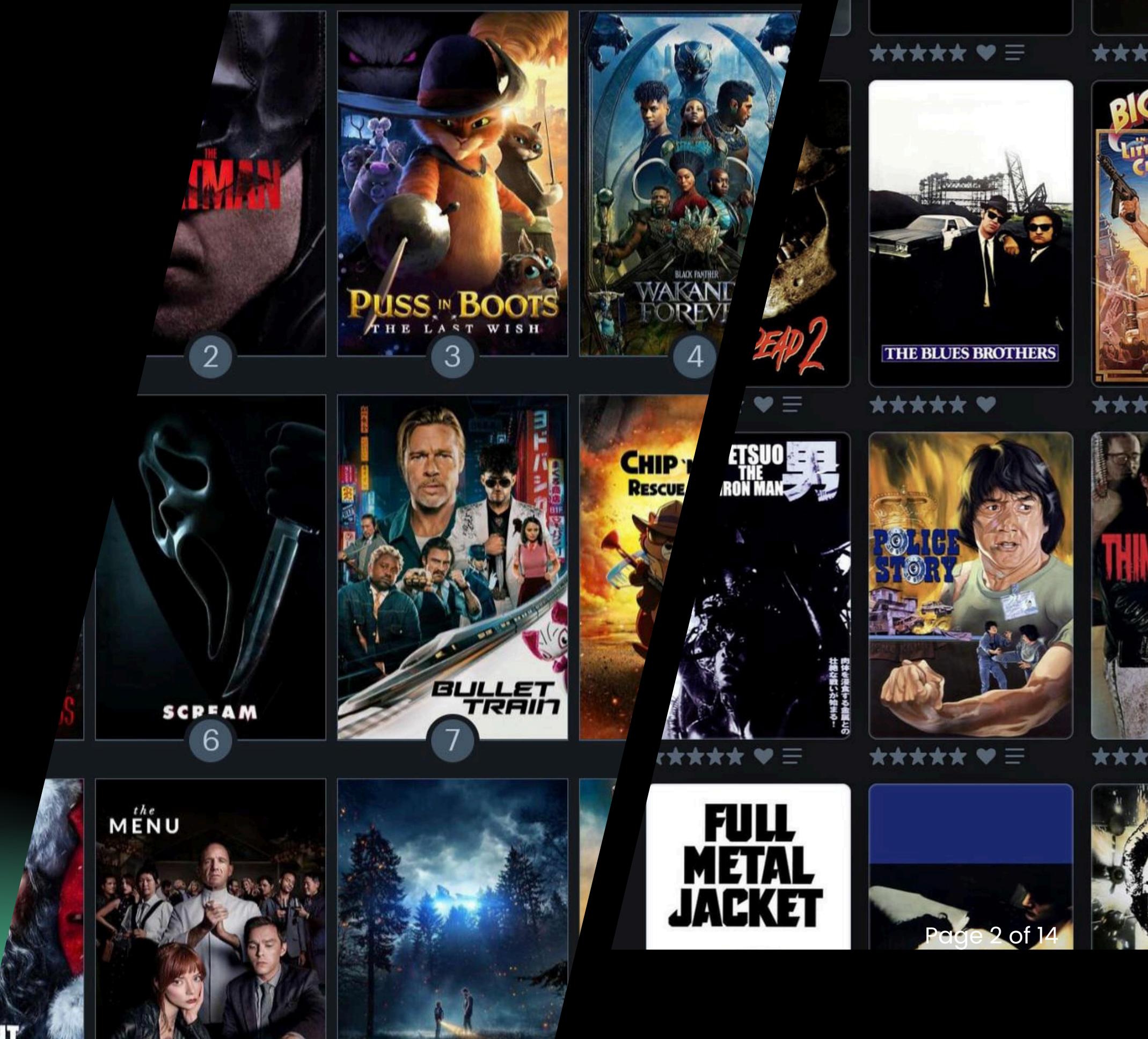
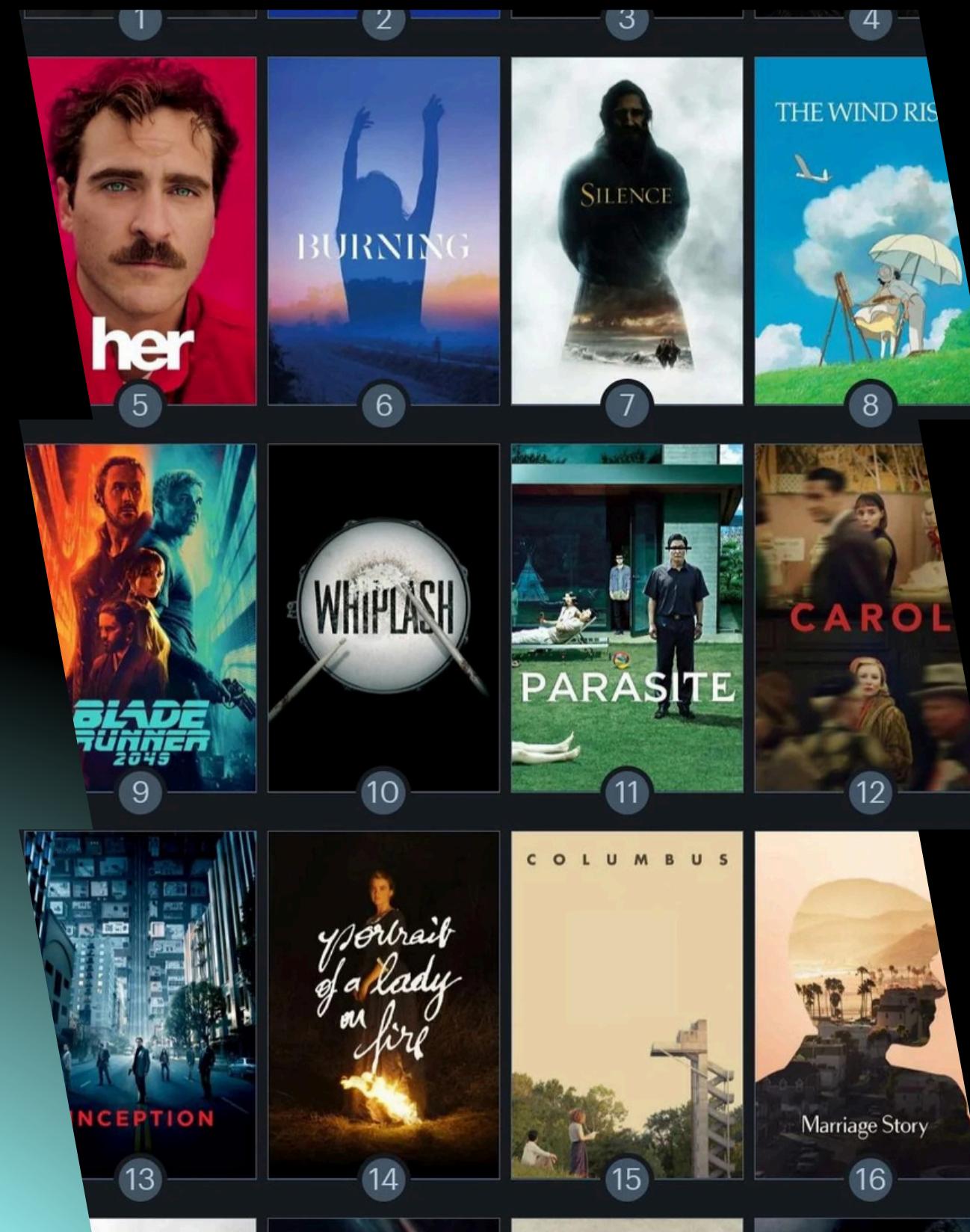


# Movie Recommendation System

# Introduction

This project aims to create a platform where users can receive personalized movie recommendations based on their preferences and past interactions. The system leverages machine learning techniques to provide tailored suggestions, employing Python libraries such as Pandas, NumPy, and Scikit-learn for data processing and model implementation. The results and visualizations will be presented in a Jupyter Notebook.





# Objectives

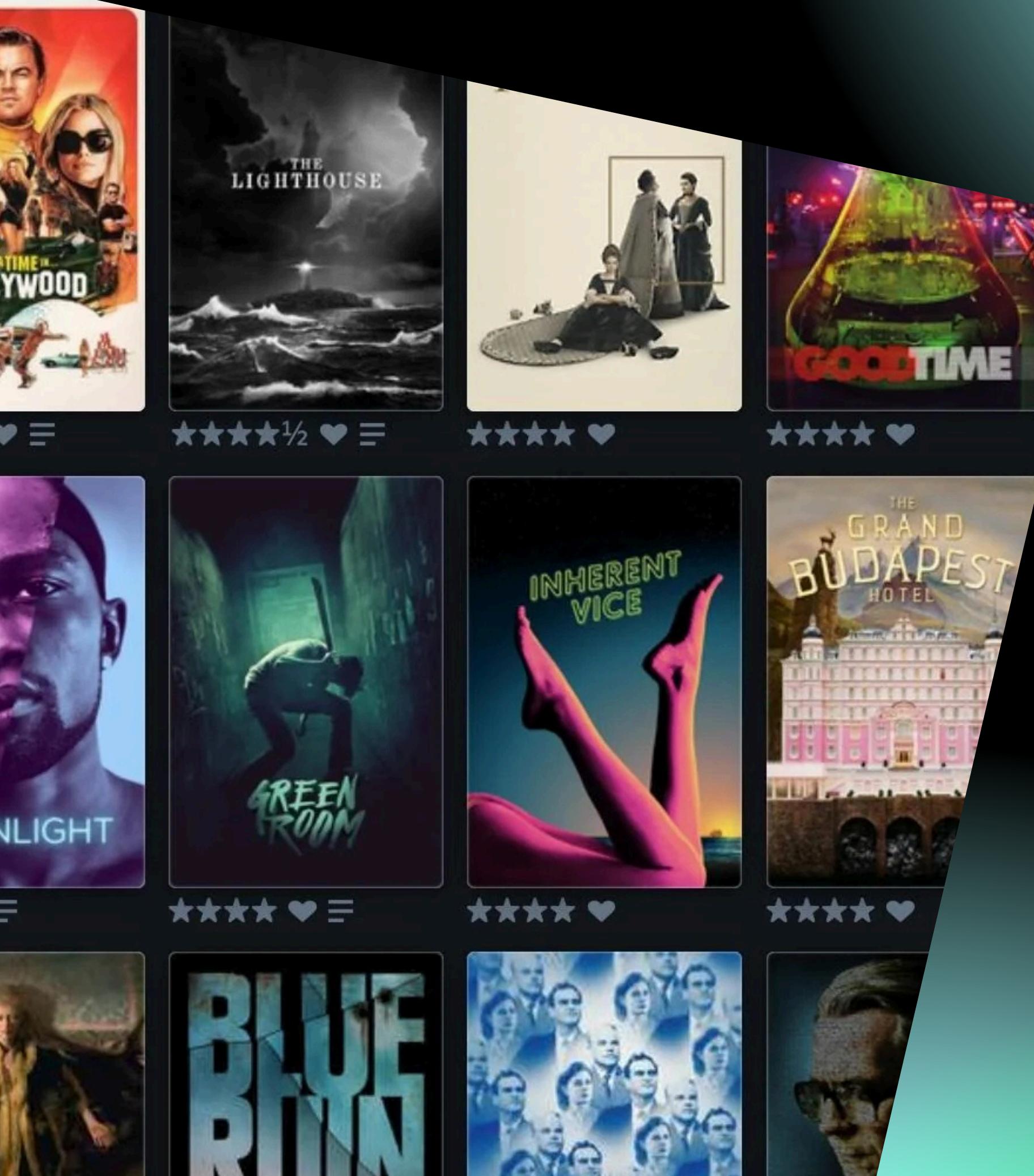
- 1- To develop a data-driven system for movie recommendations.
- 2- To implement machine learning algorithms for predicting genres, clustering similar movies, and predicting ratings.
- 3- To use an efficient and scalable approach for managing and analyzing data.

# Used Technologies

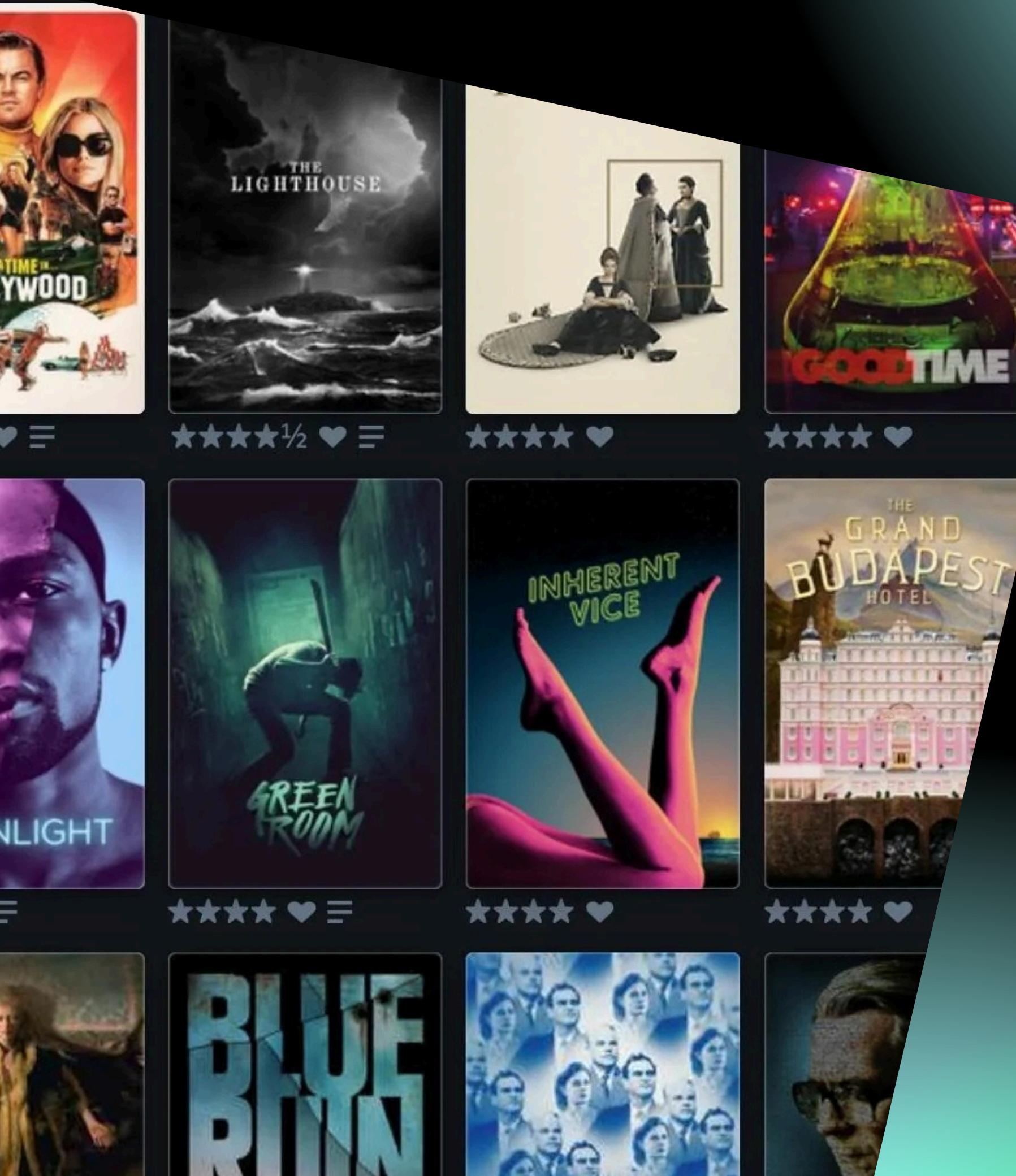
1. Data Analysis and Visualization: Pandas, NumPy, Matplotlib, Seaborn.
2. Machine Learning: Scikit-learn.
3. Development Environment: Jupyter Notebook.
4. Database: MongoDB for storing movie, user, and rating data.
5. Other Tools: Regular expressions (Regex) and Cosine Similarity for text and similarity analysis.



# System Architecture



- 1 **Data Layer** : Stores movie details, user profiles, and rating data
- 2 **Processing Layer** : Implements machine learning algorithms to generate recommendations.



# System Architecture

3

**Database Design :** by mongodb

**Users Dataset:** Contains user IDs and basic details.

**Movies Dataset:** Contains movie IDs, titles, genres, and metadata. Ratings

**Dataset:** Contains user IDs, movie IDs, and ratings.

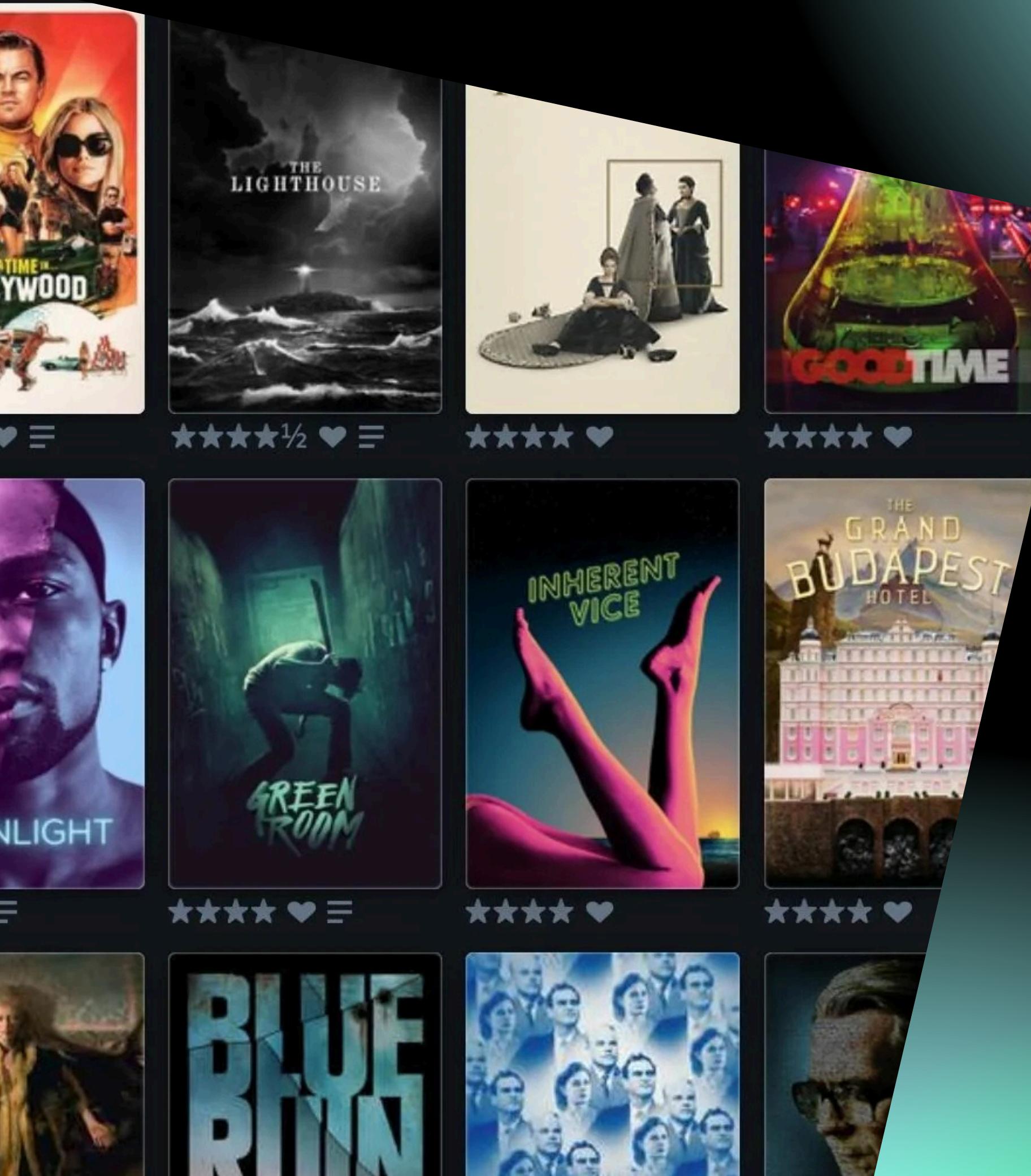
# Machine learning

## Classification

Goal: Predict movie genres a user might like based on their preferences.

Algorithms:

- 1- Naive Bayes for countries like France and the United Kingdom and for genres like Animation, Horror, Fantasy, and Crime.
- 2- Logistic Regression.
- 3- K-Nearest Neighbors (KNN).
- 4- Decision Tree for determining if movies are for adults or not.
- 5- Support Vector Machine (SVM) for checking countries like Japan and India and the genre Animation.

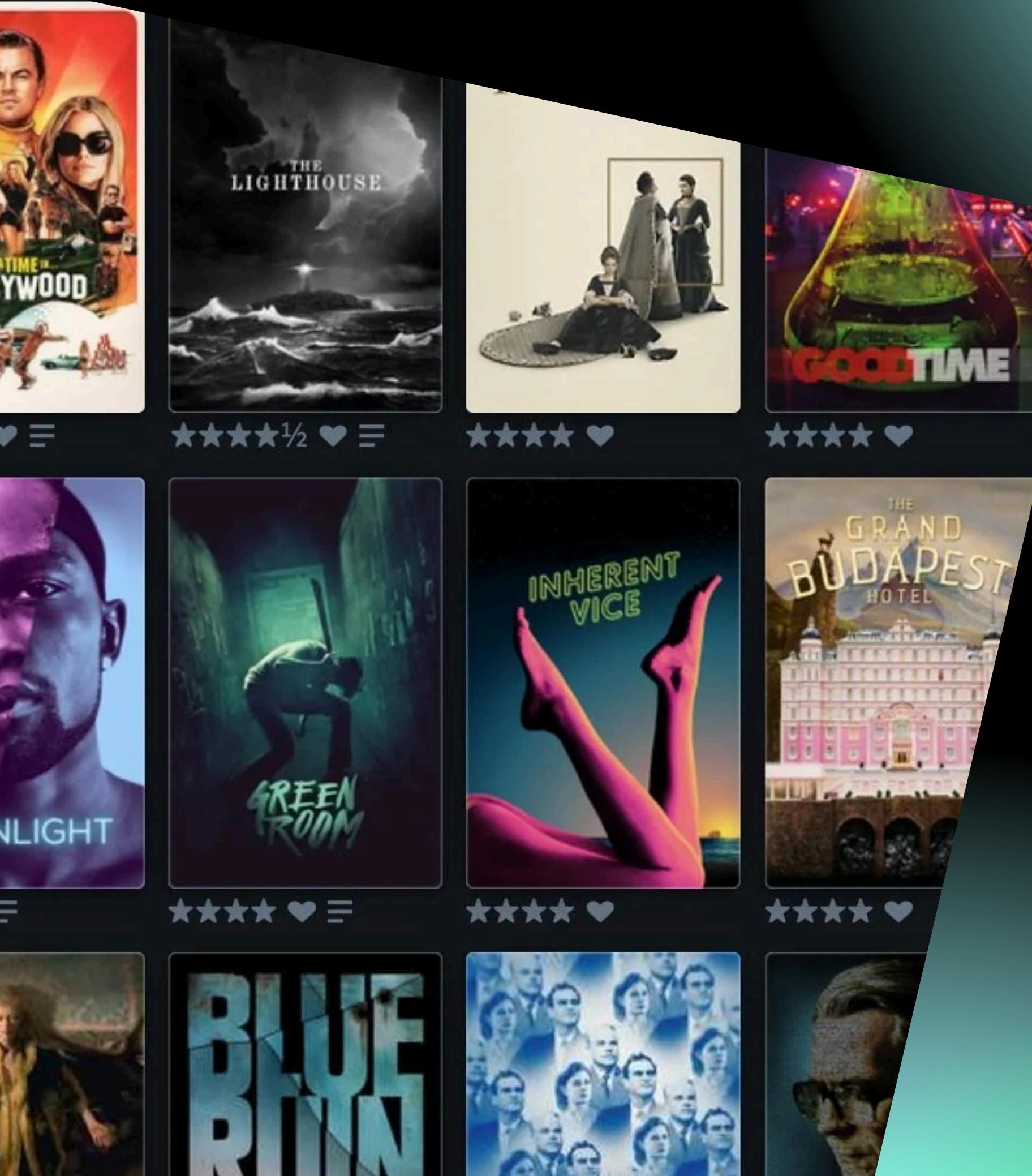


# Machine learning

## Clustering

**Goal:** Group movies with similar features (e.g., genre, release year, runtime).

**Algorithm:** K-Means Clustering.

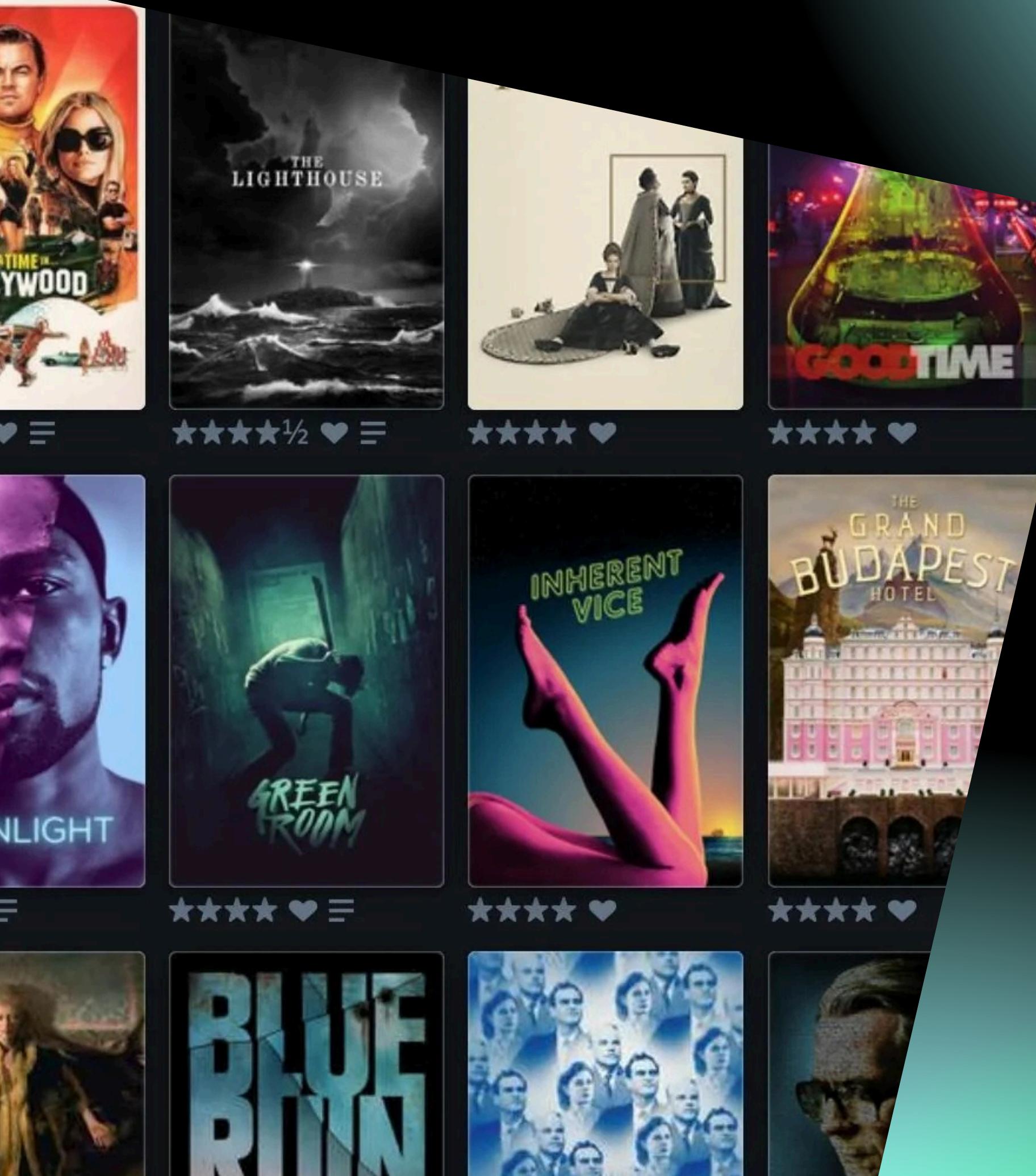


# Machine learning

## Regression

**Goal:** Predict movie ratings based on user input or past interactions.

**Algorithm:** Random Forest for predicting vote averages.



# Features

**Data Analysis:** Exploratory Data Analysis (EDA) to identify trends and insights.

**Recommendations:** Personalized movie suggestions based on machine learning predictions.

**Similarity Analysis:** Use Cosine Similarity to compare movies based on features or user preferences.

**Visualization:** Charts and plots for displaying clustering and prediction results.

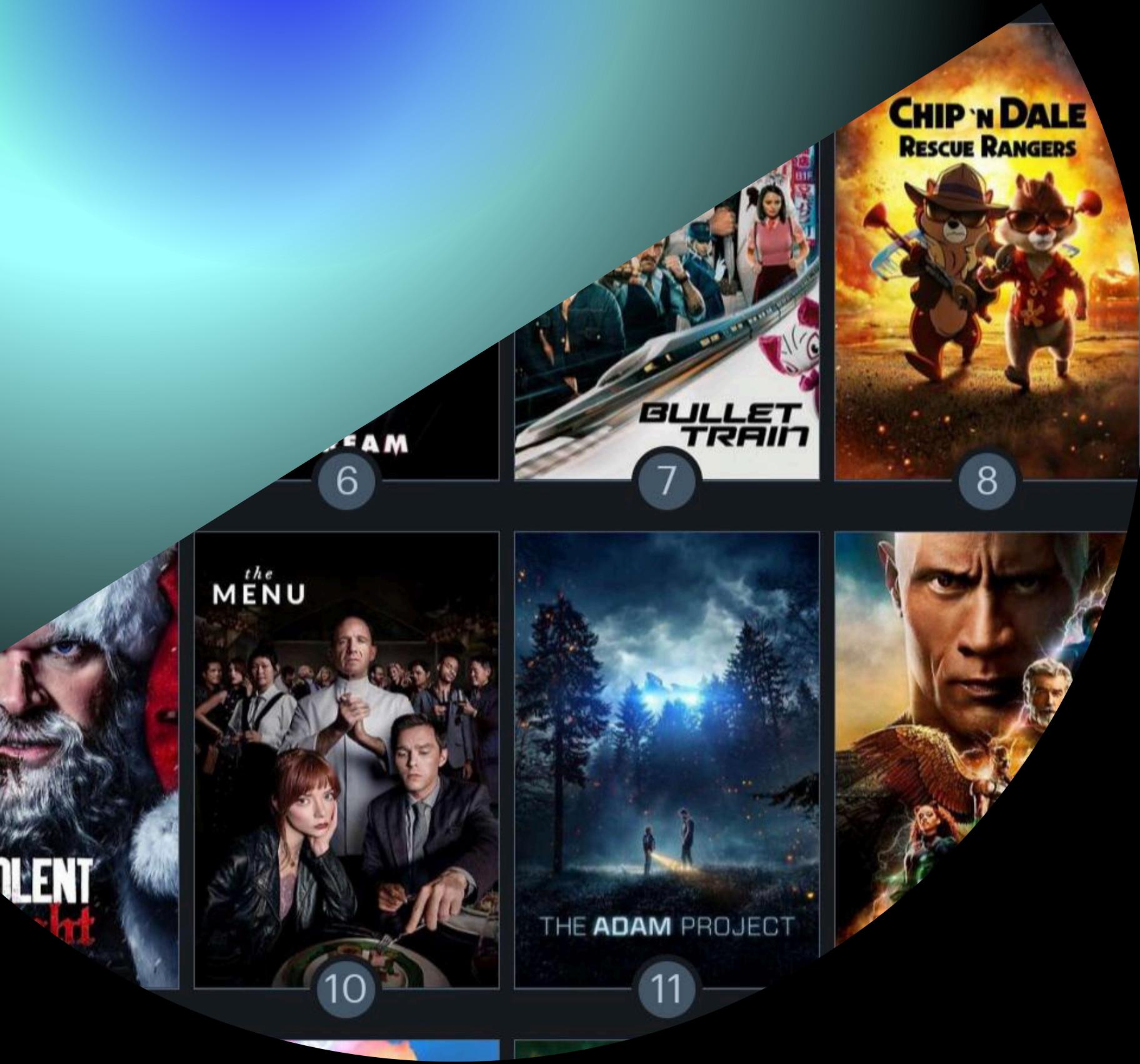


# Implementation Plan



1. Environment Setup: Install Jupyter Notebook and required libraries (Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn).
2. Data Preparation: Load and preprocess movie, user, and rating datasets.
3. Model Development: Train classification, clustering, and regression models using preprocessed data.
4. Similarity Analysis: Use Cosine Similarity and Regex to refine recommendations.
5. Evaluation: Test model accuracy using metrics like precision, recall, and RMSE.
6. Visualization: Create plots to display analysis and model performance.
7. Documentation: Present results and workflows in a well-structured Jupyter Notebook.

# Challenges and Solutions



## Data Preprocessing:

- Solution: Clean and normalize movie data during dataset preparation.

## Model Accuracy:

- Solution: Experiment with multiple algorithms and fine-tune hyperparameters.

## Data Visualization:

- Solution: Use Matplotlib and Seaborn to create insightful visualizations.

## Similarity Calculations:

- Solution: Apply Cosine Similarity for accurate recommendations and test Regex for efficient text handling.

# Conclusion

The Movie Recommendation System is a data-driven project that integrates machine learning techniques for personalized movie suggestions. By utilizing Python, Jupyter Notebook, MongoDB, Cosine Similarity, and Regex, this project provides a robust platform for analyzing data and delivering insightful recommendations.

# Thank You

