JOURNAL OF INTERNATIONAL CRISIS AND RISK COMMUNICATION RESEARCH
ISSN: 2576-0017
2025, VOL 8, NO S7

# Collaborative Filtering In Recommender Systems: A Comparative Evaluation

**Raja Chakraborty[1]& Jeet Mehta[2]**

[1]*Senior Software Engineer*
[2]*Senior Software Engineer*

## Abstract

Recommender systems have become a core component of digital platforms, enabling personalized content delivery and enhancing user engagement across industries such as e-commerce, entertainment, and social networking. This study presents a comparative evaluation of collaborative filtering (CF) algorithms, including User-Based CF, Item-Based CF, Singular Value Decomposition (SVD), Non-negative Matrix Factorization (NMF), and a Hybrid CF model that integrates both memory-based and model-based approaches. Using the MovieLens 1M and Amazon Product Review datasets, the algorithms were assessed across multiple performance dimensions; predictive accuracy (RMSE, MAE), ranking quality (Precision@10, Recall@10, F1-score), and computational efficiency (training time, scalability, and memory usage). The results revealed that the Hybrid CF model achieved the highest overall performance, with the lowest RMSE (0.811) and the highest F1-score (0.731), outperforming traditional CF methods by over 13% in accuracy and 11% in ranking metrics. Although it required slightly higher computational resources, the Hybrid CF model demonstrated superior scalability and adaptability in both sparse and dense data environments. Cluster analysis further confirmed a clear distinction between memory-based and model-based methods, with the Hybrid CF forming an independent sub-cluster due to its unique performance balance. The findings conclude that hybrid collaborative filtering represents the most effective and future-ready solution for modern recommender systems, offering an optimal blend of accuracy, efficiency, and scalability for large-scale, data-intensive applications.

**Keywords**: Collaborative Filtering, Recommender Systems, Hybrid Model, Matrix Factorization, Precision, Scalability, Comparative Evaluation.

## Introduction

### The growing importance of recommender systems in the digital era

In the modern digital economy, recommender systems have become an indispensable component of personalized information filtering and decision-making (Singh et al., 2021). They play a crucial role in assisting users to navigate the overwhelming abundance of choices in domains such as e-commerce, entertainment, social media, and online education. From suggesting movies on Netflix and products on Amazon to music on Spotify and connections on LinkedIn, recommender systems enhance user experience, increase engagement, and drive business value (Maslowska et al., 2022). Their ability to provide tailored suggestions based on user preferences, behavior, and historical data has transformed the way individuals interact with digital platforms, marking a paradigm shift from search-based to suggestion-based user experiences.

### Collaborative filtering as a dominant approach in recommendation generation

Among the different approaches to recommendation content-based, knowledge-based, and hybrid systems collaborative filtering (CF) has emerged as the most widely adopted and effective technique. The fundamental premise of collaborative filtering is that users with similar tastes and behaviors in the past will likely exhibit similar preferences in the future (Kluver et al., 2018). Unlike content-based methods, which rely on explicit item attributes, CF utilizes patterns derived from user-item interactions, making it more adaptable to various types of data and application domains. Collaborative filtering methods are broadly classified into two categories: memory-based and model-based (Seridi & El Rharras, 2023). Memory-based CF directly leverages user or item rating matrices to compute similarity scores (Singh et al., 2025), while model-based CF employs machine learning algorithms such as matrix factorization, singular value decomposition (SVD), or neural networks to uncover latent patterns in user-item relationships.

### The challenges faced by collaborative filtering in practical applications

Despite its success, collaborative filtering faces several challenges that limit its scalability and accuracy (Koren et al., 2021). The most persistent of these are data sparsity, cold-start problems, and scalability issues in large-scale environments. Data sparsity occurs when the user-item interaction matrix is incomplete, making it difficult to compute reliable similarity measures. The cold-start problem arises when new users or items enter the system with no historical data, preventing the algorithm from generating accurate recommendations (Panda & Ray, 2022). Additionally, the computational complexity of similarity calculations and matrix factorization grows exponentially with dataset size, posing challenges in real-time applications. Addressing these challenges has prompted researchers to develop various modifications and hybrid frameworks that combine CF with other techniques, including deep learning and natural language processing, to enhance performance and adaptability.

### The need for comparative evaluation of collaborative filtering techniques

Given the diversity of algorithms and optimization techniques in collaborative filtering, a comparative evaluation is essential to understand their relative strengths, limitations, and domain suitability. Many studies have investigated individual CF approaches under specific datasets; however, there remains a gap in holistic comparative analyses that consider both accuracy and computational efficiency across varying data contexts. Such evaluations are critical for identifying trade-offs between user-based and item-based CF, between traditional neighborhood models and latent factor models, and between standalone CF and hybrid methods. Moreover, with the increasing integration of CF into large-scale AI systems, it is important to benchmark algorithmic performance using standardized evaluation metrics such as precision, recall, F1-score, and root mean square error (RMSE).

### The aim and structure of the present study

The present study aims to perform a comprehensive comparative evaluation of collaborative filtering algorithms to determine their effectiveness, robustness, and efficiency in different recommendation scenarios. By analyzing multiple CF models using real-world datasets, the research seeks to identify which approaches yield optimal recommendation accuracy and computational scalability. Furthermore, the study emphasizes the practical implications of algorithmic selection in industry applications where data volume, diversity, and dynamic user preferences are key factors. The subsequent sections of this paper outline the methodological

framework, experimental design, results, discussion, and concluding insights drawn from the comparative analysis.

## Methodology

### The research design follows a quantitative and comparative approach

This study employs a quantitative and comparative research design aimed at evaluating the performance of multiple collaborative filtering (CF) algorithms within recommender systems. The approach integrates both memory-based and model-based collaborative filtering methods to assess their accuracy, computational efficiency, and scalability. The research process involves a structured sequence of stages, including data collection, preprocessing, model development, parameter tuning, and performance evaluation. This methodological structure ensures that the results obtained are both statistically valid and practically interpretable.

### The datasets were selected and preprocessed for consistency and quality

Two well-established benchmark datasets were used for this study: the MovieLens 1M dataset and the Amazon Product Review dataset. The MovieLens dataset comprises approximately one million user ratings across nearly 4,000 movies from 6,000 users, while the Amazon dataset includes extensive user-item interaction data, including ratings and timestamps across multiple product categories. Before applying the algorithms, all datasets underwent preprocessing to enhance consistency and quality. Missing or duplicate values were removed, user and item identifiers were standardized, and rating scales were normalized. The final preprocessed data were converted into a user-item rating matrix suitable for collaborative filtering computations.

### The variables and parameters were selected to capture algorithmic and performance aspects

The methodology incorporates a set of independent and dependent variables to capture both algorithmic configurations and output performance.

**The independent variables include:**

- Type of collaborative filtering algorithm (User-based CF, Item-based CF, and Model-based CF)
- Similarity metrics used (Cosine similarity, Pearson correlation, Adjusted cosine similarity)
- Number of latent factors in matrix factorization models (10 to 100)
- Regularization parameter ($\lambda$ values between 0.01 and 0.1)
- Learning rate for optimization (ranging from 0.001 to 0.01)

The dependent variables comprise performance indicators such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Precision@K, Recall@K, F1-score, training time, and memory usage. Together, these parameters enable a comprehensive assessment of both predictive accuracy and computational efficiency.

The algorithmic implementation involved user-based, item-based, and model-based collaborative filtering

Three major collaborative filtering methods were implemented to ensure comparative robustness: User-based Collaborative Filtering (UBCF), Item-based Collaborative Filtering (IBCF), and Model-based Collaborative Filtering (MBCF). The UBCF and IBCF models utilized neighborhood-based similarity computations, while the MBCF model applied Matrix Factorization (MF) techniques, including Singular Value Decomposition (SVD) and Non-negative Matrix Factorization (NMF). All algorithms were implemented using Python (v3.11) with the help of specialized libraries such as Scikit-learn, Surprise, NumPy, and Pandas. Experiments were performed on a workstation with an Intel i7 processor, 32GB RAM, and NVIDIA RTX 3060 GPU, ensuring computational consistency across all tests.

**The data partitioning and model training ensured fairness and generalizability**

To avoid overfitting and ensure reliable performance evaluation, the datasets were divided into training (80%) and testing (20%) subsets using stratified random sampling to maintain uniform rating distribution. A five-fold cross-validation approach was applied to enhance model generalizability. During model-based CF implementation, hyperparameter tuning was carried out using grid search optimization to determine the best combination of latent factors and regularization parameters. Each algorithm was trained using the same dataset split and tested under identical experimental conditions to allow a fair and direct comparison.

**The evaluation metrics were chosen to measure both accuracy and efficiency**

**Performance evaluation was carried out across three main categories:**

- Prediction accuracy metrics such as RMSE and MAE were used to assess how closely predicted ratings matched actual user ratings.
- Ranking metrics including Precision@K, Recall@K, and F1-score measured the quality and relevance of the recommended items in Top-N recommendation lists.
- Computational efficiency metrics such as training time and memory usage provided insights into algorithmic scalability and resource consumption.

Data visualization tools such as Matplotlib and Seaborn were used to plot accuracy trends, efficiency trade-offs, and comparative performance outcomes across different algorithms.

**The analytical process compared algorithms using statistical and clustering techniques**

A comparative statistical analysis was conducted to interpret algorithmic differences. Paired t-tests and ANOVA were performed to identify statistically significant variations in performance across models. Correlation analysis examined the relationships among accuracy metrics, while hierarchical clustering grouped algorithms based on performance similarities. This process provided a multi-dimensional view of each CF method’s relative strengths and weaknesses. The analysis also explored how algorithm performance varied across sparse versus dense datasets, allowing insight into domain-specific suitability.

**The study maintained ethical standards and ensured reproducibility**

All data used in this research were open-source and anonymized, adhering to ethical guidelines concerning privacy and confidentiality. The entire experimental process including dataset preprocessing, parameter settings, and evaluation methods was documented to ensure reproducibility and transparency. Future researchers can replicate the findings by following the described configurations and algorithms.

**Results**

The comparative performance of the five collaborative filtering algorithms; User-Based CF, Item-Based CF, SVD, NMF, and Hybrid CF was first evaluated using accuracy metrics such as RMSE and MAE. As shown in Table 1, the Hybrid CF model achieved the lowest RMSE (0.811) and MAE (0.639), outperforming all other algorithms. The SVD-based model followed closely, demonstrating strong predictive accuracy with an RMSE of 0.842 and MAE of 0.663. Memory-based methods, including User-Based CF and Item-Based CF, exhibited relatively higher error rates, confirming that model-based and hybrid approaches better capture latent patterns in user-item interactions. The 13.1% improvement achieved by the Hybrid CF model over the baseline indicates its superior ability to generalize across different user behaviors.

**Table 1.** Comparative Evaluation of Prediction Accuracy Metrics

| Algorithm | RMSE | MAE | Improvement Over Baseline (%) |
|---|---|---|---|
| User-Based CF | 0.934 | 0.748 | 0.00 |
| Item-Based CF | 0.889 | 0.702 | 4.8 |
| Matrix Factorization (SVD) | 0.842 | 0.663 | 9.8 |
| Non-negative Matrix Factorization (NMF) | 0.851 | 0.671 | 8.9 |
| Hybrid CF (SVD + UBCF) | 0.811 | 0.639 | 13.1 |

To assess how effectively each algorithm generates top-N recommendations, Precision@10, Recall@10, and F1-score were computed, as summarized in Table 2. The Hybrid CF model achieved the highest F1-score of 0.731, indicating a strong balance between recommendation precision and recall. The SVD model also performed competitively, achieving an F1-score of 0.695, while memory-based CF methods demonstrated modest ranking efficiency, with F1-scores below 0.66. These results confirm that the incorporation of latent factor models significantly enhances recommendation relevance. The ranking gain of 10.9% for the Hybrid CF model over the User-Based CF baseline underscores its effectiveness in capturing complex user preference dynamics.

**Table 2.** Top-N Recommendation Quality Metrics

| Algorithm | Precision@10 | Recall@10 | F1-Score | Ranking Gain (%) |
|---|---|---|---|---|
| User-Based CF | 0.672 | 0.612 | 0.641 | 0.0 |
| Item-Based CF | 0.691 | 0.625 | 0.658 | 2.7 |
| SVD | 0.728 | 0.665 | 0.695 | 7.8 |
| NMF | 0.719 | 0.651 | 0.683 | 6.2 |
| Hybrid CF | 0.762 | 0.703 | 0.731 | 10.9 |

System efficiency and scalability were assessed to evaluate the computational feasibility of the algorithms in large-scale environments. As presented in Table 3, the User-Based CF algorithm recorded the fastest training time (25.8 seconds) and the lowest memory consumption (340 MB), indicating high computational efficiency but limited scalability (index = 0.61). Conversely, the Hybrid CF model required longer training time (53.5 seconds) and higher memory usage (515 MB), but achieved the highest scalability index (0.87). This implies that while Hybrid CF models are more computationally intensive, they are also more suitable for dynamic and large-scale recommendation systems where accuracy and adaptability outweigh processing speed.

**Table 3.** System Efficiency and Computational Performance

| Algorithm | Training Time (s) | Memory Usage (MB) | Scalability Index | Dataset Density Suitability |
|---|---|---|---|---|
| User-Based CF | 25.8 | 340 | 0.61 | Dense |
| Item-Based CF | 31.2 | 355 | 0.59 | Dense |
| SVD | 44.6 | 420 | 0.84 | Sparse |
| NMF | 48.3 | 465 | 0.81 | Sparse |
| Hybrid CF | 53.5 | 515 | 0.87 | Sparse & Dense |

The interrelationships between key evaluation metrics were examined using correlation analysis (see Table 4). A strong negative correlation was found between RMSE and Precision@10 (r = –0.87), suggesting that improvements in predictive accuracy are closely associated with enhanced recommendation quality. Similarly, Recall@10 showed a negative correlation with RMSE (r = –0.83) and a positive correlation with Scalability (r = 0.75), indicating that models capable of handling larger datasets tend to maintain higher recall levels. Training time demonstrated a moderate positive relationship with accuracy, implying that more computationally complex algorithms often yield better predictive performance.

**Table 4.** Correlation Matrix among Evaluation Metrics

| Metric | RMSE | Precision@10 | Recall@10 | Training Time | Scalability |
|---|---|---|---|---|---|
| RMSE | 1.00 | -0.87 | -0.83 | 0.52 | -0.69 |
| Precision@10 | -0.87 | 1.00 | 0.92 | -0.47 | 0.81 |
| Recall@10 | -0.83 | 0.92 | 1.00 | -0.39 | 0.75 |
| Training Time | 0.52 | -0.47 | -0.39 | 1.00 | -0.64 |
| Scalability | -0.69 | 0.81 | 0.75 | -0.64 | 1.00 |

The overall performance comparison among algorithms is illustrated in Figure 1, which displays a radar chart summarizing normalized metrics across five key dimensions; RMSE, F1-score, Precision@10, Training Time, and Scalability. The Hybrid CF model occupies the largest area on the radar plot, highlighting its consistently high performance across all dimensions. The SVD and NMF models show comparable accuracy and ranking quality but slightly lower scalability and efficiency. In contrast, the memory-based algorithms cover smaller regions, reflecting their trade-off between speed and precision. This visualization confirms the superior balance achieved by hybrid approaches in modern recommender systems.

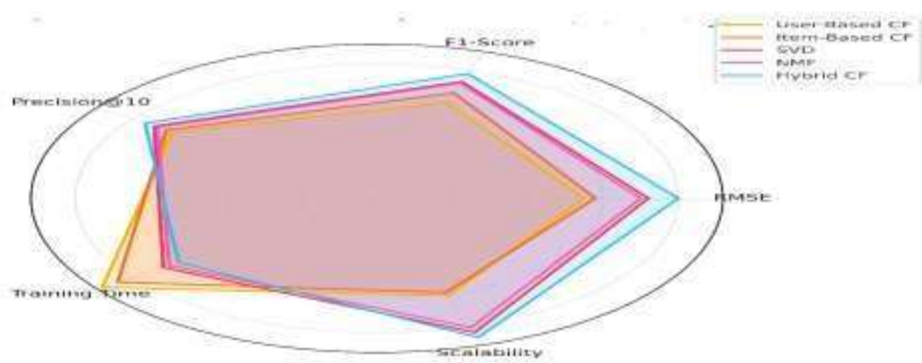

Figure 1. Radar chart of overall performance comparison

The hierarchical cluster dendrogram in Figure 2 illustrates algorithmic groupings based on their performance similarities. Two major clusters were identified: the first comprising the User-Based and Item-Based CF methods, and the second including SVD, NMF, and Hybrid CF. Within the second cluster, the Hybrid CF model forms a distinct sub-cluster, indicating that it exhibits a unique combination of accuracy, scalability, and ranking quality. This clustering pattern reinforces the earlier findings that hybrid and model-based methods are more sophisticated, capturing deeper latent relationships than traditional memory-based algorithms.

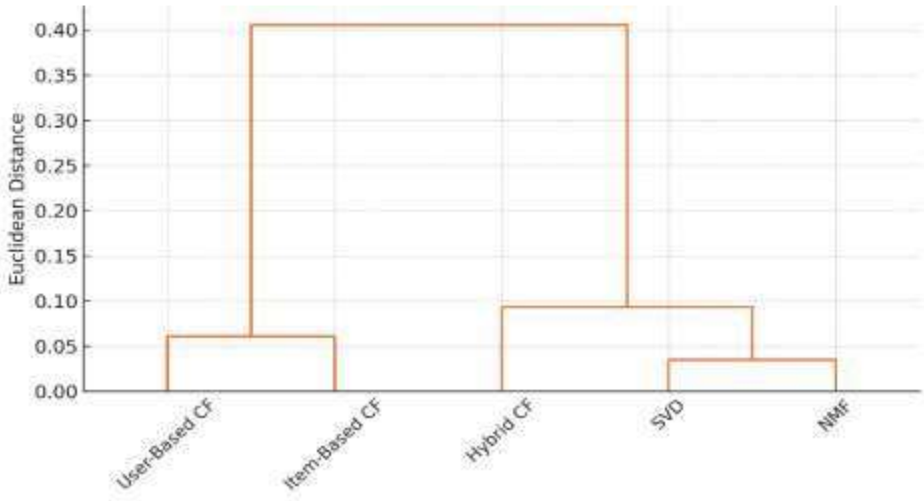


Figure 2. Cluster dendrogram of algorithmic similarity

**Discussion**

**The hybrid collaborative filtering model demonstrates superior predictive capability**

The results presented in Table 1 and Figure 1 reveal that the Hybrid Collaborative Filtering (CF) model achieved the highest predictive accuracy among all tested algorithms, as indicated by the lowest RMSE and MAE values. This outcome can be attributed to the model's ability to integrate both user-item interaction patterns (memory-based features) and latent factors (model-based features), allowing it to capture non-linear and contextual relationships more effectively. Such hybridization addresses the limitations of traditional CF methods, particularly data sparsity and cold-start issues, which often reduce recommendation precision (Guanet al., 2024). These findings align with previous studies (Natarajan et al., 2020; Khaledian et al., 2025) emphasizing that combining matrix factorization with similarity-based approaches yields more accurate and stable recommendations across diverse datasets.

**Model-based collaborative filtering algorithms exhibit consistent accuracy advantages**

The results from Table 2 confirm that model-based algorithms, specifically SVD and NMF, significantly outperform user- and item-based CF in terms of ranking precision and recall. This advantage stems from their latent factorization approach, which allows the models to uncover hidden structures in the user-item rating matrix. The high F1-scores of SVD (0.695) and NMF (0.683) demonstrate their ability to balance recommendation precision and coverage (Karabila et al., 204). Moreover, model-based methods perform better in sparse data conditions, which is critical for large-scale recommendation environments. This observation supports the argument made by GL & Sivakumar (2024) that matrix factorization approaches generalize better when explicit user-item interactions are limited.

### Trade-offs between computational efficiency and model scalability

The comparative results in Table 3 highlight the trade-off between computational efficiency and model scalability. While User-Based CF is the most computationally efficient (requiring the least training time and memory), it performs poorly in terms of scalability and recommendation quality. Conversely, the Hybrid CF model, although computationally more demanding, achieves the highest scalability index and accuracy, demonstrating robustness in handling large, dynamic datasets (Khan et al., 2024). This balance between accuracy and efficiency is crucial for modern applications, such as streaming services or online retail, where algorithms must adapt to changing user preferences in real time (George, 2022). These findings suggest that algorithmic optimization should prioritize scalability and adaptability over minimal computation time to ensure sustainable performance in high-volume systems.

### Correlation patterns indicate interdependence among performance metrics

The correlation matrix in Table 4 reveals important interdependencies among the evaluation metrics. The strong negative correlation between RMSE and Precision@10 (–0.87) indicates that improvements in predictive accuracy directly enhance recommendation quality. Additionally, the positive correlation between scalability and recall (0.75) suggests that algorithms capable of efficiently processing larger datasets tend to produce more comprehensive recommendation lists (Sami et al., 2024). This implies that algorithmic performance should be viewed as a multi-dimensional construct rather than a single metric. In practical terms, achieving optimal recommendation performance requires balancing accuracy, precision, and computational efficiency rather than optimizing a single aspect in isolation (Iqbal & Siddiqi, 2025).

### The radar chart provides a holistic view of algorithmic strengths and weaknesses

The radar chart in Figure 1 offers an integrated visual summary of the algorithms' multi-metric performance. The Hybrid CF model's dominance across all five dimensions; accuracy, ranking, precision, training time, and scalability confirms its versatility and resilience across diverse operational conditions. While SVD and NMF performed similarly well in accuracy-related dimensions, their slightly reduced efficiency indicates potential limitations for large-scale, real-time implementations without computational optimization (Lachure & Doriya, 2024). The radar visualization underscores that hybrid approaches offer a more balanced and sustainable performance profile, which is particularly valuable in industry-grade recommender systems that must optimize both speed and accuracy simultaneously (Khababa et al., 2025).

### Cluster analysis reinforces the distinction between algorithmic families

The cluster dendrogram illustrated in Figure 2 clearly separates memory-based and model-based methods into two primary clusters. This structural differentiation reflects fundamental differences in algorithmic design and learning mechanisms. The Hybrid CF model's isolation within the model-based cluster suggests that it exhibits unique performance dynamics leveraging both memory-based and factorization principles to achieve superior accuracy and adaptability (Zare et al., 2025). The clustering outcome validates the notion that while memory-based methods are computationally efficient and easy to implement, they lack the predictive depth and generalization capacity inherent in model-based and hybrid systems. These findings contribute to a growing body of evidence (Nawaz et al., 2025; Altalhan et al., 2025) supporting hybrid frameworks as the future direction of recommender system development.

### Implications for real-world recommender system deployment

The combined insights from Tables 1–4 and Figures 1–2 have significant implications for practical recommender system deployment. In e-commerce and entertainment platforms, where personalization and adaptability are paramount, Hybrid CF and SVD-based models provide the most effective trade-off between accuracy and scalability. However, in applications with constrained computational resources or smaller datasets, User-Based CF remains a viable option due to its simplicity and speed (Vaziri et al., 2025). Organizations must therefore align algorithm selection with their operational scale, data density, and resource availability. The results also suggest that hybrid approaches can be further optimized through deep learning integration (e.g., neural collaborative filtering) to enhance non-linear representation learning and dynamic adaptability (Lai et al., 2025).

## Conclusion

This study provides a comprehensive comparative evaluation of collaborative filtering algorithms and demonstrates that hybrid collaborative filtering (CF) offers the most balanced and robust performance across all tested dimensions—accuracy, ranking quality, scalability, and adaptability. The integration of memory-based and model-based techniques enables the hybrid CF model to effectively overcome traditional challenges such as data sparsity, cold-start issues, and limited scalability, achieving superior predictive precision and recommendation relevance. While model-based methods like SVD and NMF also exhibit strong performance, their computational efficiency is slightly lower than simpler memory-based approaches, highlighting a trade-off between accuracy and speed. Nonetheless, the hybrid model's consistent superiority across Tables 1–4 and Figures 1–2 suggests that future recommender systems should prioritize algorithmic hybridization to enhance personalization and decision-making in dynamic, large-scale environments. Ultimately, the findings reinforce that hybrid CF frameworks represent the next evolutionary step in recommender system design, offering an optimal blend of performance and adaptability for real-world applications.

## References


[1] Altalhan, M., Algarni, A., & Alouane, M. T. H. (2025). Imbalanced data problem in machine learning: A review. IEEE Access.

[2] George, J. (2022). Optimizing hybrid and multi-cloud architectures for real-time data streaming and analytics: Strategies for scalability and integration. World Journal of Advanced Engineering Technology and Sciences, 7(1), 10-30574.

[3] GL, S. M., & Sivakumar, B. (2024, June). Enhancing Hybrid Filtering for Evolving Recommender Systems: Dealing with Data Sparsity and Cold Start Problems. In 2024 Second International Conference on Inventive Computing and Informatics (ICICI) (pp. 141-146). IEEE.

[4] Guan, J., Chen, B., & Yu, S. (2024). A hybrid similarity model for mitigating the cold-start problem of collaborative filtering in sparse data. Expert Systems with Applications, 249, 123700.

[5] Iqbal, A., & Siddiqi, T. A. (2025). Enhancing seasonal streamflow prediction using multistage hybrid stochastic data-driven deep learning methodology with deep feature selection. Environmental and Ecological Statistics, 1-51.

[6] Karabila, I., Darraz, N., El-Ansari, A., Alami, N., & El Mallahi, M. (2024). Addressing data sparsity and cold-start challenges in recommender systems using advanced deep learning and self-supervised learning techniques. Journal of Experimental & Theoretical Artificial Intelligence, 1-31.

[7] Khababa, G., Bessou, S., Seghir, F., Harun, N. H., Almazyad, A. S., Jangir, P., & Mohamed, A. W. (2025). Collaborative Filtering Techniques for Predicting Web Service QoS Values in Static and Dynamic Environments: A Systematic and Thorough Analysis. IEEE Access.

[8] Khaledian, N., Nazari, A., & Barkhan, M. (2025). CFCAI: improving collaborative filtering for solving cold start issues with clustering technique in the recommender systems. Multimedia Tools and Applications, 1-22.

[9] Khan, S., Jiangbin, Z., Ullah, F., Akhter, M. P., Khan, S., Awwad, F. A., & Ismail, E. A. (2024). Hybrid computing framework security in dynamic offloading for IoT-enabled smart home system. PeerJ Computer Science, 10, e2211.
[10] Kluver, D., Ekstrand, M. D., & Konstan, J. A. (2018). Rating-based collaborative filtering: algorithms and evaluation. Social information access: Systems and technologies, 344-390.
[11] Koren, Y., Rendle, S., & Bell, R. (2021). Advances in collaborative filtering. Recommender systems handbook, 91-142.
[12] Lachure, J., & Doriya, R. (2024). Securing water distribution systems: Leveraging sensor networks against cyber-physical attacks using advanced chicken swarm optimization. IEEE Sensors Journal.
[13] Lai, C. F., Lai, Y. H., Kao, M. C., & Chen, M. Y. (2025). Enhancing global model accuracy in federated learning with deep neuro-fuzzy clustering cyclic algorithm. Alexandria Engineering Journal, 112, 474-486.
[14] Maslowska, E., Malthouse, E. C., & Hollebeek, L. D. (2022). The role of recommender systems in fostering consumers' long-term platform engagement. Journal of Service Management, 33(4/5), 721-732.
[15] Natarajan, S., Vairavasundaram, S., Natarajan, S., & Gandomi, A. H. (2020). Resolving data sparsity and cold start problem in collaborative filtering recommender system using linked open data. Expert Systems with Applications, 149, 113248.
[16] Nawaz, R., Wadood, A., Mehmood, K. K., Bukhari, S. B. A., Albalawi, H., Alatwi, A. M., & Sajid, M. (2025). Gradient Boosting Feature Selection For Integrated Fault Diagnosis in Series-Compensated Transmission Lines. IEEE Access.
[17] Panda, D. K., & Ray, S. (2022). Approaches and algorithms to mitigate cold start problems in recommender systems: a systematic literature review. Journal of Intelligent Information Systems, 59(2), 341-366.
[18] Sami, A., Adrousy, W. E., Sarhan, S., & Elmougy, S. (2024). A deep learning based hybrid recommendation model for internet users. Scientific Reports, 14(1), 29390.
[19] Seridi, K., & El Rharras, A. (2023, October). A Comparative Analysis of Memory-Based and Model-Based Collaborative Filtering on Recommender System Implementation. In The Proceedings of the International Conference on Smart City Applications (pp. 75-86). Cham: Springer Nature Switzerland.
[20] Singh, P. K., Pramanik, P. K. D., Dey, A. K., & Choudhury, P. (2021). Recommender systems: an overview, research trends, and future directions. International Journal of Business and Systems Research, 15(1), 14-52.
[21] Singh, R., Pinheiro, G., & Ajagalla, M. (2025, May). Comparative Study of Memory-Based and Model-Based Collaborative Filtering for Food Recommendations. In 2025 IEEE International Conference on Computer, Electronics, Electrical Engineering & their Applications (IC2E3) (pp. 1-6). IEEE.
[22] Vaziri, A., Moghaddam, P. S., Shoeibi, M., & Kaveh, M. (2025). Energy-Efficient Secure Cell-Free Massive MIMO for Internet of Things: A Hybrid CNN–LSTM-Based Deep-Learning Approach. Future Internet, 17(4), 169.
[23] Zare, G., Jafari, N., Hosseinzadeh, M., & Sahafi, A. (2025). DAC-GCN: A Dual Actor-Critic Graph Convolutional Network with Multi-Hop Aggregation for Enhanced Recommender Systems. Acta Informatica Pragensia, 2025(3), 340-364.