MOVIE ANALYSIS, RECOMMENDATION AND SUCCESS PREDICTION

PROJECT REPORT

By

ABHINAV P

UNDER THE GUIDANCE OF

Miss SHAYANA P S

TECHOLAS TECHNOLOGIES

Kochi, Kerala, 682025

CONTENTS

INTRODUCTION

The Movie Analysis, Recommendation, and Success Prediction Project applies data analytics and machine learning to uncover key factors driving success in the film industry. With the growing influence of big data, AI, and streaming platforms, understanding audience preferences, financial trends, and marketing effectiveness has become crucial for filmmakers and production houses.

The global film industry contributes significantly to entertainment and the economy but faces high financial risks. Many films fail to recover their investments, while others generate massive box office revenues. By leveraging predictive modeling, sentiment analysis, and interactive dashboards, this project helps industry stakeholders make data-driven decisions, minimizing risks and optimizing content strategies.

Using machine learning models like logistic regression, decision trees, and random forests, this project predicts whether a movie will be successful based on factors such as budget, genre, audience reviews, and past performance. It also provides recommendations to improve marketing, budget planning, and audience targeting. Streaming platforms can use these insights to suggest content that matches viewer preferences, increasing engagement.

By applying data science to the film industry, this project helps studios, filmmakers, and streaming services make better decisions, create engaging content, and increase profits in a highly competitive market.

ABSTRACT

This project aims to develop a Movie Analysis, Recommendation, and Success Prediction Model using machine learning techniques to analyse movie trends, recommend films, and predict their likelihood of success. The goal is to help filmmakers, producers, and streaming platforms make data-driven decisions regarding investments, marketing, and audience engagement by identifying key success factors such as budget, revenue, popularity, ratings, and genre trends.

The analysis is based on a comprehensive dataset that includes financial details, audience preferences, and movie characteristics. The dependent variable in this study is a movie's success status, while the independent variables include budget, revenue, popularity, vote count, average ratings, and other key factors. By identifying patterns in these variables, the model uncovers the critical elements that drive box office performance and audience reception.

Beyond success prediction, the project also features a movie recommendation system that suggests films based on user preferences. Using advanced analytics and AI-driven recommendations, streaming platforms can enhance audience engagement by offering personalized content tailored to individual tastes.

By combining predictive modelling and data science, this project enables the film industry to optimize movie production, refine marketing strategies, and reduce financial risks. The insights gained from this analysis can help studios make better investment decisions, improve content quality, and maximize profitability in a highly competitive market.

OBJECTIVE

- **Develop an Optimized Machine Learning Model:**
  - Implement and fine-tune classification algorithms (Logistic Regression, Decision Tree, Random Forest, etc.) to predict movie success.
  - Evaluate model performance using metrics like accuracy, precision, recall, and F1-score.

- **Feature Importance & Impact Analysis:**
  - Identify key factors affecting movie success, such as budget, revenue, popularity, vote count, and ratings.
  - Analyse the influence of genre, production company, and audience engagement on box office performance.

- **Data Visualization & Dashboard Development:**
  - Create interactive visualizations to analyse trends in the movie industry, including revenue, ratings, and success ratios.
  - Design a Power BI dashboard to display top genres, highest-grossing movies, and financial insights.

- **Movie Recommendation System:**
  - Develop a genre-based recommendation system to suggest movies to users based on their preferences.
  - Integrate the recommendation system into an interactive Power BI dashboard for better user experience.

DATA ANALYSIS

ABOUT DATASET

Source: Kaggle

Columns:

1. Financial Metrics
    - Budget – Production cost of the movie.
    - Revenue – Total earnings of the movie.
    - Profit – Revenue minus budget (calculated field).

2. Ratings & Popularity
    - Popularity – Popularity score based on audience engagement.
    - Vote Average – Average rating given by users.
    - Vote Count – Total number of votes received.

3. Movie Attributes
    - Adult – Indicates if the movie is rated as adult content (True/False).
    - Runtime – Duration of the movie in minutes.
    - Video – Specifies if a video version is available (True/False).

4. Success Prediction Variable
    - Success – The target variable indicating whether the movie was successful or not.

SUCCESS PREDICTION

1.Import Essential Libraries

- Import pandas, NumPy for data handling.
- Load scikit-learn models for machine learning tasks.

2.Load the Data

- Read the dataset using pandas. read_csv().
- Display dataset structure using .info(), .head(), and .describe().

3.Check for Missing Values and Clean Data

- Identify missing values using .isnull().sum().
- Handle missing values through imputation (mean/median/mode) or removal.

4.Drop Irrelevant Columns

- Remove non-essential columns that do not impact success prediction.

5.Assign Dependent and Independent Variables

- Dependent Variable: Success (Yes/No).
- Independent Variables:
  - Budget (Financial).
  - Popularity, Vote Average, Vote count (Rating and popularity).
  - Adult, run time, Video (Movie Attributes).

6.Encode Categorical Variables

- Convert categorical data (e.g., Adult, Video, Success) using Label Encoding

7.Split the Data (Train-Test Split)

- Use train_test_split () to divide data into 80% training and 20% testing.

8.Train Models Using Classification Algorithms

- Apply multiple classification models to predict job placement:
  - Logistic Regression
  - Random Forest Classifier
  - Decision Tree Classifier

9. Evaluate Model Performance

- Use accuracy, precision, recall, F1-score to compare model effectiveness.

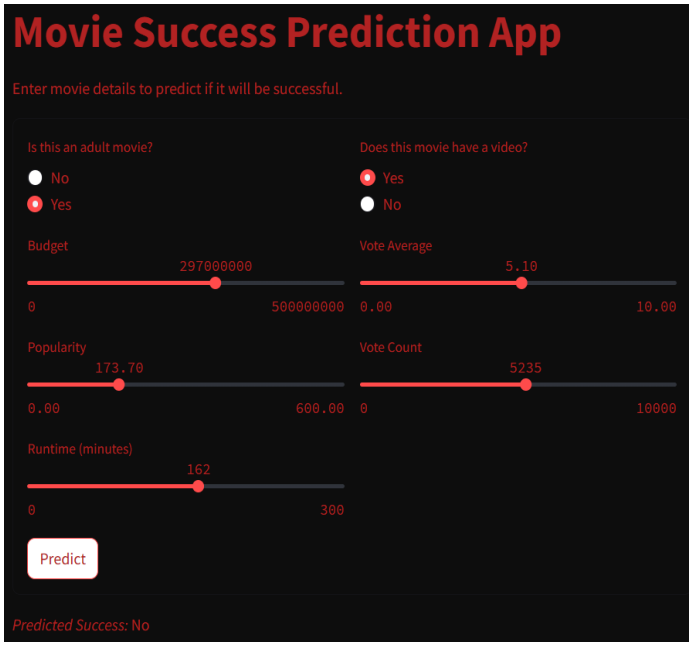
10.Create a Web Interface

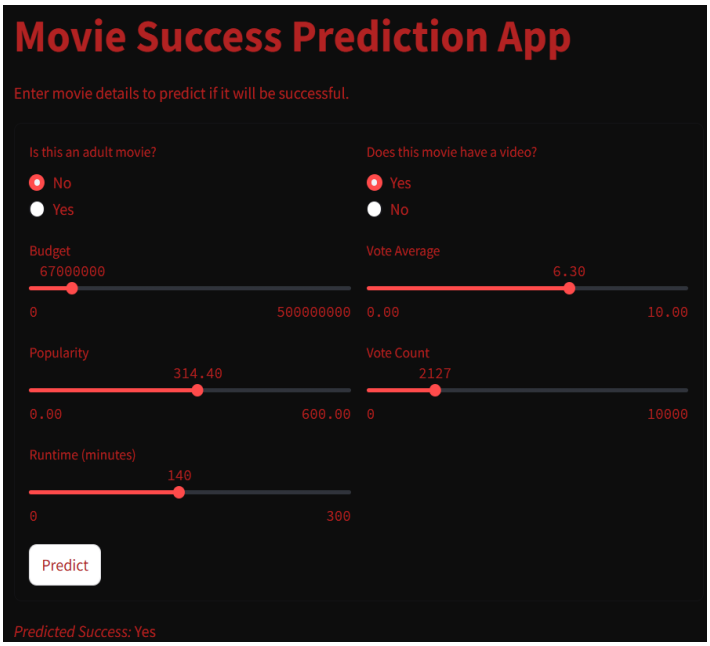
- Created a web interface for users to input values for predicting using Stream lit by taking the model with highest accuracy as the pickle file.

11.Make Predictions and Generate Insights

- Input test data into the Stream lit Interface and receive the prediction.

USER INTERFACE:





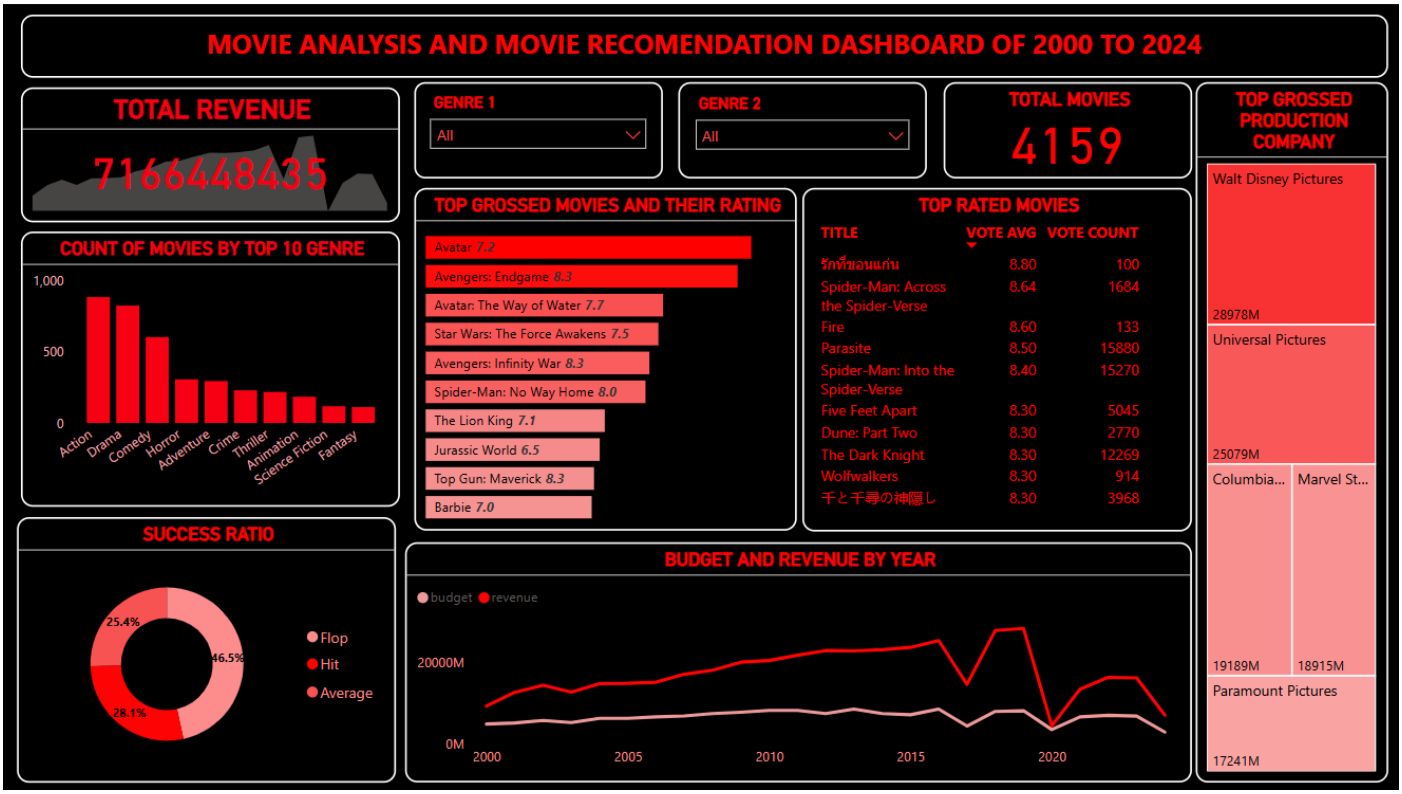
VISUALIZATION & DASHBOARD DEVELOPMENT

The project includes a **Power BI dashboard** to visualize key insights from the movie dataset, enabling stakeholders to interpret trends effectively.

1. **Movie Analysis Dashboard in Power BI**
    - **Count of Movies by Genre** – A bar chart representing the number of movies in each genre.
    - **Success Ratio Analysis** – A donut chart displaying the proportion of successful vs. unsuccessful movies.
    - **Top Grossing Movies & Their Ratings** – A horizontal bar chart highlighting the highest-grossing movies with their corresponding ratings.
    - **Budget vs. Revenue Trends** – A line chart showing how budgets and revenues have evolved over time.
    - **Top Production Companies by Revenue** – A tree map visualizing the top revenue-generating studios.
    - **Total Revenue & Movie Count** – KPI metric cards summarizing total earnings and the number of movies in the dataset.

2. **Movie Recommendation System in Power BI**
    - **Genre-Based Filtering** – Allows users to filter and explore movies by their preferred genres.
    - **Dynamic Insights** – Enables users to interact with data to discover top-rated and trending movies.

DASHBOARD OVERVIEW:



METHODOLOGY

1. **Jupyter Notebook**
    - Jupyter Notebook is used for programming in Python for machine learning tasks, including data preprocessing, analysis, and model building.
2. **Libraries**
    - Pandas – Used for data cleaning, transformation, analysis, visualization, and input/output operations.
    - NumPy – Supports numerical operations, including array manipulations, mathematical functions, and statistical calculations.

2. **Machine Learning Models for Classification**
    - Logistic Regression – Used for binary classification tasks such as predicting whether a movie will be successful (Success/Not Successful).
    - Decision Tree Classifier – Splits data into branches based on feature conditions, forming a tree-like structure to make classification predictions.
    - Random Forest Classifier – Employs multiple decision trees to classify movies into success categories.
3. **Model Evaluation & Preprocessing**
    - **Accuracy Score** – Evaluates the classification models' performance in predicting movie success.
    - **Train-Test Split** – Splits the dataset into training and testing sets for model evaluation, reducing overfitting.
    - **Label Encoding** – Converts categorical data into numerical values for machine learning models.
4. **Power BI**

    It is a business intelligence tool by Microsoft that allows users to visualize data, create interactive reports, and gain insights through dynamic dashboards.
    - **Tile** is an individual visual component within a dashboard, representing data through charts, graphs, or KPIs.
    - **Dashboard** in Power BI is a collection of visual elements that display key metrics, trends, and analytics in a single view.

RECOMMENDATIONS

1. Data-Driven Decision Making:

- Utilize Predictive Analytics: Film studios should leverage machine learning models to predict the commercial success of a movie based on historical trends, audience preferences, and budget allocation.
- Enhance Data Collection: Incorporate sentiment analysis from social media, audience reviews, and real-time engagement metrics to refine marketing strategies.

2. Budget Allocation Strategies:

- Optimize Production Spending: Allocate budgets based on past success patterns, ensuring high-return investments in cast, special effects, and marketing.
- Diversify Funding: Invest in various genres and emerging film markets to mitigate risks associated with high-budget flops.

3. Audience-Centric Content Development:

- Genre and Trend Analysis: Identify genres with growing popularity and tailor scripts accordingly.

4. Marketing Optimization:

- Targeted Advertising: Use machine learning models to analyze audience demographics and optimize ad placements across platforms.
- Viral Marketing Strategies: Leverage influencer collaborations, social media trends, and teaser campaigns to generate pre-release buzz.

5. Streaming and Theatrical Release Strategies:

- Hybrid Release Models: Adopt simultaneous theatrical and digital releases based on audience behaviours and geographic trends.
- Subscription-Based Personalization: Streaming platforms should refine their algorithms to suggest high-engagement movies based on viewing habits.

6. Post-Release Performance Analysis:

- Continuous Monitoring: Track movie performance across different platforms post-release to refine future production and marketing strategies.
- Audience Feedback Integration: Use real-time reviews and ratings to improve sequel production or series continuation decisions.

CONCLUSION

Key performance indicators such as profitability, audience ratings, and revenue trends were analysed, enabling stakeholders to pinpoint critical success factors in the film industry. The project revealed a strong correlation between budget allocation, marketing efforts, and audience engagement, with genres like action and sci-fi consistently achieving higher success rates.

By applying machine learning models such as logistic regression, decision trees, and random forest classifiers, the project successfully predicted movie success and offered strategic recommendations.

The findings highlight essential areas for industry improvement, including data-driven decision-making, dynamic marketing strategies, and AI-powered audience engagement. Leveraging predictive analytics will allow studios, distributors, and streaming platforms to refine content strategies, maximize profitability, and align offerings with evolving consumer preferences.

Moving forward, integrating real-time market trends, AI-driven personalization, and sentiment analysis will enhance content performance and audience satisfaction. By embracing data science and machine learning, the film industry can revolutionize content creation, boost revenue, and deliver compelling cinematic experiences tailored to audience demands.

APPENDIX I

	adult	budget	original_language	popularity	release_date	revenue	runtime	video	vote_average	vote_count
0	False	30000000	en	21.946943	1995-10-30	373554033.0	81.0	False	7.7	5415.0
1	False	65000000	en	17.015539	1995-12-15	262797249.0	104.0	False	6.9	2413.0
2	False	0	en	11.7129	1995-12-22	0.0	101.0	False	6.5	92.0
3	False	16000000	en	3.859495	1995-12-22	81452156.0	127.0	False	6.1	34.0
4	False	0	en	8.387519	1995-02-10	76578911.0	106.0	False	5.7	173.0
...	...	...	...	...	...	...	...	...	...	...
45461	False	0	fa	0.072051	NaN	0.0	90.0	False	4.0	1.0
45462	False	0	tl	0.178241	2011-11-17	0.0	360.0	False	9.0	3.0
45463	False	0	en	0.903007	2003-08-01	0.0	90.0	False	3.8	6.0
45464	False	0	en	0.003503	1917-10-21	0.0	87.0	False	0.0	0.0
45465	False	0	en	0.163015	2017-06-09	0.0	75.0	False	0.0	0.0

45014 rows × 10 columns

APPENDIX II

```
import pandas as pd
data=pd.read_csv(r"data_path")
data.head()
data.columns
data.isnull().sum()
```

Import the models

```
from sklearn.preprocessing import LabelEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import RandomForestClassifier
```

Encoding:

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
categorical_cols=["adult" ,"video"]
for col in categorical_cols:
    df[col]=le.fit_transform(df[col])
```

Dependent and independent variable:

```
x=df.drop(columns=['success','revenue','profit'])
y=df.success
```

Model training and Evaluating:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=41)

model1=linear_model.LogisticRegression()
model1.fit(x_train,y_train)

model2=DecisionTreeClassifier()
model2.fit(x_train,y_train)

model3=RandomForestClassifier(n_estimators=20)
model3.fit(x_train,y_train)

from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
def evaluate_model(y_true, y_pred):
    accuracy = accuracy_score(y_true, y_pred)
    precision = precision_score(y_true, y_pred, average='weighted')  # Use 'weighted' for multiclass
    recall = recall_score(y_true, y_pred, average='weighted')
    f1 = f1_score(y_true, y_pred, average='weighted')

    print(f"Accuracy: {accuracy*100}")
    print(f"Precision: {precision*100}")
    print(f"Recall: {recall*100}")
    print(f"F1-Score: {f1*100}")
    return accuracy, precision, recall, f1

y_pred1 = model1.predict(x_test)
y_pred2 = model2.predict(x_test)
y_pred3 = model3.predict(x_test)
```

```
print("LogisticRegression Metrics:")
evaluate_model(y_test, y_pred1)

print("\n DecisionTree Metrics:")
evaluate_model(y_test, y_pred2)

print("\n RandomForest Metrics:")
evaluate_model(y_test, y_pred3)
```

```
LogisticRegression Metrics:
Accuracy: 75.29610829103216
Precision: 76.27238801088492
Recall: 75.29610829103216
F1-Score: 74.2537721950106

 DecisionTree Metrics:
Accuracy: 70.3327693175409
Precision: 70.27298244732614
Recall: 70.3327693175409
F1-Score: 70.29963677128356

 RandomForest Metrics:
Accuracy: 76.3677382966723
Precision: 76.33694233039012
Recall: 76.3677382966723
F1-Score: 76.01542396258694
(0.763677382966723, 0.7633694233039012, 0.763677382966723, 0.7601542396258694)
```