Project Report: Personalized Movie Recommendation System

Section 1: Introduction

The goal of this project is to develop a personalized movie recommendation system using collaborative filtering techniques. The primary objective is to help users discover movies they are likely to enjoy based on their previous ratings and preferences, thereby enhancing the user experience on streaming or movie platforms. The motivation behind this project is to explore how data science can drive engagement by providing customized content, and to practically implement machine learning models using Python and Power BI for visualization.

We address the key question: *"Given a user's past behavior, which movies are most likely to be enjoyed by the user in the future?"* The final product is a data-driven dashboard in Power BI that displays movie recommendations, KPIs, and visual insights.

Section 2: Data and Key Variables

The dataset used for this project is a combination of movie ratings and metadata:

- **Merged Dataset (merged_df)**: Created by combining a ratings CSV file (with userId, movieId, rating, timestamp) and a movies CSV file (with movieId, title, genres).
- **Key Variables:**
    - userId: Unique identifier for each user.
    - title: Movie title.
    - rating: Rating given by a user to a movie (typically on a 1 to 5 scale).

The final prediction file (predicted_long) includes:

- userId
- title
- predicted_rating: A predicted value normalized to a 1-5 scale using cosine similarity.

Section 3: Exploratory Data Analysis (EDA)

Before modeling, several analyses were performed:

- **Rating Distribution:** Examined how ratings are distributed across movies.
- **User Activity:** Number of ratings per user to identify active vs. passive users.
- **Movie Popularity:** Most and least rated movies.
- **Missing Values:** Filled NaNs with 0 in the user-movie matrix, assuming unrated.

These EDA steps helped identify data sparsity and informed decisions for modeling.

Section 4: Modeling Approach

A collaborative filtering method based on user-based similarity was used:

- **Step 1: User-Movie Matrix Creation**

  user_matrix = merged_df.pivot(index='userId', columns='title', values='rating').fillna(0)

- **Step 2: Cosine Similarity Calculation**

  user_sim = cosine_similarity(user_matrix)

- **Step 3: Predict Ratings**

  predicted_ratings = user_sim.dot(user_matrix) / np.abs(user_sim).sum(axis=1, keepdims=True)

- **Step 4: Normalization to 1-5 scale**

  predicted_long['predicted_rating'] = 1 + 4 * (predicted_long['predicted_rating'] - min_val) / (max_val - min_val)

- **Step 5: Export to CSV** for use in Power BI

Section 5: Discussion

From the predicted rating matrix, we generated recommendations for each user. This data was visualized in Power BI using:

- **Table of Top 10 Recommended Movies** per user

- **KPIs:**
    - Total Unique Users
    - Total Movies Rated
    - Total Movies
    - Maximum Rating
    - Average Predicted Rating
- **Bar Charts & Filters:**
    - Top 10 Movies by Rating
    - Count of Ratings per Month(From 1997 to 1998)
    - Most Popular Movies (Top Rated Movies)
    - Slicer for selecting a user to recommend his top movies

The Power BI dashboard made it easy to interpret the results, enabling dynamic exploration of personalized recommendations.

Section 6: Limitations

- **Cold Start Problem:** Users or movies with few ratings may result in poor predictions.
- **Assumption of Similarity:** Cosine similarity assumes similar users have similar preferences, which may not always be true.
- **Data Sparsity:** The matrix is sparse; most users have rated only a few movies.
- **Predictions Not Grounded in Content:** No movie genre, director, or actor info was used.

Section 7: Conclusion

This project successfully demonstrates a collaborative filtering-based movie recommendation system that generates personalized suggestions using user similarity. The results were deployed and visualized in Power BI, making it practical for stakeholders to understand the recommendations. The combination of Python for modeling and Power BI for reporting created a strong end-to-end solution.

Section 8: Additional Work

Future improvements could include:

- **Item-based filtering** and hybrid models (content + collaborative)
- **Incorporating metadata** like genre, release year, and actors
- **Real-time recommendations** using streaming data
- **UI Development** to deploy the recommendations on a web platform
- **Matrix Factorization techniques** like SVD for better accuracy

This detailed report structure offers clarity on methodology, outcomes, and next steps for stakeholders and technical audiences alike.