Collaborator-based Movie Recommendation System

Project of CSE 523: Machine Learning

Jainam Shah
School of Engineering and Applied Science,
Ahmedabad, Gujarat,

AU2040186
jainam.s4@ahduni.edu.in

Akshay Parmar
School of Engineering and Applied Science,
Ahmedabad, Gujarat,

AU2040199
akshay.p@ahduni.edu.in

Shubham Bhutt
School of Engineering and Applied Science,
Ahmedabad, Gujarat,

AU2040206
shubham.b1@ahduni.edu.in

Yash Chotaliya
School of Engineering and Applied Science,
Ahmedabad, Gujarat,

AU2040193
yashkumar.c@ahduni.edu.in

***Abstract*— Collaborator-based movie recommendation systems rely on the ratings and reviews of other users with similar tastes to generate recommendations for a given user. In this system, the user's preference is inferred based on the ratings of other users with similar profiles. The system then recommends movies that similar users have enjoyed but that the given user has not yet seen. Collaborator-based recommendation systems can be implemented using different algorithms, such as neighborhood-based and matrix factorization techniques. E-commerce platforms, social networks, and streaming services have widely adopted these systems. They are expected to continue to play a significant role in improving the accuracy and relevance of movie recommendations.**

Keywords—Collaborator, Cosine Similarity, User to User

I. Introduction

In recent years, the increasing availability of data and advances in machine learning techniques have led to the development of highly accurate and effective movie recommendation systems. One popular approach to movie recommendation is collaborative filtering, where a user's preferences are inferred based on the preferences of other users with similar tastes. Collaborative-based movie recommendation systems leverage this approach to provide personalized recommendations to users based on their viewing history, ratings, and reviews. Machine learning algorithms play a critical role in implementing these recommendation systems, enabling the analysis of large datasets and the identification of hidden patterns in user behavior. In this report, we will explore the different techniques used in collaborator-based movie recommendation systems, including neighborhood-based and matrix factorization algorithms, and evaluate their performance in terms of accuracy and scalability. We will also discuss the challenges and limitations of these systems and potential avenues for future research.

II. Literature Survey

Several studies have explored using machine learning algorithms in collaborator-based movie recommendation systems. One of the most widely used approaches is the neighborhood-based algorithm, which identifies the most similar users or items based on their ratings and preferences. The k-nearest neighbors (k-NN) algorithm is a common implementation of this approach, which recommends movies based on the ratings of the k most similar users. The performance of this algorithm can be improved by using more advanced techniques, such as weighted k-NN and clustering-based methods.

Another popular approach to collaborative filtering is matrix factorization, which decomposes the user-item rating matrix into two lower-dimensional matrices representing user and item features. This approach has been shown to be highly effective in capturing latent features and making accurate recommendations. Singular value decomposition (SVD) and its variants, such as probabilistic matrix factorization (PMF) and non-negative matrix factorization (NMF), are commonly used algorithms in this category.

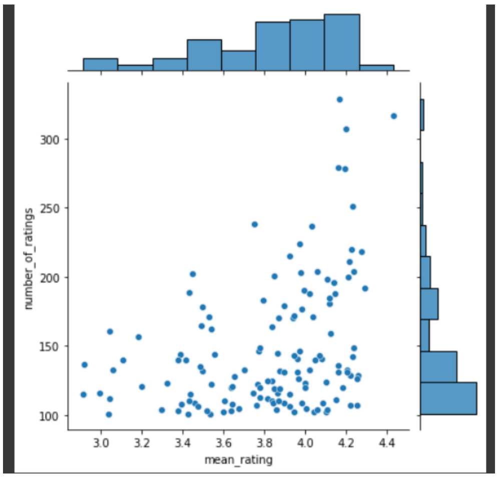
Recent studies have also explored deep learning techniques in collaborator-based recommendation systems, such as neural networks and autoencoders. These approaches have shown promising results in capturing complex patterns and interactions among users and items and handling sparse and noisy data.

III. Implementation

The system uses collaborative filtering, which is a type of recommendation system that identifies similar users and recommends movies that these similar users have rated highly. Still, the current user has yet to see it. The system follows the following steps:

1. Import Python libraries for data processing, visualization, and similarity computation.
2. Download and read data using the Pandas library.

3. Conduct exploratory data analysis (EDA) to understand the data, including aggregating ratings by movie and visualizing the distribution of ratings.
4. Create a user-movie matrix to represent users' ratings of movies.



5. Normalize the user-movie matrix to center the ratings around zero and improve the accuracy of similarity measures.
6. Identify similar users based on their ratings of movies using either Pearson correlation or cosine similarity.
7. Narrow down the pool of movies to recommend by selecting movies that similar users have rated higher, but the current user has yet to see.
8. Recommend movies to the current user based on the average ratings of these movies among similar users.

We used the above steps to implement the program to recommend the user.

IV. Result

We try to recommend the movie for a particular mentioned user in a code, we recommended the other ten movies based on the similar user with similar ratings.

	movie	movie_score
16	Harry Potter and the Chamber of Secrets (2002)	1.888889
13	Eternal Sunshine of the Spotless Mind (2004)	1.888889
6	Bourne Identity, The (2002)	0.888889
29	Ocean's Eleven (2001)	0.888889
18	Inception (2010)	0.587491
3	Beautiful Mind, A (2001)	0.466667
5	Blade Runner (1982)	0.466667
12	Donnie Darko (2001)	0.466667
10	Departed, The (2006)	0.256727
31	Shawshank Redemption, The (1994)	0.222566

From the data, we can see that user one can watch “harry potter and the Chamber of Secrets.” The movie score for that user for this movie is very high than other movies. So, it is more likely that user one will like the movie.

Further, we can create the 20% test and 80% train data set and analyze the model using the RMSE for the performance of the Movie recommendation system.

V. Conclusion

We have developed a movie recommendation system using a collaborative filtering technique. The plan recommends movies to the user based on the ratings given by other similar users. We have used the Pearson correlation and cosine similarity to calculate user similarity. We have also filtered the movies based on the number of ratings they received and kept the movies with over 100 ratings. Finally, we have used the item scores to recommend top movies to the user.

References

[1] Wu, C. S. M., Garg, D., & Bhandary, U. (2018, November). Movie recommendation system using collaborative filtering. In 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS) (pp. 11-15). IEEE.

[2] Lavanya, R., & Bharathi, B. (2021). Movie Recommendation System to Solve Data Sparsity Using Collaborative Filtering Approach. Transactions on Asian and Low-Resource Language Information Processing, 20(5), 1-14.

[3] Geetha, G., Safa, M., Fancy, C., & Saranya, D. (2018, April). A hybrid approach using collaborative filtering and content based filtering for the recommender system. In Journal of Physics: Conference Series (Vol. 1000, No. 1, p. 012101). IOP Publishing.

[4] Anwar, T., & Uma, V. (2021). Comparative study of recommender system approaches and movie recommendation using collaborative filtering. International Journal of System Assurance Engineering and Management, 12, 426-436.