Project Report: Movie Recommendation System

1. Problem Analysis

In this digital era, the number of movies available is often too much to comprehend, depriving users of the ability to search for movies that are specific to them. This issue is serious as it results in decision-making problems where the users spend more time looking for a film than watching one. This particular project aims to solve this problem through the creation of a Movie Recommendation System wherein the users can log in and provide relevant information which will be used to recommend movies to them to improve the overall user experience. The expected outcome is a straightforward application that recommends movies to its users based on what they have searched for, thus addressing the issue and improving the users' creativity.

2. Data Preparation & Analysis

The project is based on two datasets. The first dataset is the 'imdb\_top\_1000.csv' file which contains information regarding some of the best-rated movies, and the second dataset is the ‘movies.csv’ which contains more movies with corresponding ratings and genres. Data preparation in this instance included the acquisition of these datasets, which included assigning the right data types to each column and the unification of the two datasets using the movie titles as unifying items. Missing values were masked by increasing the value to zero for the relevant numerical entry columns for a complete dataset with no gaps.

3. Solution Design

The architecture of the Movie Recommendation System is designed around a user interface that allows for movie selection and displays recommendations. The system employs a Nearest Neighbors algorithm to find similar movies based on user-selected features, including IMDB ratings, Meta scores, and genre. The choice of the Nearest Neighbors algorithm is justified by its effectiveness in handling high-dimensional data and its ability to provide relevant recommendations based on similarity metrics. Additionally, fuzzy matching ensures that user input is matched accurately to movie titles, enhancing the robustness of the recommendation process.

4. Solution Implementation

The implementation of the Movie Recommendation System was executed using Python and the Tkinter library to create a graphical user interface (GUI). The code is structured to facilitate clear data flow from input selection to recommendation display. Each section of the code is well-documented, outlining the purpose and functionality of key components. The system is efficient, employing data structures such as sparse matrices to optimize memory usage when handling large datasets. Through rigorous testing, the application successfully provides recommendations based on user input, demonstrating its effectiveness in aiding movie selection. Overall, the project successfully meets its objectives, offering a valuable tool for movie enthusiasts looking to discover new films tailored to their tastes.

Dataset reference

<https://www.kaggle.com/datasets/harshitshankhdhar/imdb-dataset-of-top-1000-movies-and-tv-shows>

<https://www.kaggle.com/datasets/danielgrijalvas/movies>