PREDICTING IMDb SCORES

PHASE 3 – DEVELOPMENT PART 1

Introduction

Predicting IMDb scores is a fascinating task that lies at the intersection of data science, machine learning, and the entertainment industry. IMDb, or the Internet Movie Database, is one of the most popular platforms for users to rate and review movies and television shows. Predicting IMDb scores involves developing models that can estimate the average user rating (on a scale of 1 to 10) a movie or TV show might receive based on various features and characteristics. In this introduction, we will provide some background information and a brief data description related to this task.

Background :

IMDb, launched in 1990, is a valuable resource for movie enthusiasts, industry professionals, and critics. It houses a massive database of films and TV shows, along with user-generated ratings and reviews. These ratings provide insights into the popularity and quality of a given production, which can influence the choices of viewers and potential investors.

Predicting IMDb scores has several practical applications. Film studios and producers can use these predictions to gauge the potential success of a movie before its release. Additionally, streaming platforms can use such models to recommend content to users or decide which movies or shows to acquire for their libraries. Researchers and critics can also benefit from these predictions to evaluate the impact and reception of different productions over time.

Necessary Steps To Follow Prediction Of IMDb Score :

1. **Import Libraries**;

The essential libraries you might use for data acquisition and cleaning in an IMDb score prediction project in Python.

Program:

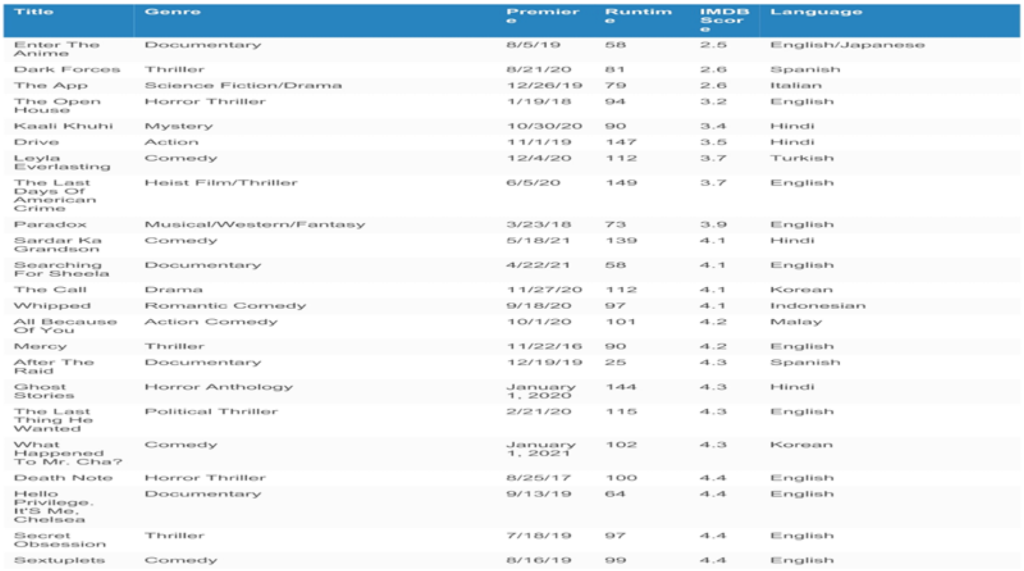
```
import pandas as pd

import numby as np

from sklearn.preprocessing import StandardScaler, LabelEncoder

from sklearn.model_selection import train_test_split

import seaborn as sns

import matplotlib.pyplot as plt
```

2. **Load The Dataset :**

To load a dataset for IMDb score prediction in Python, you can use the Pandas library, which is a powerful tool for data manipulation and analysis.

Program :

```
df = pd.read_csv('https://www.kaggle.com/datasets/luiscorter/netflix-original-filmsimdb-scores ')
```



Title	Genre	Premiere	Runtime	IMDB Score	Language
Enter The Anime	Documentary	8/5/19	58	2.5	English/Japanese
Dark Forces	Thriller	8/21/20	81	2.6	Spanish
The App	Science Fiction/Drama	12/26/19	79	2.6	Italian
The Open House	Horror Thriller	1/19/18	94	3.2	English
Kaali Khuhi	Mystery	10/30/20	90	3.4	Hindi
Drive	Action	11/1/19	147	3.5	Hindi
Leyla Everlasting	Comedy	12/4/20	112	3.7	Turkish
The Last Days Of American Crime	Heist Film/Thriller	6/5/20	149	3.7	English
Paradox	Musical/Western/Fantasy	3/23/18	73	3.9	English
Sardar Ka Grandson	Comedy	5/18/21	139	4.1	Hindi
Searching For Sheela	Documentary	4/22/21	58	4.1	English
The Call	Drama	11/27/20	112	4.1	Korean
Whipped	Romantic Comedy	9/18/20	97	4.1	Indonesian
All Because Of You	Action Comedy	10/1/20	101	4.2	Malay
Mercy	Thriller	11/22/16	90	4.2	English
After The Raid	Documentary	12/19/19	25	4.3	Spanish
Ghost Stories	Horror Anthology	January 1, 2020	144	4.3	Hindi
The Last Thing He Wanted	Political Thriller	2/21/20	115	4.3	English
What Happened To Mr. Cha?	Comedy	January 1, 2021	102	4.3	Korean
Death Note	Horror Thriller	8/25/17	100	4.4	English
Hello Privilege. It'S Me, Chelsea	Documentary	9/13/19	64	4.4	English
Secret Obsession	Thriller	7/18/19	97	4.4	English
Sextuplets	Comedy	8/16/19	99	4.4	English

3. Exploratory Data Analysis :

Exploratory Data Analysis (EDA) in IMDb score prediction refers to the systematic process of examining and understanding the IMDb dataset. It typically includes data visualization and summary statistics to gain insights into the data's distribution, relationships between variables, and potential data quality issues.

Program :

```
print(df.head())

plt.figure(figsize=(12, 6))

plt.figure(figsize=(8, 6))

sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm")

plt.title('Correlation Heatmap')

plt.subplot(1, 2, 1)

sns.histplot(df['IMDb'], kde=True)

plt.title('Distribution of IMDb Ratings'

plt.show()
```

4. Features Engineering :

Feature engineering in IMDb score prediction involves identifying, creating, or transforming features that can enhance the accuracy and predictive power of your model. These features are derived from the original dataset and often capture meaningful information that the model may not extract from the raw data.

Program :

```
def count_lead_actors(row):

lead_actors = sum([1 for actor_salary in row['ActorSalaries'] if actor_salary > 1000000])

return lead_actors

df['Number of Lead Actors'] = df.apply(count_lead_actors, axis=1)

print(df[['MovieTitle', 'Number of Lead Actors']])
```

5. Split The Data :

Splitting the data into training and testing sets in IMDb score prediction involves dividing your dataset into two subsets: one for training the prediction model and the other for testing the model's performance. This process helps you evaluate how well your model will perform on new, unseen data.

Program :

```
X = df[['Feature1', 'Feature2', ...]

y = df['IMDb']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

6. Features Scaling :

Feature scaling in IMDb score prediction refers to the process of standardizing or normalizing the numerical features in your dataset to ensure that they all have a similar scale.

Program :

```
X = df[['Budget', 'BoxOfficeEarnings', 'Runtime', 'NumUserReviews']]

y = df['IMDbScore']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

scaler = MinMaxScaler()

X_train_scaled = scaler.fit_transform(X_train)

X_test_scaled = scaler.transform(X_test)

model = LinearRegression()

model.fit(X_train_scaled, y_train)

y_pred = model.predict(X_test_scaled)

mse = mean_squared_error(y_test, y_pred)

r2 = r2_score(y_test, y_pred)

print(f"Mean Squared Error: {mse}")

print(f"R-squared (R2) Score: {r2}")
```

Importance Of Loading And Preprocessing Dataset:

Data Understanding:

It provides an opportunity to understand the dataset's structure, including the features and the target variable (IMDb scores). This understanding is essential for choosing the right features and building an effective model.

Feature Selection:

Preprocessing the dataset involves feature selection. By carefully selecting relevant features and removing irrelevant ones, you can improve the model's efficiency and reduce overfitting.

Data Cleaning:

Preprocessing includes data cleaning, where you can fix or remove inconsistent or erroneous data. Clean data ensures the model is trained on reliable and accurate information.

Data Transformation :

You can apply transformations to the data during preprocessing. For example, converting categorical variables into numerical representations and scaling numerical features to make them comparable. These transformations enhance the model's ability to interpret the data.

Handling Missing Values :

Missing data is common in real-world datasets. Preprocessing involves handling missing values by either imputing them with appropriate values or removing affected rows or columns.

Outlier Detection :

Detecting and addressing outliers is crucial for maintaining the model's performance. Outliers can significantly impact the model's predictive power.

Data Splitting :

Splitting the data into training and testing sets is a critical part of preprocessing. It allows you to evaluate your model's performance on unseen data and helps you avoid overfitting.

Challenges Involved In Loading And Preprocessing :

1. **Data Quality Issues:**

   Datasets may have missing values, duplicates, or errors that need to be addressed.

2. Handling Categorical Data:

IMDb datasets often include categorical variables like movie genres, which need to be converted into a numerical format.

3. Outliers:

Outliers can distort the model's predictive power.

4. Data Scaling:

Features may have different scales, which can affect the performance of some machine learning algorithms.

5. Imbalanced Data:

IMDb ratings may not be evenly distributed; certain ratings could be underrepresented in the dataset.

6. Data Privacy and Legal Considerations:

IMDb datasets may contain sensitive or copyrighted information.

7. Data Exploration and Understanding:

Challenge: Understanding the data and its relationships can be challenging, especially with a large and complex dataset.

8. Model Evaluation:

Challenge: Evaluating the IMDb score prediction model can be challenging due to the subjective nature of IMDb ratings.

Overcome Challenges :

1. **Data Quality Issues:**
   - Impute missing values using techniques like mean, median, or predictive modeling.
   - Remove duplicate records to avoid data redundancy.
   - Manually inspect and correct errors in the data, if possible.
2. **Handling Categorical Data:**
   - Use techniques like one-hot encoding or label encoding to convert categorical variables into numerical representations.
   - Be cautious with one-hot encoding, as it can lead to a high-dimensional dataset, potentially causing issues with model complexity.
3. **Outliers:**
   - Identify outliers through visualization and statistical methods.
   - Decide whether to remove, transform, or handle outliers depending on their impact on the model.
4. **Data Scaling:**
   - Standardize or normalize numerical features to bring them to a common scale.
   - Choose the appropriate scaling method based on your data distribution and the algorithms you intend to use.
5. **Imbalanced Data:**
   - Implement techniques like oversampling, undersampling, or synthetic data generation to balance the dataset.
   - Choose evaluation metrics that are robust to class imbalances, like F1-score or area under the ROC curve (AUC).
6. **Data Privacy and Legal Considerations:**
   - Ensure compliance with data privacy regulations (e.g., GDPR) by anonymizing or aggregating sensitive data.
   - Respect IMDb's terms of use and data licensing agreements when using IMDb

datasets.

7. **Data Exploration and Understanding:**
    - Perform exploratory data analysis (EDA) to visualize and understand the data's distribution, correlations, and patterns.
    - Use domain knowledge to guide feature selection and engineering.
8. **Model Evaluation:**
    - Use appropriate evaluation metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) that measure prediction accuracy.
    - Consider collecting user feedback and iteratively improving the model based on user preferences

Conclusion :

Predicting IMDb scores involves a multifaceted process that includes data exploration, data cleaning, data visualization, and data preprocessing. Here's a comprehensive conclusion that highlights the key aspects of this predictive task:

Data Exploration:

- Data exploration is the initial step in understanding the IMDb dataset.
- It helps identify patterns, outliers, and relationships in the data.
- Data exploration provides insights into the distribution of IMDb scores, the range of features, and their potential impact on scores.

Data Cleaning:

- Data cleaning is crucial to ensure data quality and reliability for predictions.
- It involves handling missing values, outliers, and inconsistent data.
- Cleaning the dataset helps create a more accurate and reliable basis for modeling.

Data Visualization:

- Data visualization is a powerful tool for understanding the dataset visually.
- Visualizations, such as histograms, scatter plots, and box plots, provide insights into data distributions and correlations.
- Visualizations help identify trends and relationships that may impact IMDb scores.

Data Preprocessing:

- Data preprocessing is essential to prepare the dataset for modeling.
- It includes feature selection, encoding categorical variables, scaling numerical features, and handling target variable transformations.
- Preprocessing ensures that the data is in a suitable format for machine learning algorithms.
- In summary, predicting IMDb scores is a multifaceted task that requires a combination of data exploration, data cleaning, data visualization, and data preprocessing to prepare the dataset and gain insights. These steps set the foundation for building predictive models that can accurately estimate IMDb scores, providing valuable information for various stakeholders in the movie industry and beyond.