

Problem Definition:

Project Title: *IMDb Movie Score Prediction*

DatasetLink: <https://www.kaggle.com/datasets/luisortor/netflix-original-films-imdb-scores>

Problem Statement:

The objective of this project is to develop a machine learning model that predicts IMDb scores of movies available on Films based on features such as genre, premiere date, runtime, and language. The primary goal is to create a model that accurately estimates the popularity of movies, helping users discover highly rated films that match their preferences.

Project Goals:

Prediction Accuracy: Develop a model that can predict IMDb scores with a high degree of accuracy.

User-Friendly Interface: Create a user-friendly interface for users to input movie features and receive predicted IMDb scores.

Discoverability: Enable users to discover highly rated movies that align with their preferences.

Data Visualization: Provide visualizations to help users explore trends in movie scores based on different features.

Model Explainability: Ensure the model's predictions are interpretable and explainable to users.

Stakeholders:

Users: Movie enthusiasts looking for highly rated films.

Films Platform: The platform hosting the movie data and predictions.

Data Scientists and Engineers: Responsible for model development and deployment.

Design Thinking Steps:

Step 1: Empathize

User Research: Conduct surveys or interviews to understand what users expect from a movie rating prediction system.

Platform Analysis: Analyze the Films platform to determine data availability and user interaction points.

Stakeholder Input: Gather input from data scientists and engineers regarding technical requirements and constraints.

Step 2: Define

Problem Statement: Clearly define the problem of predicting IMDb scores for movies.

Project Goals: Document the specific goals and objectives of the project.

Stakeholder Roles: Define the roles and responsibilities of each stakeholder in the project.

Step 3: Ideate

Data Collection: Identify sources for movie data, including IMDb, genre databases, and language information.

Feature Selection: Brainstorm and select relevant features such as genre, premiere date, runtime, and language.

Model Selection: Use supervised machine learning algorithms such as linear regression.

UI Design: Sketch or describe the user interface for inputting movie features and displaying predictions.

Step 4: Prototype

Data Preprocessing: Develop scripts to clean and preprocess the movie data, handling missing values and outliers.

Feature Engineering: Create derived features or transform existing ones to enhance model performance.

Model Development: Build and test machine learning models using a subset of the data.

UI Prototype: Develop a basic prototype of the user interface for data input and prediction display.

Step 5: Test

Model Evaluation: Evaluate model performance using appropriate metrics such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE).

User Testing: Conduct usability testing with potential users to gather feedback on the UI prototype.

Iterate: Refine the model and UI based on feedback and testing results.

Step 6: Implement

Full-Scale Model Training: Train the machine learning model using the entire dataset.

UI Development: Build the complete user interface with all required features.

Integration: Connect the UI to the model for real-time predictions.

Step 7: Evaluate

Model Performance: Assess the final model's accuracy and interpretability.

User Satisfaction: Gather user feedback on the fully implemented system.

Platform Integration: Ensure seamless integration with the Films platform.

Step 8: Iterate

Continuous Improvement: Based on feedback and usage data, continue to enhance the model and user interface.

Scaling: Plan for scalability as the number of users and movies on the platform grows.

Maintenance: Establish a maintenance plan to address issues and update the system as needed.