Title: Movie Recommendation System Report

Recommendation systems play a crucial role in modern platforms by helping users discover content they'll likely enjoy. In the context of movies, recommending relevant films enhances user engagement and satisfaction.

This project explores three core approaches to movie recommendation:

User-based collaborative filtering – recommends movies based on users with similar preferences.

Item-based collaborative filtering – suggests movies similar to those the user liked.

Random-walk-based (Pixie-style) – uses a graph-based approach to find connections between users and movies.

2. Dataset Description

We used the MovieLens 100K dataset, which contains:

943 users, 1,682 movies, and 100,000 ratings (from 1 to 5 stars).

Features include user_id, movie_id, rating, and timestamp.

Preprocessing steps included:

Building user-movie interaction matrices.

Filtering out missing values or duplicates.

Converting the data into a suitable format for similarity and graph-based analysis.

3. Methodology

a. User-Based Collaborative Filtering

Constructs a user-item rating matrix.

Calculates cosine similarity between users.

Recommends movies that similar users liked but the target user hasn't rated.

b. Item-Based Collaborative Filtering

Uses a transposed version of the user-item matrix.

Computes cosine similarity between movies.

Suggests movies that are similar to those already rated highly by the user.

c. Random-Walk-Based (Pixie-Inspired) Algorithm

Constructs a bipartite graph of users and movies.

Performs random walks from a user node through the graph.

Tracks the most frequently visited movies as recommendations.

Captures indirect and more diverse relationships between users and content.

4. Implementation Details

Custom functions were written for each algorithm.

For the graph-based method, an adjacency list was built to represent user-to-movie connections.

The random walk simulates multiple steps from the user node, with weighted random selection to explore likely movies.

Walk frequencies were counted and ranked to determine recommendations.

5. Results and Evaluation

Each method generated top-5 movie recommendations. Key observations include:

User-based often recommends mainstream or commonly liked movies, especially when user overlap is high.

Item-based gives more consistent results and is less sensitive to user sparsity.

Random-walk-based tends to return more diverse and serendipitous recommendations, especially effective when graph connections are rich.

Limitations:

No formal accuracy metrics (e.g., RMSE or precision/recall) were computed.

Algorithms assume static preferences (no time dynamics).

Cold-start problem remains for new users or movies with few interactions.

6. Conclusion

This project successfully implemented three movie recommendation strategies using collaborative filtering and graph traversal.

Key takeaways:

User and item similarity are powerful tools for identifying preferences.

Graph-based approaches uncover deeper relationships in the data.

Each method has trade-offs in accuracy, diversity, and scalability.

Potential improvements:

Combine methods into a hybrid model.

Use additional features (genres, timestamps, tags).

Introduce evaluation metrics for quantitative comparison.

Real-world use cases include platforms like Netflix, YouTube, and Spotify where recommendation systems drive user engagement and satisfaction.