

Phase-1 Submission

Student Name: Bharathkumar L

Register Number: 410723104008

Institution: Dhanalakshmi college of engineering

Department: CSE

Date of Submission: 28-4-2025

1.Problem Statement

An **AI-driven movie matchmaking system** enhances personalization by analyzing user emotions, viewing habits, and social influences. **Machine learning models** like collaborative filtering and NLP refine recommendations dynamically. Sentiment analysis ensures films match mood and personality traits. Privacy and ethical considerations help maintain transparency and trust in recommendations.

2.Objectives of the Project

The system uses AI to **personalize movie recommendations** based on user preferences and behaviors.

It **matches users with relevant films**, continuously refining suggestions for better accuracy and engagement

3.Scope of the Project

The **scope** of this project involves designing an AI-driven system that delivers personalized movie recommendations based on user behavior and preferences.:

- **User Data Analysis:** Collecting and analyzing viewing history, ratings, and interactions to understand preferences.

- **Recommendation Algorithms:** Implementing machine learning models like collaborative filtering, deep learning, or hybrid approaches.
- **Dynamic Adaptation:** Refining suggestions over time based on user feedback to improve accuracy.
- **User Interface & Experience:** Designing a seamless platform for users to explore recommendations effortlessly.
- **Scalability & Performance:** Ensuring the system can handle large datasets and evolving user demands.

4.Data Sources

- **User Data:** Viewing history, ratings, preferences, interactions
- **Movie Metadata:** Title, genre, director, cast, synopsis, ratings
- **User Interaction Data:** Likes, dislikes, search queries, shares
- **Contextual Data:** Time of viewing, location-based preferences, device type

Source (Kaggle), and it is public dataset and it is a dynamic dataset.

5.High-Level Methodology

1. Data Collection

- Gather movie metadata from public sources (e.g., IMDb, TMDb) via **API access**.
- Scrape reviews and ratings from websites (ensuring ethical scraping practices).
- Collect **user interaction data** from streaming platforms, if available.
- Generate synthetic data for missing user preferences.

2. Data Cleaning

- Handle missing values using imputation techniques.

- Remove duplicate entries to ensure data integrity.
- Normalize inconsistent formats (e.g., genre classification, rating scales).
- Detect and remove outliers in user preferences.

3. Exploratory Data Analysis (EDA)

- Use **visualizations** (histograms, scatter plots, heatmaps) to explore trends.
- Identify **correlations** between user preferences and movie features.
- Perform **clustering** to group similar users or movies.

4. Feature Engineering

- Extract **text-based features** from movie descriptions using NLP techniques.
- Generate new variables, like **watch-time frequency** or **genre affinity scores**.
- Perform **dimensionality reduction** if needed for optimization.

5. Model Building

- Experiment with **collaborative filtering** (user-based and item-based).
- Use **content-based models** (TF-IDF, word embeddings) for personalized matches.
- Explore **deep learning approaches** (neural networks, autoencoders).
- Try **hybrid models** to combine multiple techniques.

6. Model Evaluation

- Measure performance using **precision, recall, F1-score, and RMSE**.
- Apply **cross-validation** to ensure robustness.

- Evaluate user satisfaction through **A/B testing**.

7. Visualization & Interpretation

- Present insights through **interactive dashboards**.
- Use **heatmaps and bar charts** to visualize movie recommendations.
- Generate **explainable AI metrics** to interpret model decisions.

8. Deployment

- Deploy as a web application using Flask/Django.
- Integrate with streaming platforms via API connections.
- Allow continuous improvement via feedback loops.

6.Tools and Technologies

- **Programming Language** – (Python).
- **Notebook/IDE** – (Google Collab, Jupyter Notebook,).
- **Libraries** – (pandas, numpy, seaborn, matplotlib, scikit-learn).
- **Optional Tools for Deployment** – (Gradio, FastAPI).]

7.Team Members and Roles

S no	Name	Role	Responsibility
1	Bharath M	Leader	Project Manager
2	Abinеш G	Member	Data Preparation
3	Bharath Kumar L	Member	Data Visualization
4	Monish M	Member	Data Cleaning
5	Harish P	Member	Data Modeling