Project Report: Hybrid Movie Recommendation System

Objective

The primary objective of this project is to develop a **Hybrid Movie Recommendation System** that combines the strengths of both **Content-Based Filtering** and **Collaborative Filtering** to provide more accurate and personalized movie suggestions to users.

Methodology

1. Dataset

- `movies.csv`: Contains movie metadata including titles and genres.

- `ratings.csv`: Contains user ratings for various movies.

2. Data Preprocessing

- Missing values in the `genres` column were handled by replacing them with empty strings.

- The `ratings.csv` file was split into training (80%) and testing (20%) sets using `train_test_split`.

Recommendation Techniques

Content-Based Filtering

- Implemented using **TF-IDF Vectorization** on the `genres` field.

- Cosine similarity is computed between movie vectors to find similar titles.

- Users receive recommendations based on content similarity to a selected movie.

Collaborative Filtering

- Built using **Truncated Singular Value Decomposition (SVD)** on the user-item rating matrix.

- The system predicts ratings for unseen movies by reconstructing the rating matrix.

- Recommendations are generated by identifying top-rated unseen movies for a user.

Hybrid Approach

- Combines both content-based similarity scores and collaborative predicted ratings.

- A **hybrid score** is calculated as the average of both methods, improving recommendation quality.

Evaluation Metrics

1. **Root Mean Square Error (RMSE)** and **Mean Absolute Error (MAE)** were calculated for:

   - Collaborative Filtering

   - Content-Based Filtering

   - Hybrid Model

2. **Top-N Recommendation Evaluation**:

   - Precision, Recall, and F1-Score were calculated for sample users to evaluate how effectively the system recommends high-rated movies.

User Interface

Developed using **Streamlit**, providing a user-friendly front-end with the following functionalities:

- **Content-Based Recommendation**: Based on a selected movie title.

- **Collaborative Recommendation**: Based on a given user ID.

- **Hybrid Recommendation**: Based on both a movie title and user ID.

- **Evaluation Panel**: Displays RMSE, MAE, Precision, Recall, and F1 metrics.

Challenges Faced

- Compatibility issues with Python packages on Streamlit Cloud were encountered.

- Errors occurred due to improper indexing using `get_loc` on a list instead of a pandas Series or Index.

- Dependency management was crucial to ensure smooth deployment.

Conclusion

The Hybrid Movie Recommendation System successfully integrates content and collaborative filtering to enhance recommendation accuracy. The evaluation results show that the **hybrid model outperforms** individual methods in most cases, providing a more balanced and personalized user experience.

Future Work

- Incorporate user demographics and movie metadata like actors/directors for deeper insights.

- Integrate deep learning models for dynamic feature extraction.

- Enable real-time recommendation updates based on recent user interactions.