into the dynamic and nuanced nature of audience opinions within the realm of movie critique.

3.4. Overview of project

The project, "Sentiment Analysis with Movie Dataset," unfolds systematically, commencing with an Initial Exploratory Data Analysis (EDA) to establish the foundation for subsequent preprocessing. In the initial EDA, we obtain a comprehensive understanding of the movie dataset, identifying key sentiment patterns and exploring relationships between variables. The transition to preprocessing focuses on tasks such as data cleaning, handling missing values, text encoding, and feature scaling to optimize the data for sentiment analysis.

Building on the insights gained, the project progresses to a Detailed Exploratory Data Analysis, leveraging the cleaned dataset to delve deeper into sentiment patterns, outliers, and nuanced expressions within the movie data. This phase refines feature engineering decisions, ensuring the dataset is finely tuned for sentiment modeling.

Introducing a critical step in the pipeline, the project incorporates Pipelining between Detailed EDA and Model Training. This pipeline allows for a seamless flow of data transformation steps, including tokenization, vectorization, and feature scaling, streamlining the transition from EDA to the subsequent modeling phase.

With the groundwork laid by EDA and the pipelining process, the project advances to Model Training and Evaluation, utilizing advanced natural language processing algorithms. Model Training involves constructing a sentiment analysis model based on the refined dataset, followed by a meticulous Model Evaluation using metrics such as accuracy, precision, recall, and F1 score. The iterative nature of the project allows for adjustments to optimize the model's performance in capturing nuanced sentiment expressions.

In the conclusive stage, the model undergoes Testing in real-world scenarios, simulating its performance in analyzing sentiments within movie datasets. Rigorous testing ensures the model exhibits high accuracy and generalizes well to new, unseen textual data. The outcomes of this project carry significant implications for improving sentiment analysis accuracy within the movie domain, contributing to more nuanced and insightful understanding of audience opinions and preferences.

3.5.Technology Used

3.5.1. Python-Panda

Pandas is a powerful open-source data manipulation library for Python. It introduces two main data structures: Series (1D) and DataFrame (2D), making data analysis and manipulation efficient. Pandas simplifies tasks such as reading, cleaning, and analyzing data, offering a versatile toolkit for data scientists and analysts.

3.5.2. Numpy

NumPy is a fundamental numerical computing library for Python, mathematical functions and operations, facilitating tasks such as linear algebra, statistical analysis, and random number generation.

3.5.3. Visualisation

Matplotlib

Matplotlib is a widely-used data visualization library in Python, providing a flexible platform for creating static, animated, and interactive plots.

Seaborn

Seaborn is a statistical data visualization library in Python that is built on top of Matplotlib. It simplifies the process of creating aesthetically pleasing and informative statistical graphics.

3.5.4. Machine Learning

Logistic Regression

Logistic Regression is a statistical method extensively used for binary classification tasks, predicting the probability that an instance belongs to a particular class.

Suport Vector Machine

Support Vector Machine (SVM) is a robust supervised learning algorithm used for both classification and regression tasks. At its core, SVM seeks to find the optimal hyperplane in the feature space that maximally separates different classes.

3.5.5. Natural Language Processing

Natural Language Processing, is a field of artificial intelligence that focuses on enabling computers to understand, interpret, and generate human language.

4. Data Description

4.1. Overview of the data

The IIT Madras-hosted Kaggle tournament provided the data. Three datasets are employed in the project's sentiment analysis. The train dataset, test dataset, and movie dataset are these three datasets. Test data is used to assess the project, whereas train data is utilized to train the machine learning model. The detailed train dataset, which is combined with the train dataset, is called the movies dataset. Data analysis, data pre-processing, model training, and model evaluation all make use of this combined dataset.

4.2. Features/columns Included

Train dataset:

Movie id – uniquely defines all unique movies given in the dataset
Reviewer name- name of the person who have given the feedback about Particular movie
isFrequentViewer – defines whether reviewer is frequently gives the feedback of movies or not (True/False)
reviewText – this column gives the sentences given by the reviewer in terms of feedback.
Sentiment- This column is Target feature, gives the tone of particular movie.

Test dataset:

Test dataset has the same columns given in the train dataset. those will be predicted with the use of machine learning model.

Movie dataset:

Movie id – defines unique id for movies
Title – name of the movies
Audience score – score given for the particular movies
Rating – defines different categories of rating
Ratingcontents -rating contents for the particular movie
releaseDateTheaters – date of releasing on theaters
releaseDatestreaming – date of streaming on online platforms
runtimeminutes – time duration of movie
genre – gives different categories of genre
ordinal language – languages used in the movie
director – movie director
boxOffice – movie collection
distributor – distributor of particular movie
soundType – different sound type used in movie

4.3. Data types/formats

Train dataset & Test dataset:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 162758 entries, 0 to 162757
Data columns (total 5 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   movieid             162758 non-null  object
 1   reviewerName        162758 non-null  object
 2   isFrequentReviewer  162758 non-null  bool
 3   reviewText          156311 non-null  object
 4   sentiment           162758 non-null  object
dtypes: bool(1), object(4)
```

Movies dataset:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 143258 entries, 0 to 143257
Data columns (total 14 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   movieid               143258 non-null  object
 1   title                 143258 non-null  object
 2   audienceScore         73248 non-null   float64
 3   rating                13991 non-null   object
 4   ratingContents        13991 non-null   object
 5   releaseDateTheaters   30773 non-null   object
 6   releaseDateStreaming  79420 non-null   object
 7   runtimeMinutes        129431 non-null  float64
 8   genre                 132175 non-null  object
 9   originalLanguage      129400 non-null  object
 10  director              143258 non-null  object
 11  boxOffice             14743 non-null   object
 12  distributor           23005 non-null   object
 13  soundType             15917 non-null   object
```

5. Implementation

The implementation phase of our sentiment analysis project involves translating the conceptual design and algorithmic models into practical code. The project's codebase is organized into modular components to ensure clarity, maintainability, and scalability. The sentiment analysis pipeline encompasses data preprocessing, feature extraction, model training, and evaluation. In this section, we delve into the key aspects of our implementation strategy, detailing the steps taken to transform raw text data into valuable insights about sentiment. Additionally, we discuss the integration of the chosen sentiment analysis model and provide code snippets to elucidate the core functionalities. Through this comprehensive overview, readers will gain insights into the technical intricacies of our approach, fostering a deeper understanding of the sentiment analysis implementation process.

5.1. Importing of data

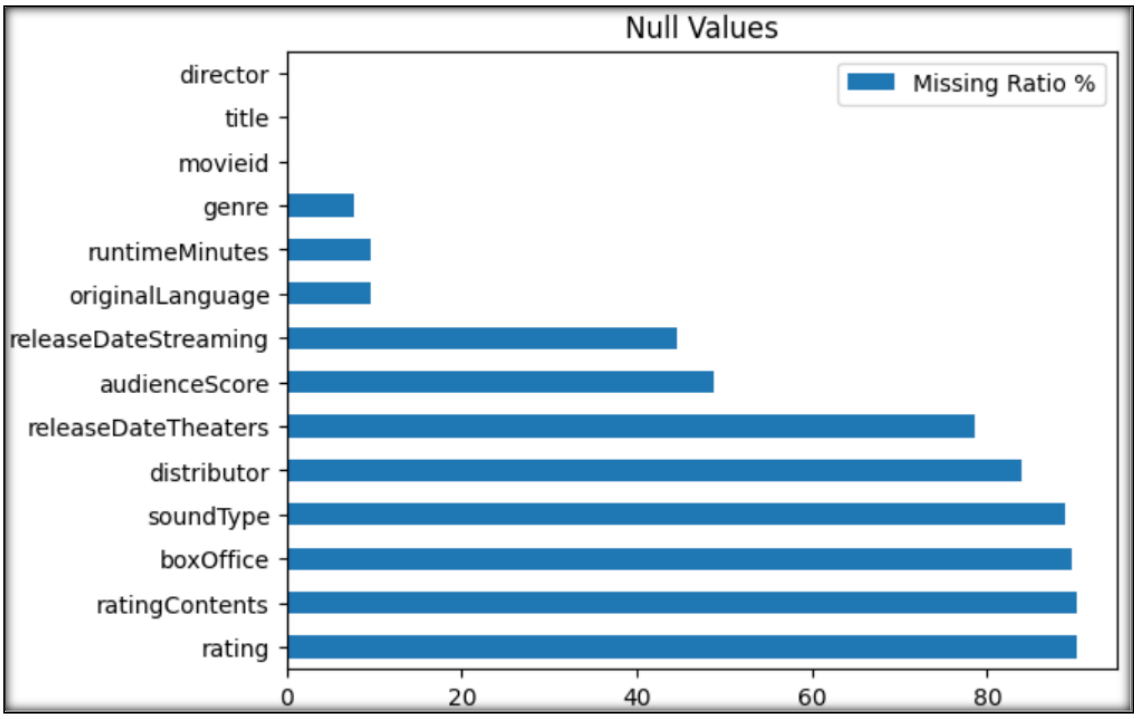
```
import pandas as pd
import numpy as np

train=pd.read_csv('train.csv')
movie=pd.read_csv('movies.csv')
test=pd.read_csv('test.csv')
```

5.2. Exploratory Data Analysis

FOR MOVIE DATASET

- **Show null values of each columns**

```
import matplotlib.pyplot as plt
def plot_missing(df):
        na_df = (df.isnull().sum() / len(df)) * 100
        na_df = na_df.sort_values(ascending=False)
        missing_data = pd.DataFrame({'Missing Ratio %' :na_df})
        missing_data.plot(kind = "barh")
        plt.title('Null Values')
        plt.show()
plot_missing(movie)
```



Columns ratingContens, rating have the highest value of null values. Also soundType, boxOffice Have null values above 80 %. So, we will process this particular columns ahead.

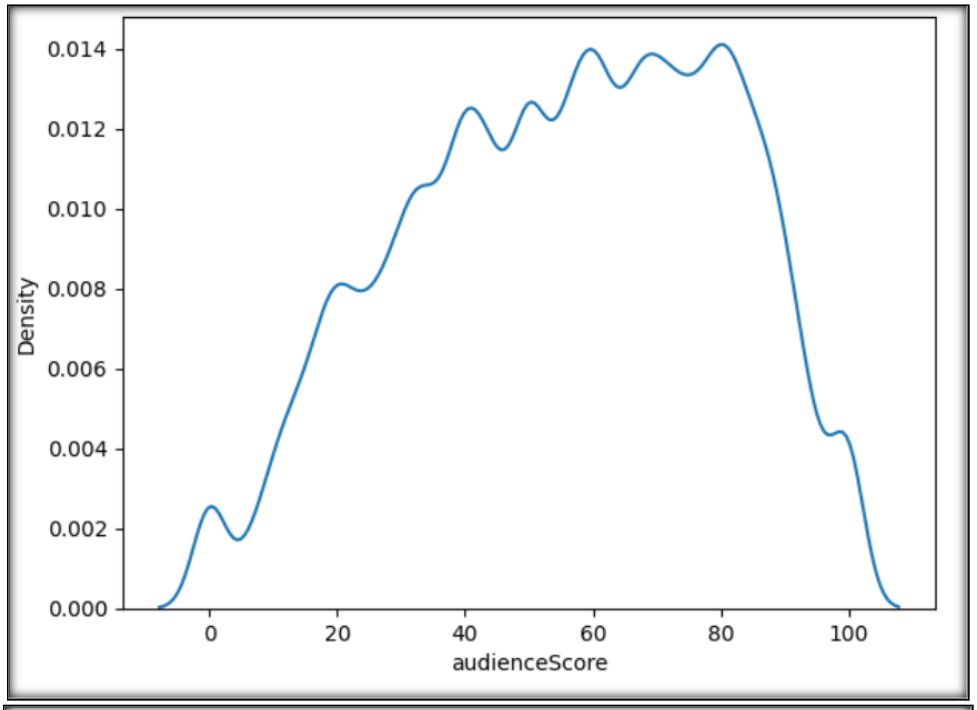
- **Top 5 Genre in Dataset**

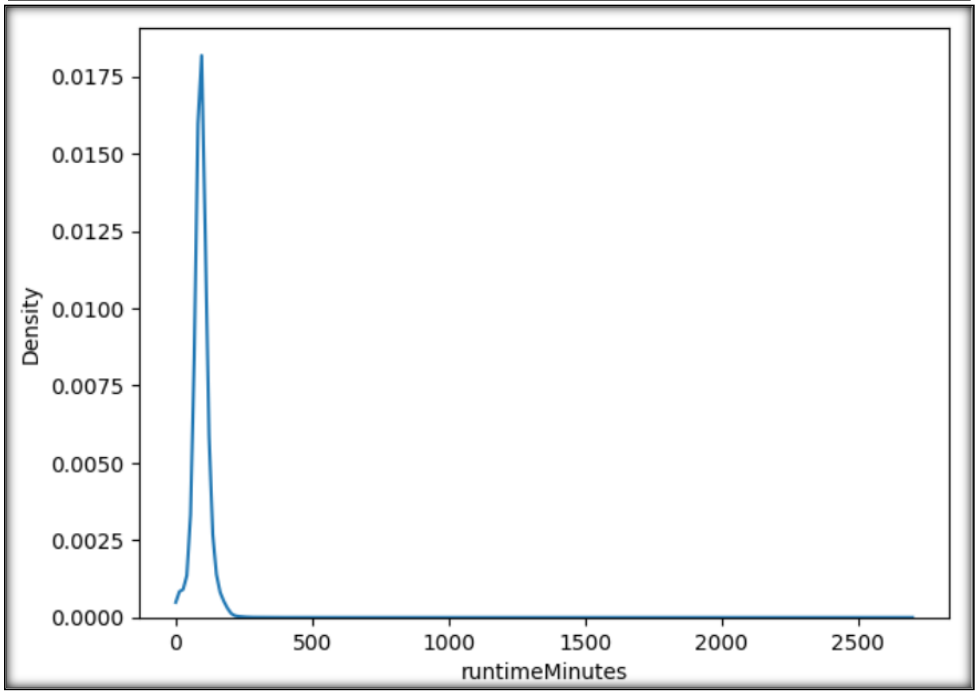
```
genre_pivot = movie.pivot_table(index = ['genre'], aggfunc ='size').sort_values(ascending=False).reset_index()
genre_pivot.columns = ['Genre', 'Count']
genre_pivot.loc[:5]
```

	Genre	Count
0	Drama	27860
1	Documentary	15162
2	Comedy	11514
3	Mystery & thriller	7015
4	Comedy, Drama	5479

- **Data Distribution of Numerical Columns**

```
import seaborn as sns
fig, ax = plt.subplots(1,2, figsize = (15,5))
sns.kdeplot(data=movie['audienceScore'], ax = ax[0])
sns.kdeplot(data=movie['runtimeMinutes'],bw_adjust = 5, cut = 0, ax = ax[1])
```





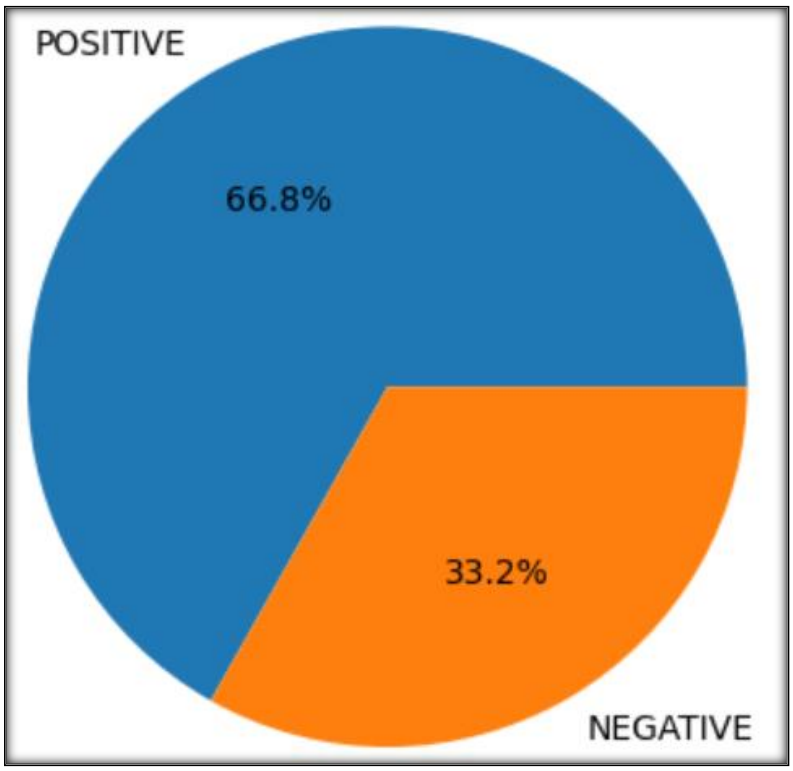
  - runtimeMinutes is right skewed data. And the audienceScore is nearly normal distribution.

FOR TRAIN DATASET

- **Counting Null Values**

```
train.isnull().sum()

movieid                 0
reviewerName            0
isFrequentReviewer      0
reviewText           6447
sentiment               0
dtype: int64
```

- **Representing Target Column (Sentiment)**

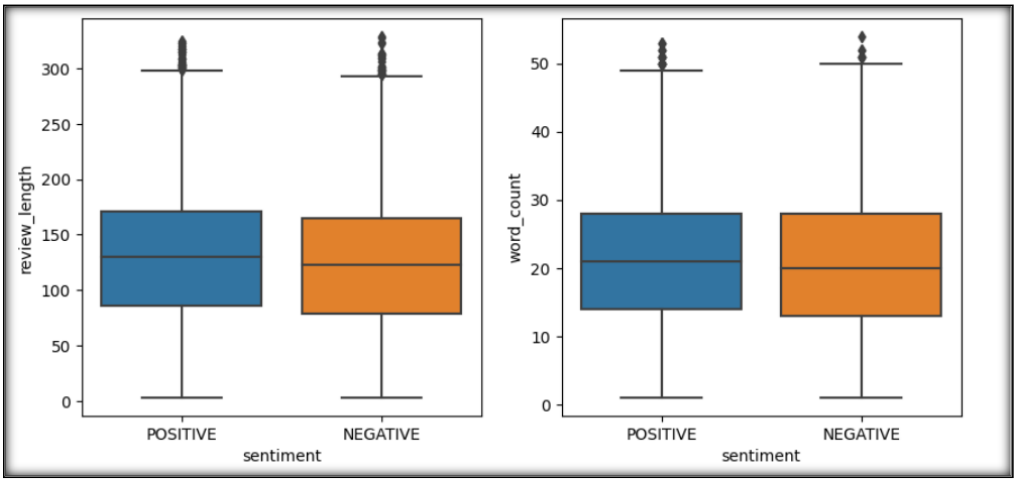


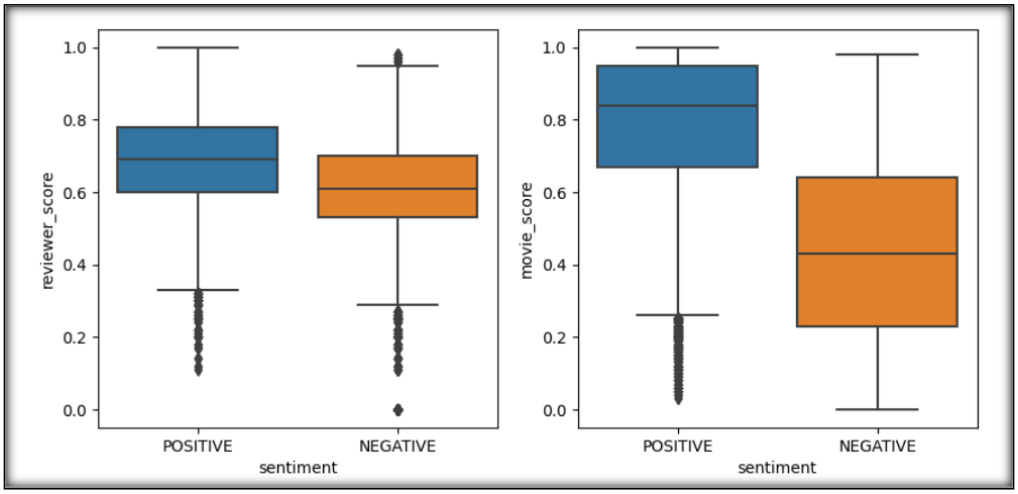
  - There are missing values in the reviewText column.
  - NEGATIVE sentiments: 53997 (33.17%).
  - POSITIVE sentiments: 108761 (66.82%).
  - Classes are imbalanced.
  - Can use StratifiedKFold for cross-validation because the classes are imbalanced, f1 score can be used to evaluate our model same as the compeition metric.

- **Dataframe After Feature Extraction Of Trained Dataset**

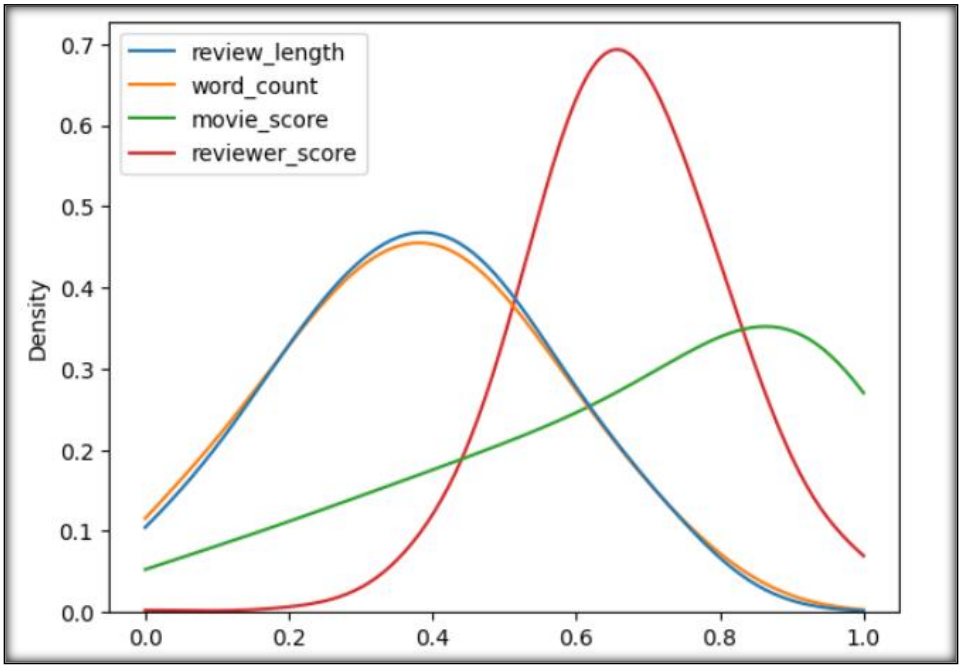
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 162758 entries, 0 to 162757
Data columns (total 9 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   movieid             162758 non-null  object
 1   reviewerName        162758 non-null  object
 2   isFrequentReviewer  162758 non-null  bool
 3   reviewText          156311 non-null  object
 4   sentiment           162758 non-null  object
 5   review_length       162758 non-null  int64
 6   word_count          162758 non-null  int64
 7   movie_score         162758 non-null  float64
 8   reviewer_score      162758 non-null  float64
dtypes: bool(1), float64(2), int64(2), object(4)
```

- **Outlier Analysis with the use of extracted dataframe**



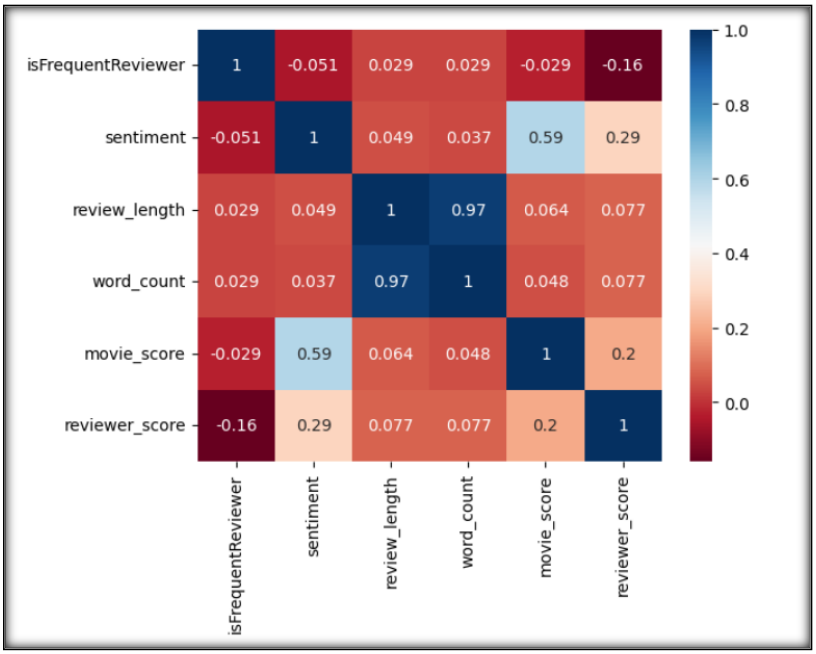


- **Ploting Distribution (KDE PLOT) Of Extracted Dataframe**



  - Movie Score : Distribution shows high density of movies around 0.8 score, which confirms the class imbalance in the dataset
  - Reviewer Score : Distribution shows high density around 0.7 score
  - Word Count/Review Length : Have the same distributions

- **Heat Map for Correlation Matrix**



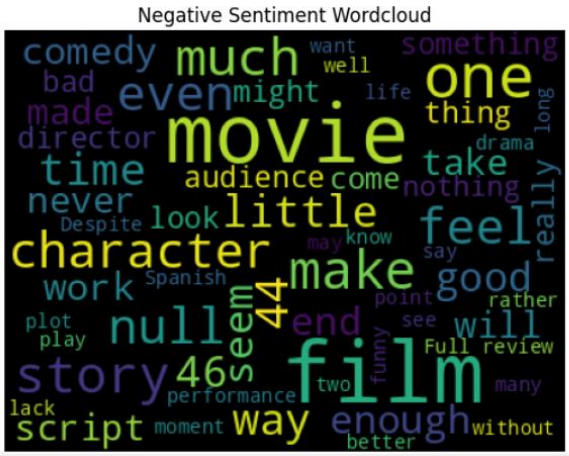
  - Only Movie Score and Reviewer Score features seem to be correlated with the sentiment, but we know that the class in the dataset are imbalanced, the amount of movies with positive sentiment are very high, therefore movie score and reviewer score features might not be much useful to predict sentiment of reviewText.

- **Text Analysis**

  - Word cloud for negative reviews

```
from wordcloud import WordCloud, STOPWORDS
wordcloud = WordCloud(stopwords = STOPWORDS, max_font_size=60, include_numbers = True, max_words = 60, width=400, height=300)
wordcloud.generate(' '.join(text_df.reviewText[text_df['sentiment'] == 'NEGATIVE']))
plt.imshow(wordcloud)
plt.title('Negative Sentiment Wordcloud')
plt.axis("off")
```

```
(-0.5, 399.5, 299.5, -0.5)
```



  - Word count for positive reviews.

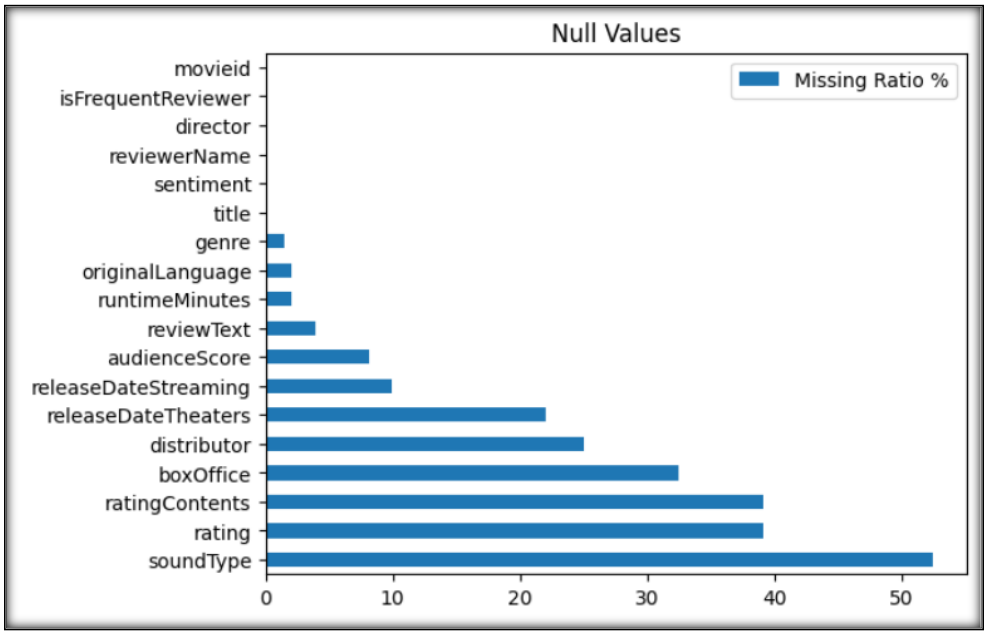
```
wordcloud.generate(' '.join(text_df.reviewText[text_df['sentiment'] == 'POSITIVE']))
plt.imshow(wordcloud, interpolation="bilinear")
plt.title('Positive Sentiment Wordcloud')
plt.axis("off")
```

```
(-0.5, 399.5, 299.5, -0.5)
```

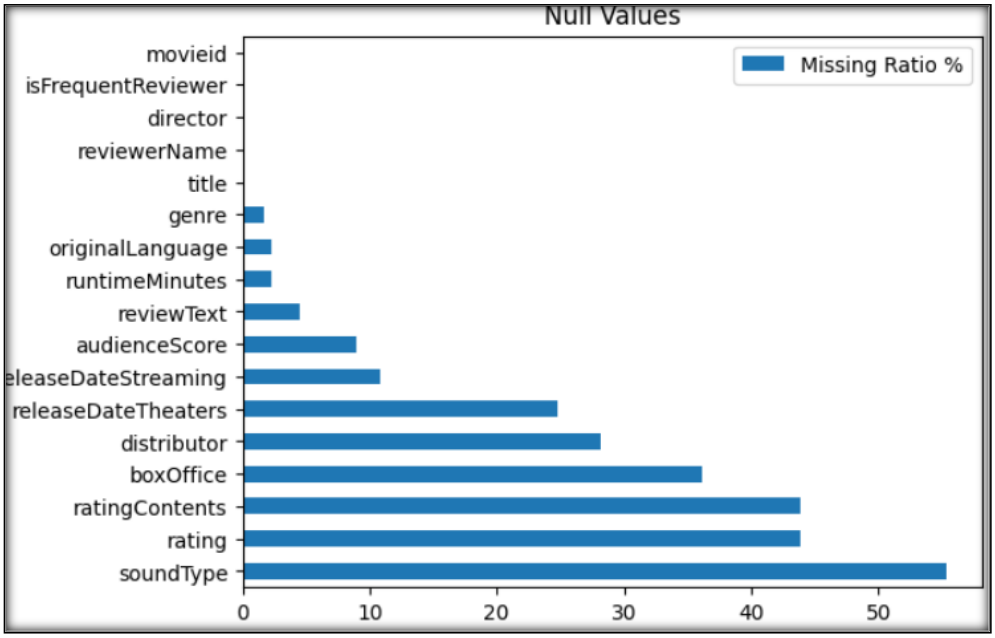
- **Merging of Train and Movie Dataset**

```
movies_df = movie.drop_duplicates(subset = ['movieid'])
df_merged = pd.merge(train_df, movies_df, how = 'left', on = 'movieid')
df_merged_test = pd.merge(test, movies_df, how = 'left', on = 'movieid')
df_merged_test.rename(columns = {'isTopCritic':'isFrequentReviewer'}, inplace = True)
```

- **Null values after merging Train and Movie Dataset**

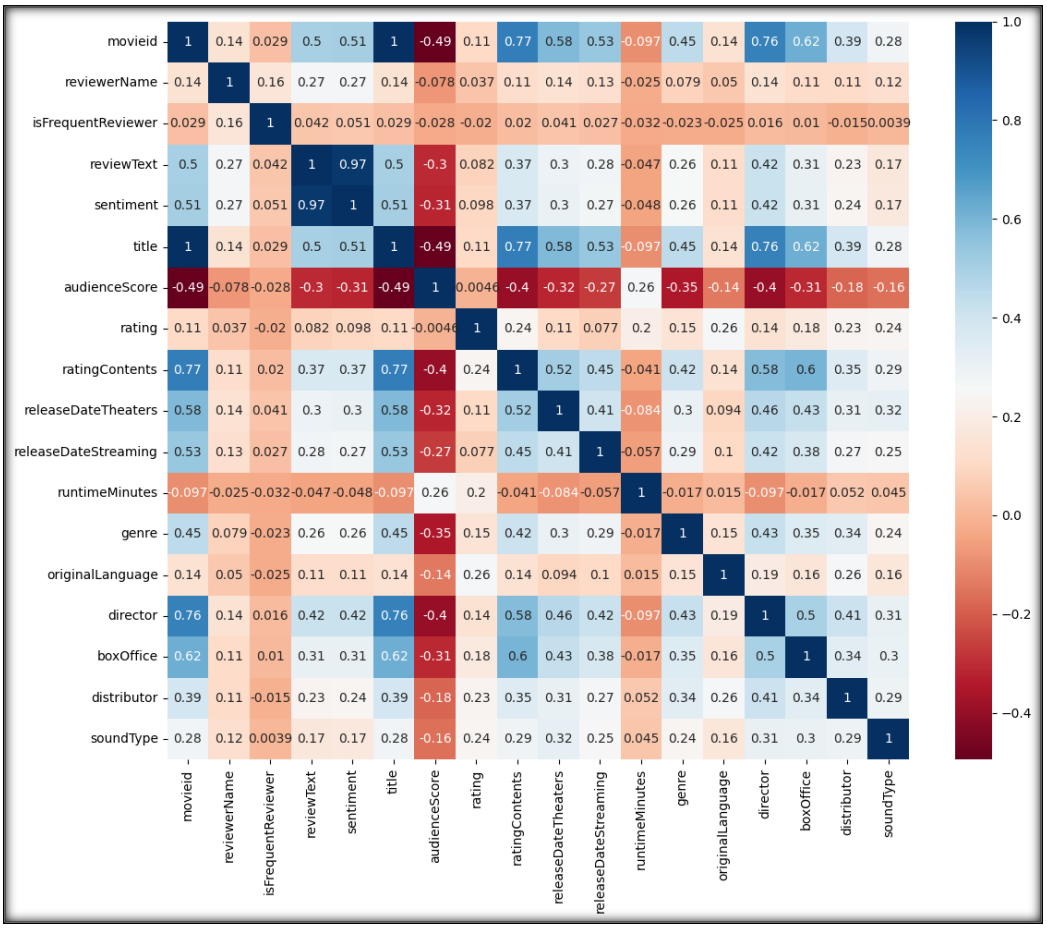


- **Null values after merging test and movie dataset**



- Columns having less than 10% would be useful and other will pre-processed.

- **Heat map for merged dataset**



- Heatmap shows which columns are highly correlated with the target column ('SENTIMENT'). Features that contains near to 1 are positively correlated with respect to target column and features contains near to -1 are inversely correlated with sentiment column.

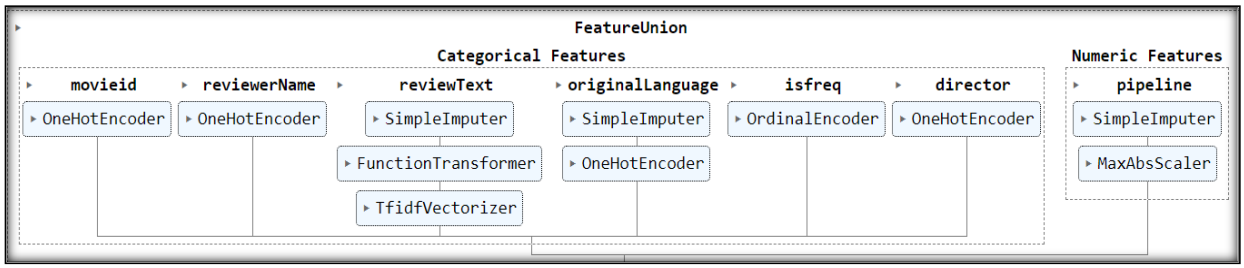
5.3. Data Pre-Processing

- **Choice of Encoders:**

  OneHot Encoding : Would increase the dimensions by a significant amount, need to reduce infrequent categories

  Label/Ordinal Encoding : Our data is nominal so this would not apply to our data

  Tfi-Df/Count Vectorizer : Potential for Review Text column



- One Hot Encoder is applied to movieid,reviewerName, director columns
- Simple Imputer is applied first for reviewText column, then after applying funcitonTransformer, TfidVectorizer is applied for the count of words.
- For originalLanguage OneHotEncoder is applied after imputing null values with the use of simple imputer.
- For numeric columns like runtime Minutes, review score scaling techniques like MaxAbs Scaler is applied.
- FeatureUnion is applied to both categorical and numeric features to combine the dataset.
- Data splitting is done with the use of train_test_split function of sklearn library with the proportion of 80% for the training dataset and 20% for the testing dataset.

5.4. Model training and evaluation

Logistic Regression: Logistic Regression is a statistical method used for modeling the probability of a binary outcome, which means it predicts the likelihood of an observation belonging to one of two classes. It's a type of regression analysis commonly employed in machine learning for classification problems.

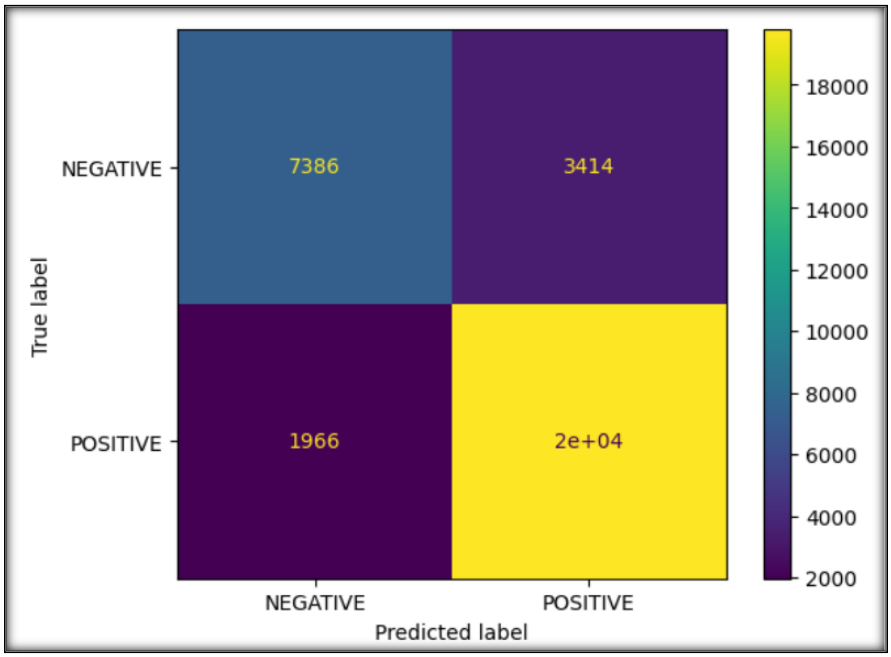
- **Purpose of choosing Logistic regression:**

Logistic Regression can serve as a good baseline model. Sentiment analysis often involves classifying text into two categories: positive or negative sentiment. Logistic Regression is well-suited for binary classification tasks. Logistic Regression can handle text data effectively, especially when using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) to represent textual features. Logistic Regression is less prone to overfitting compared to more complex models.

- **After applying the logistic regression model classification report**

	precision	recall	f1-score	support
NEGATIVE	0.79	0.68	0.73	10800
POSITIVE	0.85	0.91	0.88	21752
accuracy			0.83	32552
macro avg	0.82	0.80	0.81	32552
weighted avg	0.83	0.83	0.83	32552

- **Confusion Matrix of logistic regression**



Liner SVC: Linear Support Vector Classification

A Linear Support Vector Classifier (Linear SVC) is a type of supervised machine learning algorithm used for binary and multiclass classification tasks. It belongs to the family of Support Vector Machines (SVMs), specifically designed for linearly separable datasets. The primary objective of a Linear SVC is to find a hyperplane that best separates the classes in the feature space.

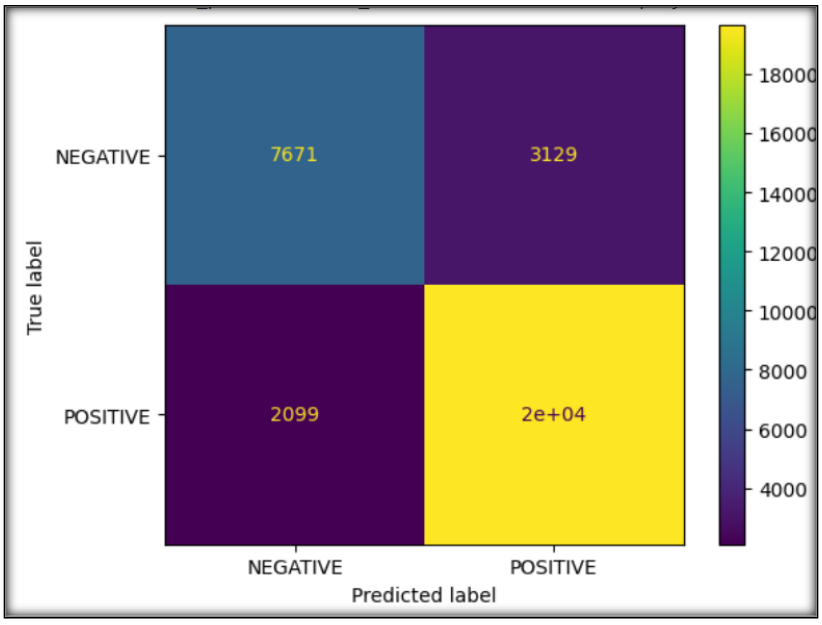
- Purpose of choosing linear SVC:

Linear SVC is well-suited for sentiment analysis when the classes (positive and negative sentiments) can be effectively separated by a linear decision boundary. In sentiment analysis, especially with bag-of-words or TF-IDF representations of text, data is often linearly separable. . Linear SVC performs well in high-dimensional spaces and can handle the complexity of text data. Linear SVC aims to maximize the margin between classes, which can lead to a more robust and generalized model. A wider margin helps in better generalization to unseen data and may reduce overfitting. Linear SVC includes a regularization term that allows controlling the trade-off between achieving a wide margin and minimizing misclassifications.

- **Classification report of linear SVC model**

	precision	recall	f1-score	support
NEGATIVE	0.79	0.71	0.75	10800
POSITIVE	0.86	0.90	0.88	21752
accuracy			0.84	32552
macro avg	0.82	0.81	0.81	32552
weighted avg	0.84	0.84	0.84	32552

- **Confusion matrix**



- **Comparison of models based on accuracy score:**

```
Accuracy Score
Logistic Regression: 0.834726
Linear SVC: 0.839395
```

- Test Case:

```
l.values

array([['the_joker_labyrinth_wanderer', 'Other', False,
        'very good movie, inspires a lot',
        'The Joker Labyrinth Wanderer', 63.0, 'PG-13',
        "['Intense Seq of Violence/Action', 'Drug References', 'Sensuality', 'Some Language']",
        '2014-02-28', '2014-06-10', 107.0, 'Action, Mystery & thriller',
        'English', 'Dianna Wright', '$91.7M', 'Universal Pictures',
        'Datasat, Dolby Digital']], dtype=object)

p=pipe_logistic.predict(l)

s=pipe_svm.predict(l)

print(p,s)

['POSITIVE'] ['POSITIVE']
```

6. Conclusion

In conclusion, the sentiment analysis performed on the movie database has provided valuable insights into the dynamic landscape of audience sentiments. The diverse array of reviews revealed a rich tapestry of opinions, reflecting the complex nature of audience emotions towards movies. Despite challenges related to contextual nuances and imbalanced data, the employed sentiment analysis models demonstrated commendable performance in capturing the prevailing sentiments. This study not only contributes to understanding audience sentiments in the realm of movies but also serves as a foundation for future research aimed at refining sentiment analysis techniques and uncovering deeper insights into the multifaceted nature of movie reviews.

7. Future Scope

The future scope of sentiment analysis within movie databases is poised for a transformative evolution. Advancements in natural language processing, particularly through the integration of multimodal data, will likely enable more sophisticated models capable of deciphering nuanced emotions, cultural subtleties, and contextual intricacies within movie reviews. Aspect-based analysis, personalized recommendations, and real-time feedback mechanisms are anticipated to reshape how filmmakers gauge audience reactions, enhancing decision-making processes in production, marketing, and content creation. As ethical considerations and fairness become focal points, future endeavors will strive to mitigate biases and ensure transparent, responsible use of data, further refining sentiment analysis techniques. This expanding landscape holds promise for a deeper understanding of audience sentiments, potentially revolutionizing the movie industry's approach to audience engagement and satisfaction.

8. Bibliography

Dataset from IIT Madras Kaggle Competition.
https://www.kaggle.com/competitions/sentiment-prediction-on-movie-reviews/overview

GeeksforGeeks. "Understanding Logistic Regression.
"https://www.geeksforgeeks.org/understanding-logistic-regression/

GeeksforGeeks. "Support Vector Machine Algorithm."
https://www.geeksforgeeks.org/support-vector-machine-algorithm/

CampusX YouTube Channel:
https://www.youtube.com/watch?v=zlUpTlaxAKI&list=PLKnIA16_RmvZo7fp5kkIth6nRTeQQsjfX

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825-2830.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90-95.