Phase-1 Submission Template

Student Name: Vishanth V

Register Number: 410723106036

Institution: Dhanalakshmi College of engineering

Department: Electronics and Communication Engineering

Date of Submission: 28/04/2025

1.Problem Statement

Delivering Personalized Movie Recommendations with an AI-Driven Matchmaking System

In today's digital world, users face an overwhelming number of movie choices across streaming platforms. Searching for movies that match individual preferences often leads to frustration and wasted time. Standard recommendation systems offer limited personalization, failing to account for unique tastes such as preferred genres, actors, or themes. This project aims to develop an AI-driven matchmaking system to deliver highly personalized movie recommendations. Solving this problem enhances user satisfaction, increases platform engagement, and improves overall content discovery efficiency.

2.Objectives of the Project

- To develop a personalized movie recommendation system using AI and machine learning techniques.

- To predict and suggest movies that align with individual user preferences and behaviors.

- To uncover insights into user interests based on viewing history and interactions.
- To design a recommendation model that improves over time with user feedback.
- To present recommendations through an intuitive interface or dashboard.

3.Scope of the Project

- User profiling based on historical ratings and preferences.
- Movie profiling using metadata like genre, cast, and keywords.
- Implementation of collaborative filtering, content-based filtering, and hybrid recommendation techniques.
- Visualization of system performance and recommendations.

Limitations/Constraints:

- Dataset limited to popular English-language/global movies.
- Static datasets (no real-time data integration).
- Basic deployment without full-scale production deployment.
- Model performance constrained by dataset size and quality.

4.Data Sources

Primary Dataset: MovieLens Dataset (available publicly at GroupLens).

Dataset Type: Public, static dataset (downloaded once).

Supplementary Sources (optional): TMDB (The Movie Database) API for additional metadata like genres, ratings, or cast information.

Data Set Link:
https://www.kaggle.com/datasets/dev0914sharma/dataset/code

5.High-Level Methodology

- **Data Collection** – Download the MovieLens dataset from the GroupLens website.
- Optionally retrieve enriched metadata through APIs like TMDB.
- **Data Cleaning** – Handle missing ratings and duplicate records.
- Standardize inconsistent formats (e.g., genre tags, date fields).
- Normalize rating scales if required.
- **Exploratory Data Analysis (EDA)** – Visualize rating distributions, user activity, and popular genres.
- Identify trends in movie preferences and user behavior patterns.
- Use histograms, bar charts, heatmaps, and genre clouds.

- **Feature Engineering** – Develop user profiles based on viewing patterns and ratings.
- Create content-based features for movies (genres, directors, keywords).
- Encode categorical features using methods like one-hot encoding.
- **Model Building** –Experiment with:
- Content-Based Filtering (based on movie attributes).
- Collaborative Filtering (based on user behavior similarities).
- Matrix Factorization techniques (e.g., SVD).
- **Model Evaluation** –Metrics to be used:
- Root Mean Square Error (RMSE) for rating predictions.
- Precision@K and Recall@K for Top-N recommendation evaluation.
- Mean Average Precision (MAP) for ranking quality.
- **Visualization & Interpretation –** Present personalized movie recommendations.
- Display performance metrics and analysis results through visualizations.
- Use interactive dashboards or plots for clarity.
- **Deployment** – Build a lightweight web app or dashboard using Streamlit or Flask.

- Enable users to input preferences and receive real-time movie recommendations.

6.Tools and Technologies

- **Programming Language** – Python
- **Notebook/IDE** – Google Colab, Jupyter Notebook, Visual Studio Code (VS Code)
- **Data Processing Libraries** – pandas, numpy
- **Visualizations Libraries –** matplotlib, seaborn, plotly
- **Machi**ne Learning Libraries – scikit-learn, Surprise library
- **Deep Learning Libraries –** TensorFlow, PyTorch
- **Optional Tools for Deployment** – Streamlit, Flask, Gradio

7.Team Members and Roles

NAME	ROLES	RESPONSIBILITY
VISHANTH V	LEADER	Data collection, Data Cleaning, Exploratory Data Analysis, Feature Engineering, Visualization & Interpretation
VINOTH V	MEMEBER	Model building, Model Evaluation
SANTHKUMAR C	MEMBER	Tools and Technologies