Movie Recommendation System Report - Issam Alzouby

1. Introduction

Movie recommendation systems help users discover new content based on their past preferences and the behavior of users similar to them. These systems are widely used in movie streaming platforms, online stores, and social networks. In this project, I implemented three major approaches: user-based collaborative filtering, item-based collaborative filtering, and a Pixie-inspired random-walk method. Each method offers a unique perspective on modeling user behavior and generating recommendations.

2. Dataset Description

We used the MovieLens 100K dataset, which contains:

- 943 users
- 1,682 movies
- 100,000 ratings

The dataset includes user demographics, movie titles and metadata, and timestamped ratings. We used the user_id, movie_id, and rating columns for collaborative filtering, and additional title metadata for interpreting results. Preprocessing included loading with correct delimiters, handling missing values, converting timestamps, and combining rating information with movie titles for graph construction. I also normalized user ratings before building the random-walk graph.

3. Methodology

User-Based Collaborative Filtering

I computed similarities between users by using cosine similarity on the user–movie rating matrix. For a target user, I find the most similar users, aggregate their ratings, and recommend movies the target user hasn't watched yet.

Item-Based Collaborative Filtering

For Item-Based CF, I compare movies rather than users. I took a transpose of the matrix so each row represents a movie, computed item–item similarities, and recommended movies most similar to the target movie's rating patterns.

Pixie-Inspired Random Walk Method

I built a bipartite graph where users connect to the movies they'd rated. A random walk begins from either a user or a movie and hops between neighbors. Where the movies that are most frequently visited during the walk become the recommendations. This method captures graph structure and performs well even with sparse data.

4. Implementation Details

I first created the user–movie matrix and computed similarities for both collaborative filtering methods. For the Pixie-inspired approach, I merged ratings with movie titles, averaged duplicates, and normalized ratings per user. Then I built a list where users and movies are nodes connected by rating edges. The random walk was implemented by selecting a starting node, randomly choosing neighbors at each step, counting movie visits, and ranking them by frequency. We removed movies already rated by the target user to avoid redundant recommendations.

5. Results and Evaluation

User-based and item-based methods produced respectable results, typically recommending movies within the same genre or with similar ratings. However, they usually struggle with a lack of data/input since most users rate only a small amount of movies. The Pixie-inspired random walk generated more diverse recommendations by expanding and capitalizing off the user–movie interactions. It also produced results quicker due to short walk lengths. It still has limitations though, which include how randomness affects consistency and occasional drift into less relevant neighborhoods.

```
similarity matrix shape: (943, 943)
sample similarity scores:
user_id         1         2         3         4         5
user_id
1        1.000000  0.166931  0.047460  0.064358  0.378475
2        0.166931  1.000000  0.110591  0.178121  0.072979
3        0.047460  0.110591  1.000000  0.344151  0.021245
4        0.064358  0.178121  0.344151  1.000000  0.031804
5        0.378475  0.072979  0.021245  0.031804  1.000000

==================================================
Recommendations for user 10:
==================================================
```

	Movie Name
Ranking	
1	Stalingrad (1993)
2	Horseman on the Roof, The (Hussard sur le toit...
3	Steel (1997)
4	Bed of Roses (1996)
5	Baby-Sitters Club, The (1995)

```
Item similarity matrix shape: (1682, 1682)

Sample similarities for first 5 movies:
movie_id        1         2         3         4         5
movie_id
1        1.000000  0.402382  0.330245  0.454938  0.286714
2        0.402382  1.000000  0.273069  0.502571  0.318836
3        0.330245  0.273069  1.000000  0.324866  0.212957
4        0.454938  0.502571  0.324866  1.000000  0.334239
5        0.286714  0.318836  0.212957  0.334239  1.000000

==================================================
Movies similar to Jurassic Park (1993):
==================================================
```

	Movie Name
Ranking	
1	Top Gun (1986)
2	Speed (1994)
3	Raiders of the Lost Ark (1981)
4	Empire Strikes Back, The (1980)
5	Indiana Jones and the Last Crusade (1989)

```
==================================================
Recommendations for user 1:
==================================================
                                    Movie Name
Ranking
1                         River Wild, The (1994)
2            Man Who Knew Too Little, The (1997)
3                     Arsenic and Old Lace (1944)

==================================================
Movies similar to Jurassic Park (1993):
==================================================
                                              Movie Name
Ranking
1                             Return of the Jedi (1983)
2                                     Braveheart (1995)
3                                         Brazil (1985)
4                             Cyrano de Bergerac (1990)
5        Old Lady Who Walked in the Sea, The (Vieille q...
```

6. Conclusion

This project presents three effective recommendation strategies. Collaborative filtering provides strong results when user ratings overlap sufficiently, while the Pixie-inspired method introduces a more scalable, graph-aware approach that captures broader connectivity. Future improvements could include hybrid models, incorporating movie genres, weighting edges based on rating magnitude, or running multiple walks to stabilize output. These methods have real-world applications in platforms like Netflix, Amazon, Pinterest, and YouTube.