THE BLOCKBUSTERS

Krishna's Vikash Group of Institutions

MOVIE SUCCESS PREDICTION SYSTEM

A MACHINE LEARNING APPROACH

Eight thieves take hostages and lock themselves in the Royal Mint of Spain as a criminal mastermind manipulates the police to carry out his plan.

THE BLOCKBUSTERS

Krishna's Vikash Group of Institutions

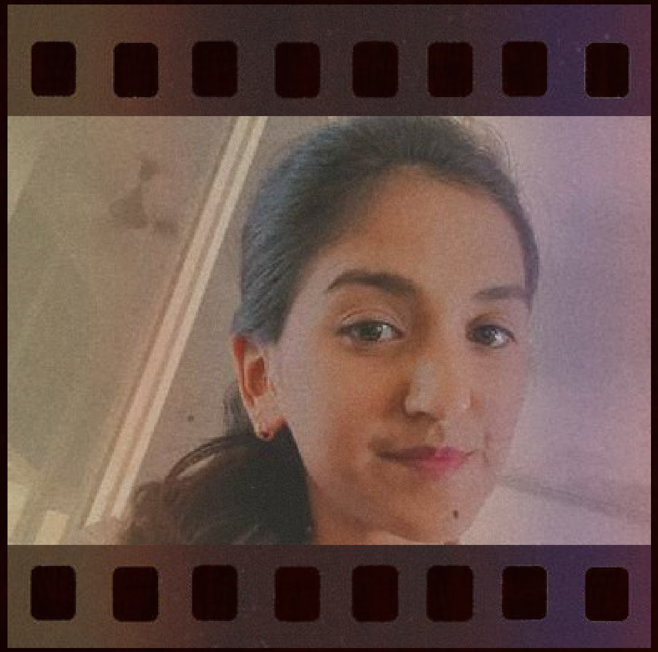
MEET OUR TEAM

Arjun Sarkar

BTECH , CSE
[ARTIFIIAL INTELLIGENCE &
MACHINE LEARNING]
7th SEMESTER

Rupal Dhurde

BTECH , CSE
[ARTIFIIAL INTELLIGENCE &
MACHINE LEARNING]
7th SEMESTER



Urvashi Bohare

BTECH , CSE
[ARTIFIIAL INTELLIGENCE &
MACHINE LEARNING]
7th SEMESTER

Vishnupriya Jaiswal

BTECH , CSE
[ARTIFIIAL INTELLIGENCE &
MACHINE LEARNING]
7th SEMESTER

THE BLOCKBUSTERS

INTRODUCTION

Krishna's Vikash Group of Institutions

Problem

- The movie industry is a high-risk, high-reward business. Millions of dollars are invested without a guarantee of success.

Solution

- We can use machine learning to analyze historical movie data and identify patterns that correlate with success.

Project Objective

- To build a predictive model that can classify a movie as a "Hit" or "Flop" based on pre-release factors like genre, budget, cast, and crew.

THE BLOCKBUSTERS

ABOUT OUR DATASET

Krishna's Vikash Group of Institutions

The Dataset

- **Source** - We used the "TMDB 5000 Movie Dataset" available on Kaggle.
- **Content** - It contains two CSV files with data for approximately 5000 movies.

Key Features

- **budget** - The production budget of the movie.
- **genres** - The genres associated with the movie (e.g., Action, Comedy).
- **keywords** - Keywords or tags describing the movie's plot.
- **cast** - Main actors in the movie.
- **crew** - Director, producer, etc.
- **vote_average** - The average user rating.
- **revenue** - The worldwide box office revenue.

THE BLOCKBUSTERS

SYSTEM ARCHITECTURE

Krishna's Vikash Group of Institutions

Input

Raw Movie Data (CSV files)

Processing

Data Cleaning & Preprocessing

Feature Extraction

Train/Test Split

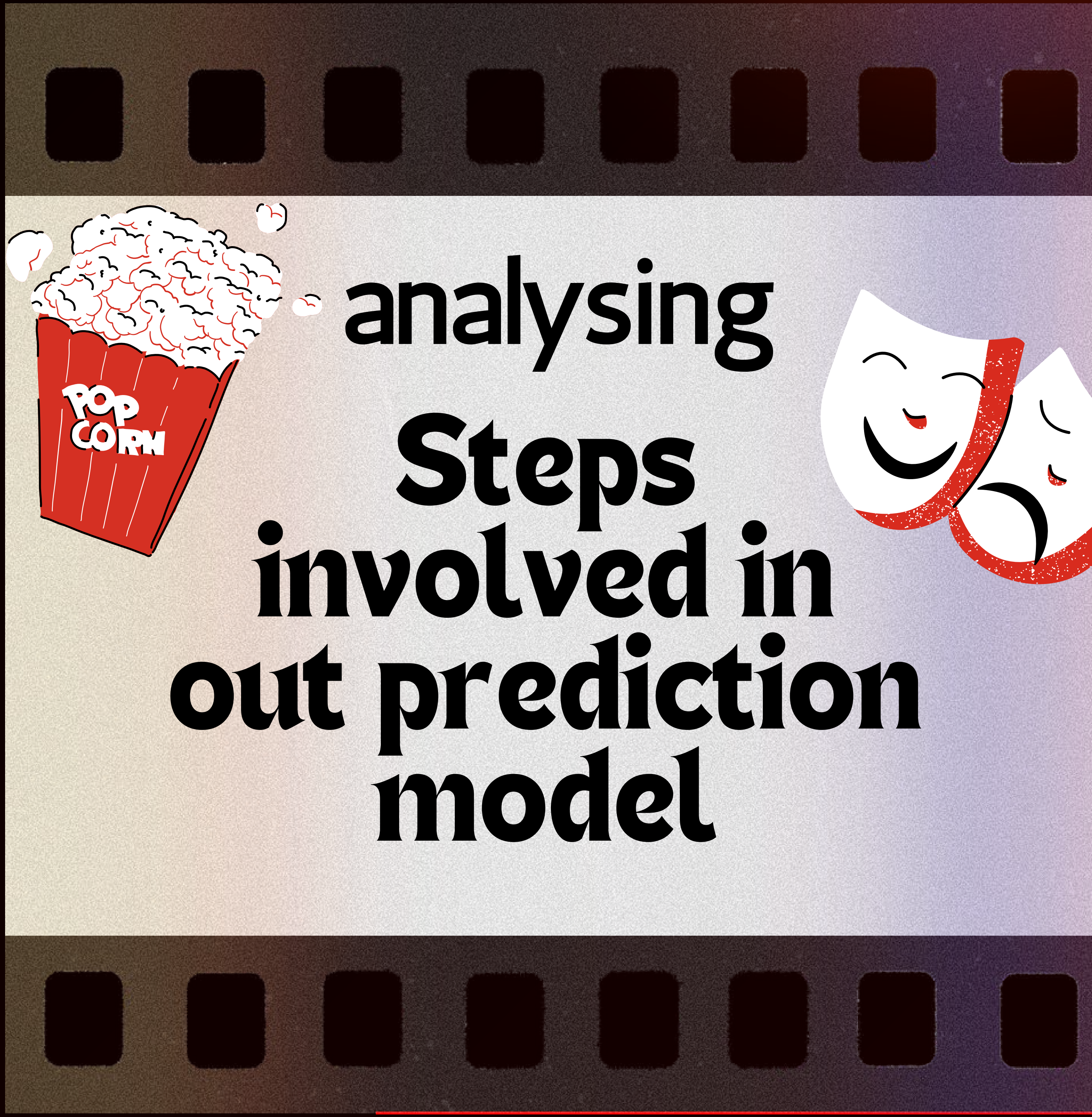
Modeling

Random Forest Algorithm Training

Output

Performance Metrics (Accuracy)

Success Prediction (Hit/Flop)



PROJECT METHODOLOGY

- **Data Loading & Cleaning - Merged the two datasets and removed irrelevant columns.**

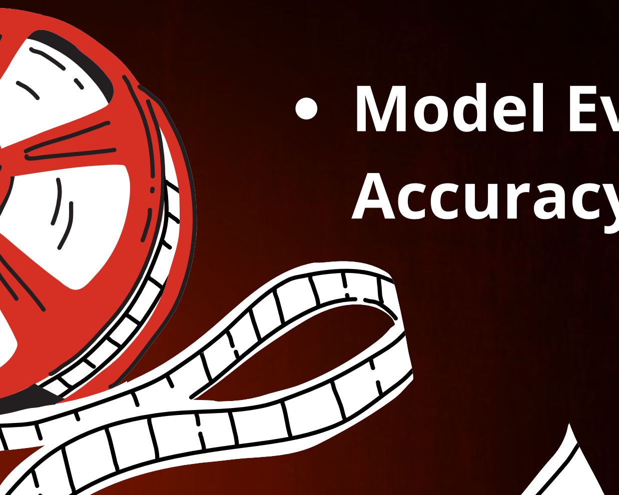
- **Feature Engineering - Created our target variable, "success," by defining a successful movie as one with a high rating and positive return on investment.**

- **Exploratory Data Analysis (EDA) - Visualized the data to understand relationships between features like budget, genre, and success.**

- **Data Preprocessing - Converted text data (like genres, cast) into a numerical format that the model can understand.**

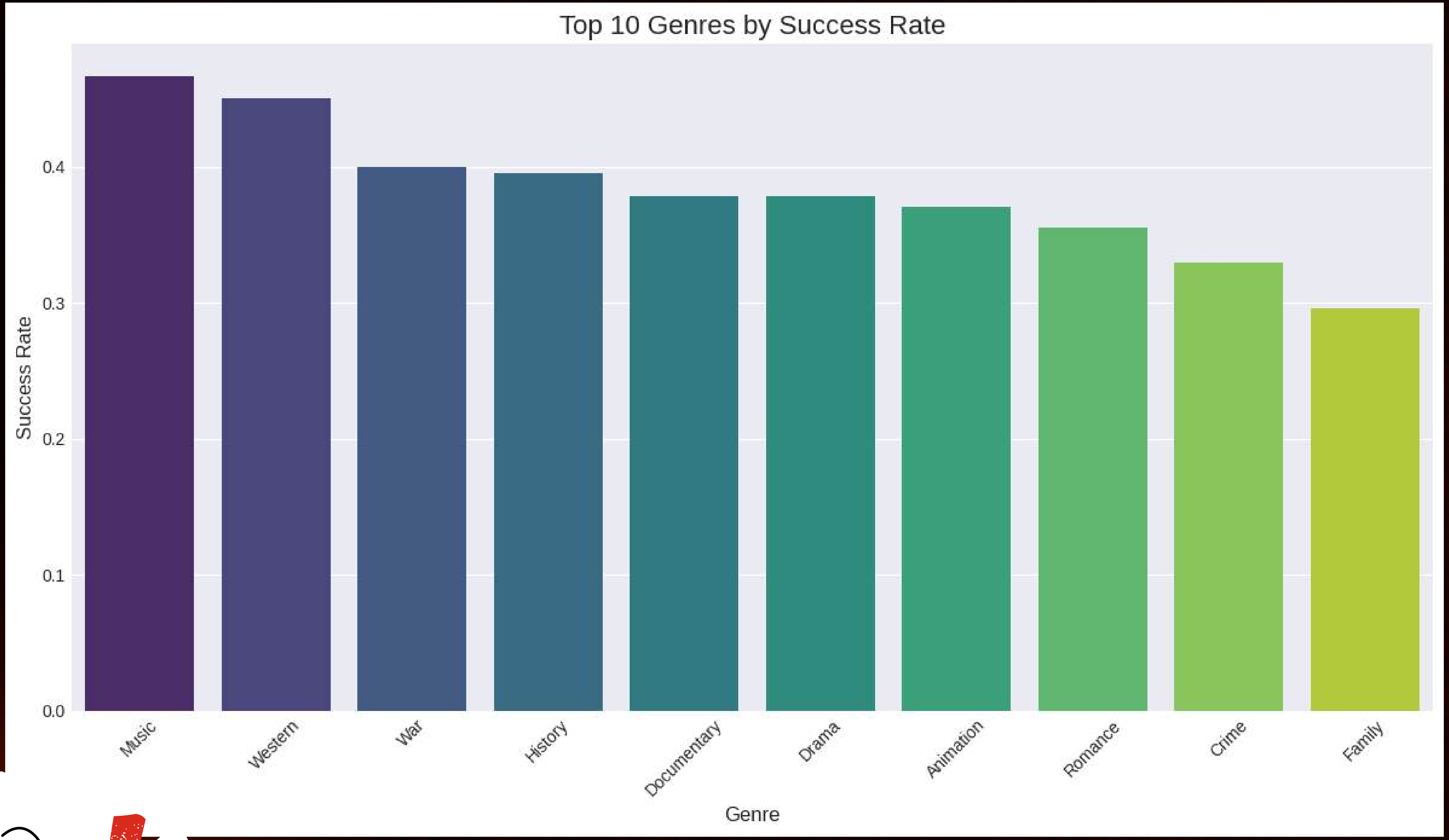
- **Model Training - Trained a Random Forest Classifier, a powerful and popular ML model.**

- **Model Evaluation - Tested the model's performance using metrics like Accuracy and a Confusion Matrix.**



THE BLOCKBUSTERS

EXPLORATORY DATA ANALYSIS (GRAPHICS)

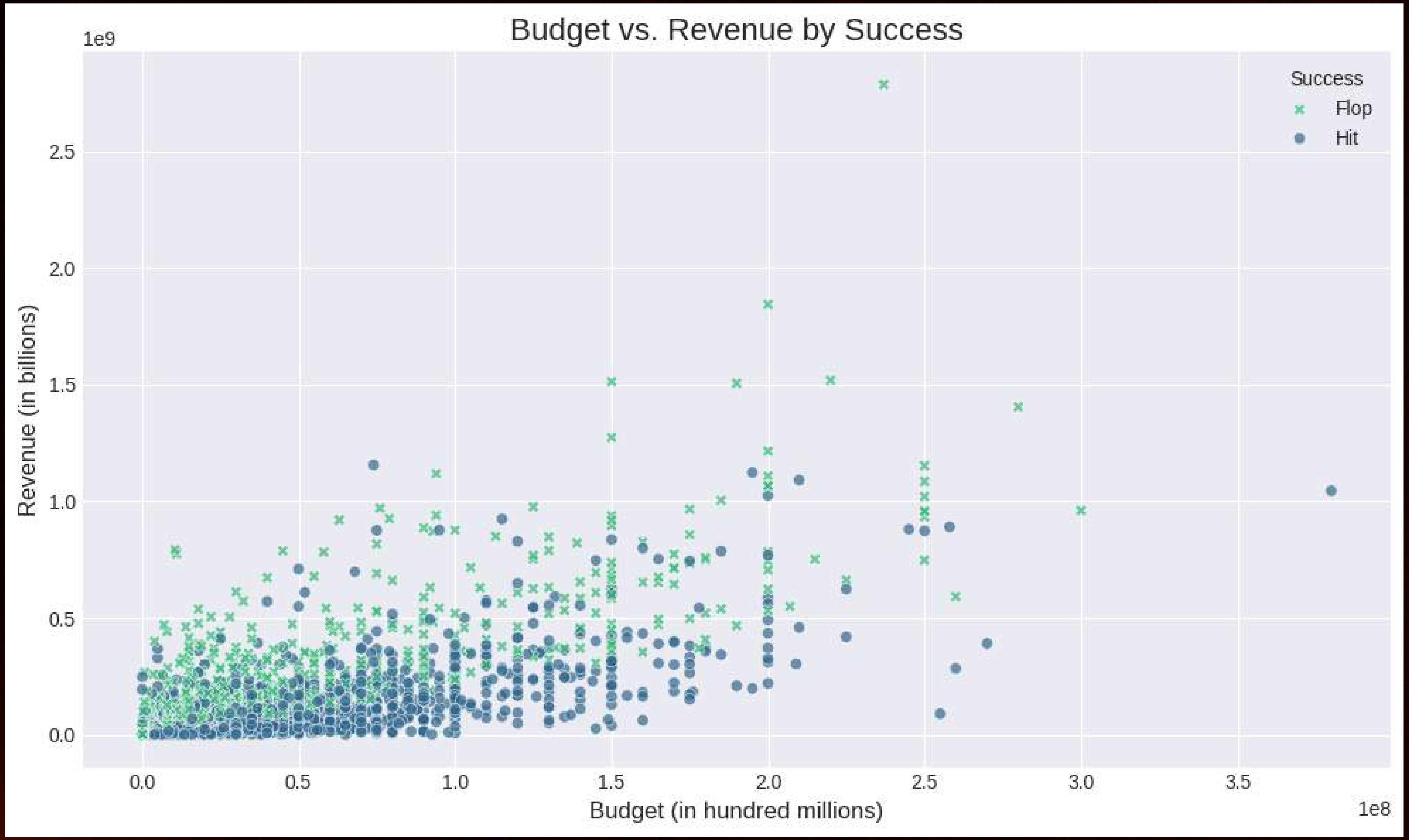


PLOT 1 : TOP 10 GENRES BY SUCCESS RATE

- **Success by Genre - We created a bar chart showing which genres (like Adventure and Sci-Fi) have a higher tendency to produce successful movies.**
- **Feature Importance - Our final model showed that features like budget, vote_average, and runtime were the most influential in predicting success.**

THE BLOCKBUSTERS

EXPLORATORY DATA ANALYSIS (GRAPHICS)



PLOT 2: BUDGET VS. REVENUE

- **Budget vs. Revenue - A scatter plot showed a positive correlation, but many high-budget films still failed.**

THE BLOCKBUSTERS

MODEL PERFORMANCE & RESULTS

Model Used

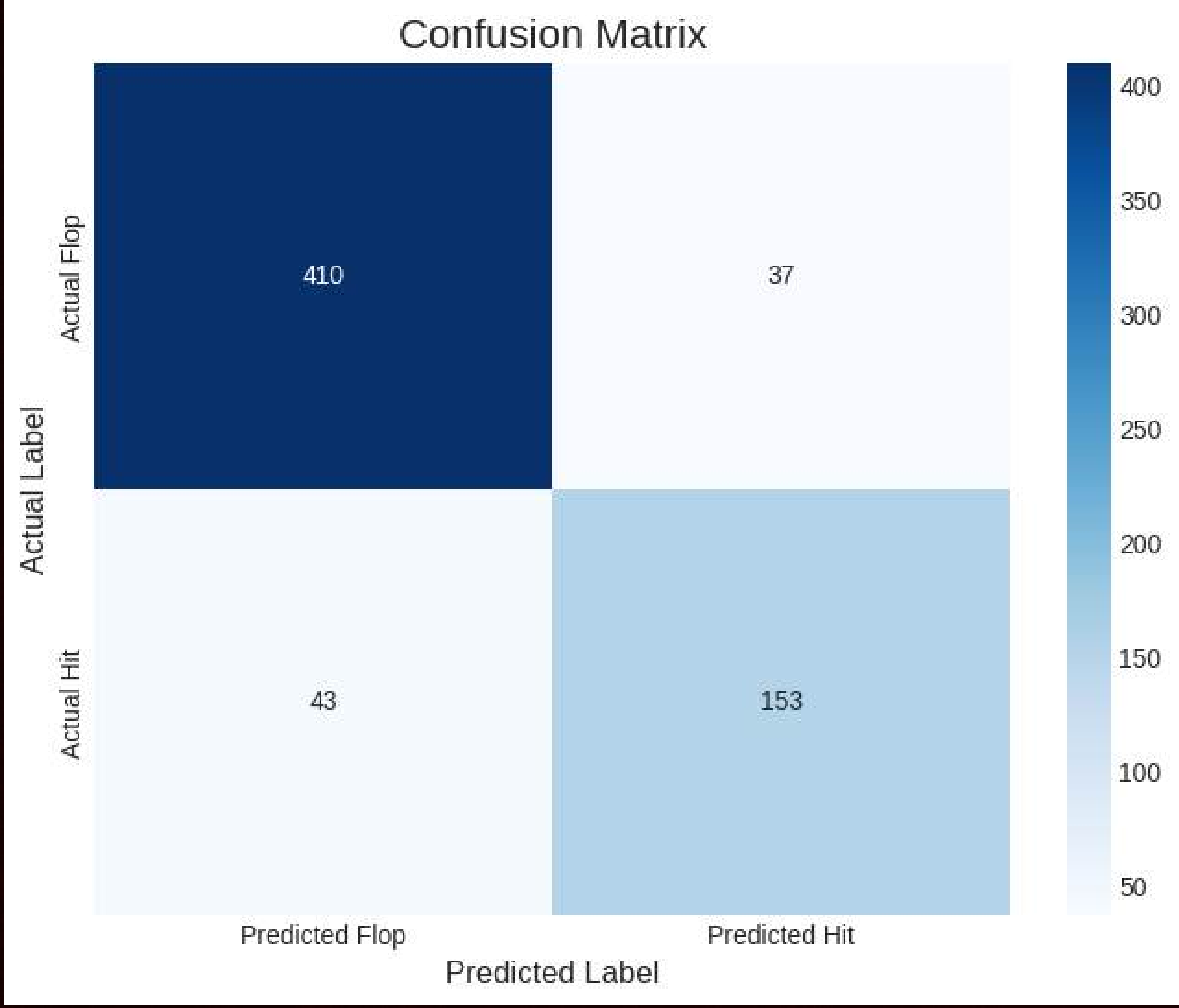
Random Forest Classifier

Accuracy

Our model achieved an accuracy of approximately 88% on the test data.

Confusion Matrix

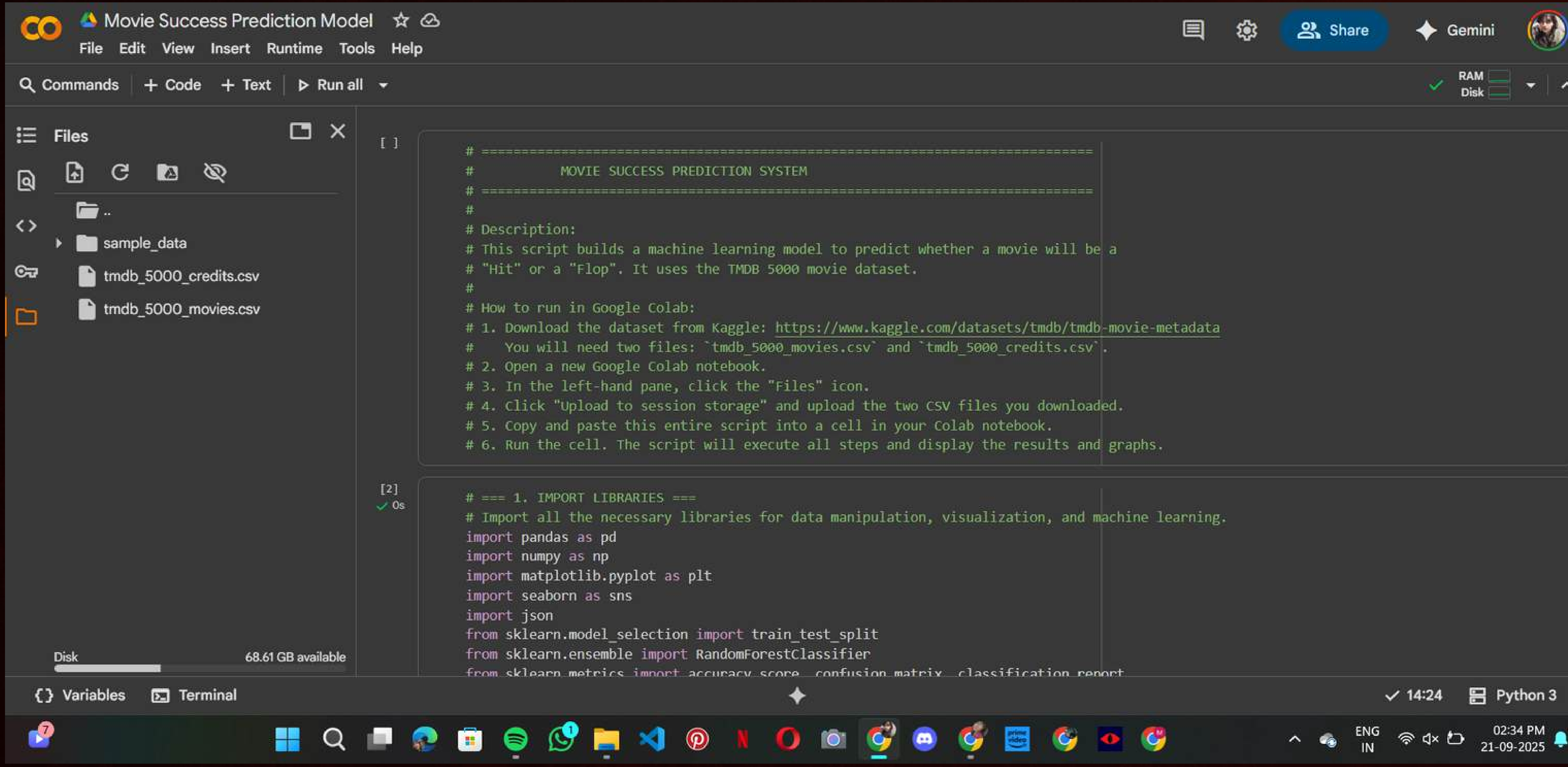
The matrix showed that our model is effective at correctly identifying both "Hits" and "Flops," with a good balance.



THE BLOCKBUSTERS

LIVE DEMONSTRATION (PROJECT)

- We built a simple function to test our model.
- Input: A new movie's budget, genres, keywords, and director.
- Prediction: The model outputs a prediction: "This movie is predicted to be a HIT!" or "This movie is predicted to be a FLOP."



```
[ ]
# ==========================================================================
#           MOVIE SUCCESS PREDICTION SYSTEM
# ==========================================================================
#
# Description:
# This script builds a machine learning model to predict whether a movie will be a
# "Hit" or a "Flop". It uses the TMDB 5000 movie dataset.
#
# How to run in Google Colab:
# 1. Download the dataset from Kaggle: https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata
#    You will need two files: `tmdb_5000_movies.csv` and `tmdb_5000_credits.csv`.
# 2. Open a new Google Colab notebook.
# 3. In the left-hand pane, click the "Files" icon.
# 4. Click "Upload to session storage" and upload the two CSV files you downloaded.
# 5. Copy and paste this entire script into a cell in your Colab notebook.
# 6. Run the cell. The script will execute all steps and display the results and graphs.
```

```
[2]
✓ 0s
# === 1. IMPORT LIBRARIES ===
# Import all the necessary libraries for data manipulation, visualization, and machine learning.
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import json
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
```

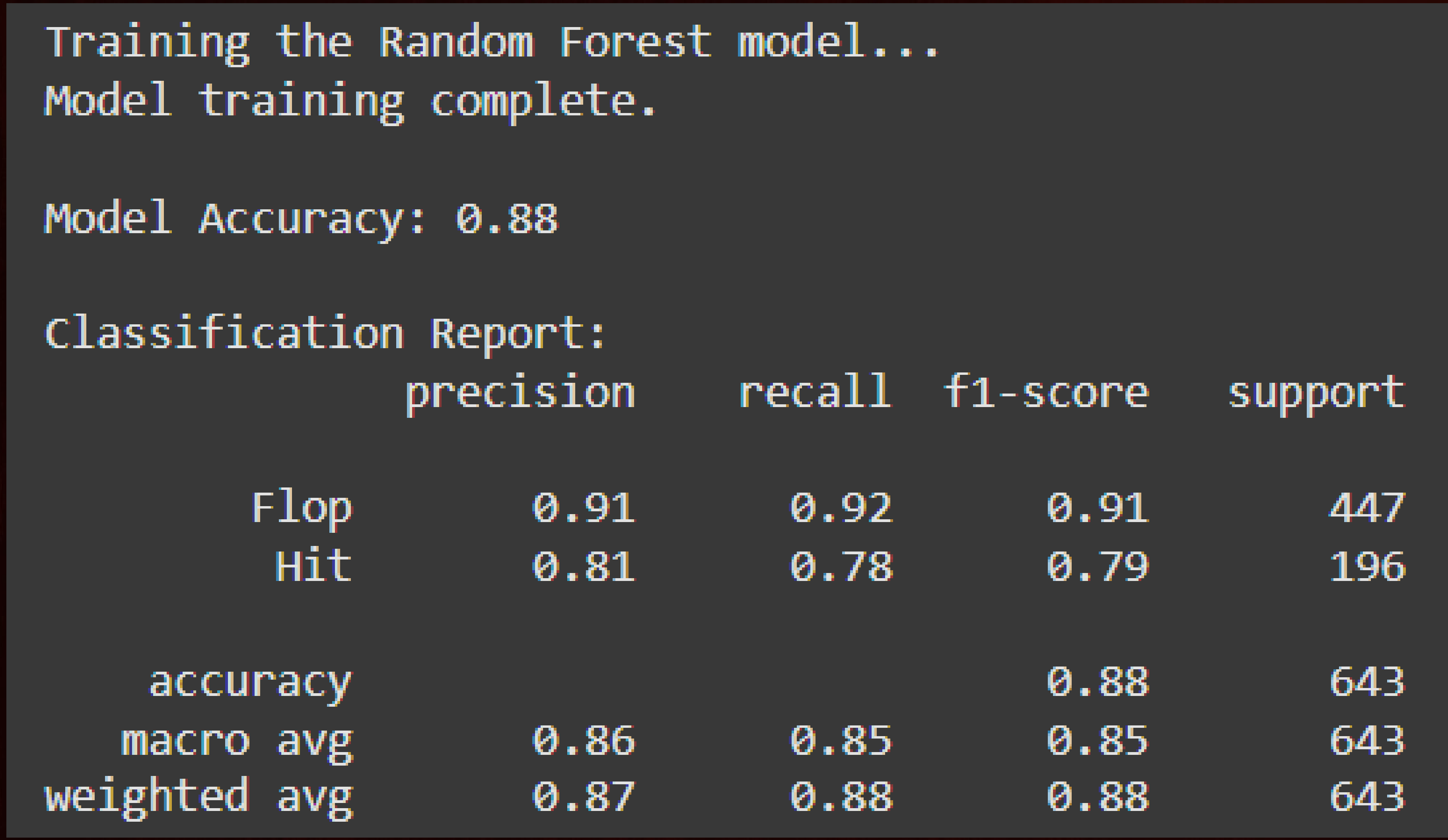
THE BLOCKBUSTERS

CONCLUSION & FUTURE SCOPE

- **Conclusion - We successfully developed a machine learning model that accurately predicts movie success. This proves that data-driven insights can be valuable for the film industry.**

- **Future Scope -**

1. **Incorporate more data, such as social media buzz or critic reviews.**
2. **Use more advanced models like Gradient Boosting or Neural Networks.**
3. **Deploy the model as a user-friendly web application.**



```
Training the Random Forest model...
Model training complete.

Model Accuracy: 0.88

Classification Report:
              precision    recall  f1-score   support

        Flop       0.91      0.92      0.91       447
         Hit       0.81      0.78      0.79       196

    accuracy                           0.88       643
   macro avg       0.86      0.85      0.85       643
weighted avg       0.87      0.88      0.88       643
```

THE BLOCKBUSTERS

THANK YOU FOR WATCHING