Movie Recommendation System - Project Report

Abstract / Introduction

This mini-project implements a movie recommendation system using both content-based and collaborative filtering techniques. Recommendation systems are a subclass of information filtering systems that seek to predict the preference a user would give to an item. Here, the system is built to recommend movies based on the similarity of content and user inputs, leveraging natural language processing and machine learning libraries.

Objective

The main objective of this project is to build a recommendation engine that suggests movies to users. It aims to predict what movies a user might like based on the similarity to other movies or user preferences. The system focuses on improving the relevance of movie suggestions through text analysis and similarity measures.

Dataset Description

The dataset used in this project is based on the MovieLens dataset, which contains information such as:

- Movie titles
- Genres
- Metadata including keywords and descriptions
  This data is preprocessed to remove missing or irrelevant entries and converted into a usable format for vectorization and similarity computation.

Tools & Libraries Used

- Python 3.x
- Pandas
- NumPy
- Scikit-learn

- Difflib
- Jupyter Notebook

Methodology / Model

The recommendation engine is built using the following steps:

1. **Data Cleaning & Preprocessing:** Load movie metadata and clean it for inconsistencies or missing values.
2. **Feature Extraction:** Use `TfidfVectorizer` to convert text data (like genres and keywords) into numerical format.
3. **Similarity Computation:** Use `cosine_similarity` to measure how similar a movie is to another.
4. **Recommendation Generation:** Take user input, find the closest matching title, and suggest similar movies based on cosine similarity.

Conclusion

This project demonstrates the effectiveness of a movie recommendation system using content-based filtering methods. The model provides relevant suggestions based on movie similarity by leveraging textual features. Such systems can enhance user experience on entertainment platforms and can be further improved by including user ratings, behavioral data, or deep learning approaches.