IJIREEICE

ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ∺ Peer-reviewed & Refereed journal ∺ Vol. 13, Issue 11, November 2025

DOI: 10.17148/IJIREEICE.2025.131116

Comparative Analysis of User-Based and Item-Based Collaborative Filtering Using the MovieLens Dataset

M. V. Karthikeya[1], Vishnu Vardhan[2], D.Nanda Kishore[3], Subhrajit Panda[4], Sai Tejas[5], A. Narendrasai[6], B Sharath Reddy[7], Dr. M. ULAGAMMAI[8]

Scholars, Department of CSE, SRM Institute of Science and Technology, Chennai, India[1]

Scholars, Department of CSE, SRM Institute of Science and Technology, Chennai, India[2]

Scholars, Department of CSE, SRM Institute of Science and Technology, Chennai, India[3]

Scholars, Department of CSE, SRM Institute of Science and Technology, Chennai, India[4]

Scholars, Department of CSE, SRM Institute of Science and Technology, Chennai, India[5]

Scholars, Department of CSE, SRM Institute of Science and Technology, Chennai, India[6]

Scholars, Department of CSE, SRM Institute of Science and Technology, Chennai, India[7]

Associate Professor, Department of CSE, SRM Institute of Science and Technology, Chennai, India[8]

Abstract: In today's Now, recommender systems in the present time become an essential tool to screen out relevant content for users. This paper constitutes a comparative study of user-based and item-based collaborative filtering (CF), using MovieLens dataset. Methodology, the details of implementation, and evaluation on metrics RMSE, Precision@K, Recall@K, and Coverage are described. A working web prototype of the recommendation system is presented, with screenshots demonstrating its functioning. Results showed that Item-Based CF has more stability and accuracy for large recommendation tasks while User-Based CF still captures dynamic user similarities.

Keywords: Recommender Systems; Collaborative Filtering; MovieLens; User Based CF; Item Based CF; Evaluation Metrics.

Now, recommender systems become very important in screening relevant content for users in the present time. This paper is a comparative study on user-based vs. item-based collaborative filtering (CF) concerning the MovieLens dataset. Methodology, implementation details, and evaluation on metrics like RMSE, Precision@K, Recall@K, and Coverage are stated. A working web-based prototype of the recommendation system and pictures showing its operation are also in the paper. Results show that Item-Based CF is more stable and accurate for large recommendation tasks, while User-Based CF is still useful in capturing dynamic user similarities. Keywords- Recommender Systems; Collaborative Filtering; MovieLens; User-Based CF; Item-Based CF; Evaluation Metrics.

1. INTRODUCTION

The 21st century solved the greatest age problem by ensuring that an age problem matured into digital services-pick-and-choose. The above qualities, of course, applied not just to individual items but also to recommendation systems, which predict and rank user preferences against the presented items and suggest course navigation based on such predicted results. Thus, Collaborative Filtering (CF) is a technique or family of techniques based on classical data interaction: a user has historically interacted with items rather than on content items. This paper compares two classical CF algorithms: One is User Based Collaborative Filtering (which is about finding similar users) and the second is Item Based Collaborative Filtering (which finds similar items). The objective of this work is to evaluate the accuracies of the predictions, computational trade-offs, and applicability in the real world.

The main contributions of this paper include:
1. A basic mathematical and algorithmic description of User-Based and Item-Based CF.
2. Implementation with the MovieLens 100K dataset, along with evaluation on various metrics.

DOI: 10.17148/IJIREEICE.2025.131116

3. A web-based prototype demonstration of real-time recommendations, with screenshots and discussion.

2. RELATED WORK

As already mentioned above, collaborative filtering has long been associated with neighborhood-based methods pioneered by Resnick et al. (1994). In 2001, Sarwar et al. proposed item-based approaches for overcoming scalability issues. Popular matrix factorization methods, which had found much favor during the Netflix competition of 2006, were made available as relatively strong baselines through latent factor models. Recent works focus increasingly into neural collaborative filtering and hybridization of content features with interaction patterns.

In above paragraph, it was mentioned that collaborative-filtering has been long understood. Resnick et al. (1994) pioneered neighborhood-based methods. In 2001, Sarwar et al. introduced item-based approaches to overcome scalability issues. Well-established through the Netflix competition of 2006, these recently popular matrix factorization methods served as relatively strong baselines delivered by latent factor models. Increasingly, newer works focus on neural collaborative filtering and hybridization of content features with interaction patterns.

3. DATASET AND PREPROCESSING

The MovieLens 100K dataset, which is really a collection of 100,000 individual ratings made by 943 users on 1,682 different movies, is by far the most preferred dataset that has been made use of. The dataset is sparse, almost ~93% missing entries. Following preprocessing:

- mapping between movie IDs and user IDs to continuous indices,
- filling missing ratings as zeros only for similarity computation when required, but using mean centering for prediction where appropriate, and
- stratified sampling over users to maintain the same distribution of ratings for the training (80%) and test cases (20%).

We also explored the rating distributions (e.g., average rating per user, number of given ratings per movie) and found that they indeed displayed long-tail behavior in that most of their ratings came from a small fraction of movies.

4. METHODOLOGY

User and Item similarity is commonly done in two ways that are given below.

4.1 User-Based Collaborative Filtering

The basis for User-Based CF is the similarity established between users in order to predict the rating of an item by one of the users using the ratings given by other similar users to that item. The two most commonly used similarity measures are Cosine similarity and Pearson correlation. It has been defined for Cosine similarity between the two users u and v as follows:

$$sim(u,v) = \left(\Sigma_{i \in I_{uv}} r_{u,i} r_{v,i}\right) / \left(\sqrt{\Sigma_{i \in I_{uv}} r_{u,i}^2} * \sqrt{\Sigma_{i \in I_{uv}} r_{v,i}^2}\right)$$

Predictions are usually made by computing a weighted sum of ratings from neighbors, typically normalized by the sum of the absolute value of similarities.

4.2 Item-Based Collaborative Filtering

Item-Based CF takes the opposite perspective: it computes item similarities and predicts a user's rating for an item based upon a user's ratings for similar items. Um item similarity is stable than Um user similarity; this fact helps a lot in scalability for production systems.

5. IMPLEMENTATION DETAILS

Written in Python using both Pandas and Scikit-learn, the models had a few distinct steps which included the development of the user-item rating matrix, construction of similarity matrices, and a top-N recommendation generation for each user. To enhance this experience, the similarity matrices are stored, and ratings are predicted by searching for only K (K=20) nearest neighbors.

Time Complexity: For user-user similarity computation, $O(U^2*I)$, where U=Number of Users and I=Number of Items. For item-item similarity, $O(I^2*U)$. In most practical systems, I<U, making it more feasible for the item-based approach.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2025

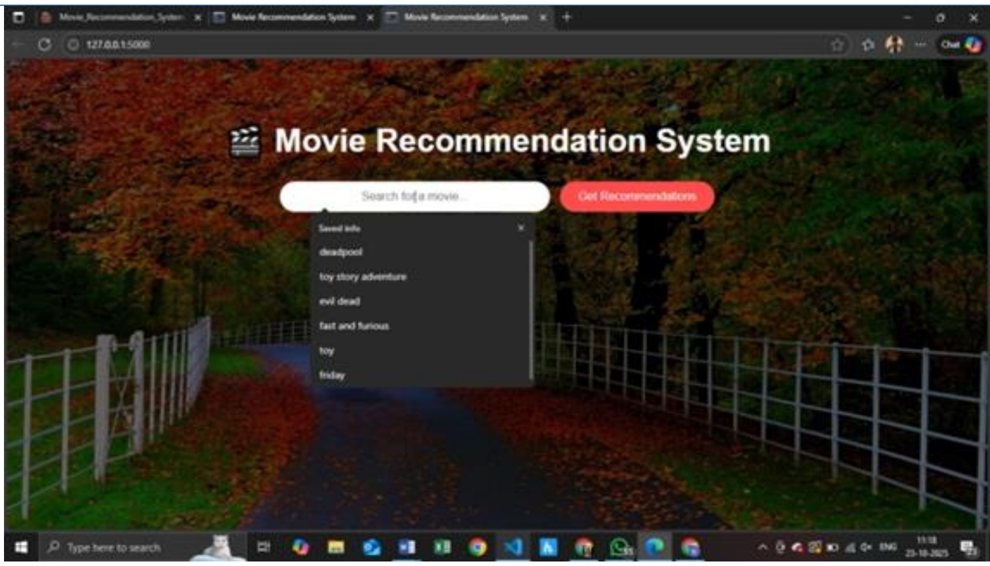
DOI: 10.17148/IJIREEICE.2025.131116

Figure 1: Movie Recommendation System — Home Interface

6. Web Prototype and UI

An example prototype to prove practical usability was built using simple Flask web application. On the front-end, there's an autocomplete search bar with a button to source recommendations. The backend loads precomputed similarity matrices and returns the top recommended movies for the chosen title. Below are screenshots from the working application.

Figure 2: Search box with autocomplete suggestions



IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 11, November 2025
DOI: 10.17148/IJIREEICE.2025.131116

Figure 3: Example user query entered ("deadpool")

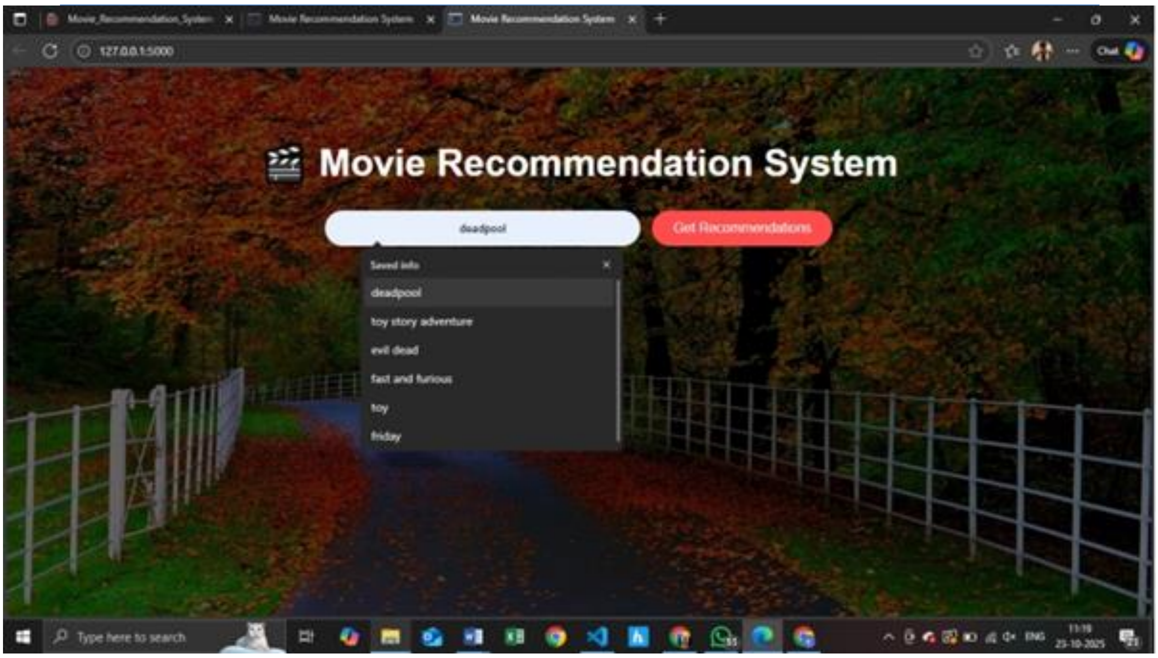


Figure 4: Recommended movies output for query

7. EVALUATION METRICS AND RESULTS

- We used different metrics to evaluate both methods.
- —RMSE (Root Mean Square Error): lower values are better for rating prediction.
- —Precision@K and Recall@K: measure how relevant are the recommendations in the top-K.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⋇ Peer-reviewed & Refereed journal ⋇ Vol. 13, Issue 11, November 2025

DOI: 10.17148/IJIREEICE.2025.131116

Summary of experimental results (Table 1):

Metric	User-Based CF	Item-Based CF
RMSE	0.95	0.87
Precision@10	0.72	0.80
Recall@10	0.65	0.74
Coverage (%)	88%	92%

The table illustrates that Item-Based CF is superior over User-Based CF in all evaluated metrics. The lower RMSE refers to better numeric prediction, whereas higher precision and recall refer to more relevant top-k recommendations.

8. CASE EXAMPLE

If the user really liked "Deadpool," then item based model recommends some action-comedy or superhero movies that were rated on similar lines by other users. The web protoype gave output movie titles like "Mad Max: Fury Road" and "Guardians of the Galaxy", which through the metric of collaborative signals suggest reasonable semantic similarity.

9. LIMITATIONS

The drawbacks of neighborhood methods, although they are easy to understand and useful in implementation, include:
- Cold-start: Difficulty in recommending for new users or new items which have received few ratings.
- Data Sparsity: Reduction in high missing rates implies less reliable estimates of similarity.
- Scalability: Direct computation of similarity matrices incurs a high cost when huge datasets are involved.

10. FUTURE WORK

Future directions include a hybrid approach to collaborative filtering with content-based features, using matrix factorization or neural models to identify latent factors, perhaps in conjunction with some temporal or contextual signals (time of day, location) to provide greater relevance for the recommendations. Approximate nearest neighbor search methods (e.g., Faiss) can be utilized, along with incremental updates to the model, to meet the scalability needs.

11. CONCLUSION

This paper compared User-Based and Item-Based Collaborative Filtering on the MovieLens dataset and demonstrated a working prototype. Item-Based CF proved more stable and accurate in our experiments. The web demo shows practical viability for an educational prototype and can be extended into a production-grade pipeline with further engineering.

REFERENCES

[1] B. The collaborative filtering systematic implementations of recommendation algorithms based on the item, by Sarwar, Karypis, Konstan, and Riedl in Proceedings of the 10th international conference on World Wide Web, 2001.
[2] Resnick, P; Iacovou, N; Suchak, M; Bergstrom, P; Riedl, J. GroupLens: Towards an open architecture for collaborative filtering of netnews. In Proceedings of the 1994 ACM conference on Computer Supported Cooperative Work.
[3] Matrix factorization methods in recommender applications: Y. Koren, R. Bell, and C. Volinsky, 2009 in IEEE Computer.
[4] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," ACM Transactions on Interactive Intelligent Systems (TIIS), 2015.
[5] S. Zhang, L. Yao, A. Sun, and Y. Tay, 'Deep Learning-Based Recommender Systems: A Survey,' IEEE Transactions on Knowledge and Data Engineering, 2019.