Project Report: Movie Recommendation System

Developed by: Divyansh Srivastava

Introduction

With the growing number of movies released every year, choosing the right movie to watch has become increasingly challenging. A Movie Recommendation System helps users discover films tailored to their interests, reducing time spent searching and improving overall user experience. This project leverages machine learning techniques to suggest movies based on user preferences, offering a personalized experience in content consumption.

Abstract

This project focuses on building a Movie Recommendation System using collaborative and content-based filtering techniques. By analyzing user ratings and movie features (such as genres and popularity), the system predicts and recommends films similar to the ones a user likes. The interface is built using Streamlit, enabling users to interact with the recommendation engine seamlessly. The goal is to enhance user engagement and satisfaction by offering intelligent and relevant suggestions using machine learning techniques.

Tools Used

- Python: Programming language for backend logic and data processing.

- Pandas: For data cleaning, transformation, and manipulation.

- Scikit-learn: For implementing machine learning models.

- Streamlit: To develop an interactive and user-friendly web interface.

Steps Involved in Building the Project

1. Data Collection

   - Used the MovieLens dataset containing user ratings, movie titles, genres, and metadata.

2. Data Preprocessing

- Cleaned missing values, merged datasets, and encoded categorical features.

- Transformed the data into a suitable format for recommendation algorithms.

3. Model Development

  - Collaborative Filtering: Based on user-movie interaction, using cosine similarity on user rating vectors.

  - Content-Based Filtering: Used movie genres and metadata to recommend similar movies.

4. Recommendation Logic

   - When a user selects a movie, the system calculates similarities and suggests the top 5 most relevant movies.

  - Integrated both filtering methods for better accuracy and relevance.

5. User Interface with Streamlit

  - Built a dropdown to select a movie and a button to generate recommendations.

  - Displayed recommended movies in an intuitive and visually appealing manner.

Conclusion

The Movie Recommendation System developed successfully meets its objective of suggesting relevant movies based on user preferences. By combining machine learning techniques with an interactive UI, the system offers a personalized and enjoyable user experience. This project serves as a strong foundation for building more advanced recommendation engines, with potential extensions such as real-time user feedback and integration of sentiment analysis for enhanced accuracy.