Educational Administration: Theory and Practice
2024, 30(6)(s), 91-98
ISSN: 2148-2403
https://kuey.net/

Research Article

"Analyzing User-Based and Item-Based Recommender Systems: A Comparative Examination"

Dr. Divyesh P. Gohel[1*], Dr. Pratik A. Vanjara[2]

[1*]Assistant Professor, Department of Computer Science & Information Technology, Atmiya University, Rajkot.
[2]Head & Assistant Professor, Department of Computer Science, Shree M P Shah Commerce College, Surendranagar.

Citation: Dr. Divyesh P. Gohel et al. (2024) "Analyzing User-Based and Item-Based Recommender Systems: A Comparative Examination", *Educational Administration: Theory And Practice*, *30* (6) (s), 91-98
Doi: 10.53555/kuey.v30i6(S).5331

ARTICLEINO	ABSTRACT
	This research paper presents an in-depth analysis and comparative examination of two prominent recommender system approaches: user-based collaborative filtering and item-based collaborative filtering. Recommender systems play a pivotal role in enhancing user experiences by providing personalized recommendations. This study aims to dissect the mechanisms, strengths, and limitations of user-based and item-based methods, offering valuable insights for researchers and practitioners in the field. Through a comprehensive evaluation, we aim to shed light on the comparative effectiveness of these approaches in different scenarios and highlight considerations for their practical implementation.

1. Introduction

1.1 Background
Recommender systems have become indispensable tools in various domains, influencing user choices in content consumption, e-commerce, and beyond. Among the diverse approaches, user-based and item-based collaborative filtering stand out as widely adopted techniques. Understanding their nuances and comparative performance is crucial for tailoring recommendations to user preferences effectively.

1.2 Motivation
The motivation behind this research lies in the need to conduct a comprehensive analysis of user-based and item-based recommender systems. By scrutinizing their underlying principles, strengths, and weaknesses, this study aims to provide actionable insights for practitioners seeking to implement or enhance recommendation algorithms.

2. User-Based Collaborative Filtering

User-Based Collaborative Filtering (CF) is a technique used in recommendation systems to generate personalized recommendations by identifying similarities between users. It relies on the assumption that users who have exhibited similar preferences or behaviors in the past are likely to have similar preferences in the future.
In User-Based Collaborative Filtering:
- User Similarity: The system assesses the similarity between users based on their interactions with items. This is typically done by calculating a similarity metric, such as cosine similarity, Pearson correlation coefficient, Jaccard index, or others, using the users' ratings or interactions.
- Neighborhood Selection: Once similarities between users are determined, the system selects a group of users, often referred to as a neighborhood or set of nearest neighbors, who are most similar to the target user.
- Recommendation Generation: Based on the interactions of similar users, the system generates recommendations for the target user. Items that the similar users have liked or interacted with but the target user has not yet engaged with are recommended to the target user.
- Prediction of Ratings: In scenarios where explicit ratings are missing for certain items by a user, User-Based CF can predict these missing ratings by aggregating the ratings of similar users for those items.

User-Based Collaborative Filtering does not require explicit information about items but solely relies on user-item interactions. It is suitable for scenarios where item features or characteristics are sparse or not readily available. However, it can suffer from scalability issues in large systems due to the computation involved in calculating user similarities.

User-Based Collaborative Filtering is a user-centric approach that leverages similarities between users' preferences to make personalized recommendations, thereby enhancing user experiences in various domains such as e-commerce, entertainment, and content platforms.

2.1 Similarity Measures in User-Based CF

Similarity measures in user-based collaborative filtering (CF) are mathematical techniques used to quantify the similarity or proximity between users based on their interactions with items. These measures play a crucial role in identifying users with similar preferences to make personalized recommendations.
There are several similarity measures commonly used in user-based collaborative filtering:

2.1.1 Cosine Similarity

- Measures the cosine of the angle between user vectors, emphasizing shared preferences.
- This measure calculates the cosine of the angle between two user vectors in the user-item interaction matrix.

It assesses the similarity of user preferences by considering the direction of their ratings in the multidimensional space of items. It is particularly effective in high-dimensional spaces and is robust against variations in the magnitude of ratings.

2.1.2 Pearson Correlation Coefficient

- Captures linear relationships between user ratings, providing a robust similarity metric.
- It measures the linear correlation between the ratings given by two users. It captures the strength and direction of the linear relationship between the users' ratings, providing a measure of similarity that is sensitive to the relative differences in ratings given by users.

2.1.3 Jaccard Index

- This measure assesses the similarity between two users based on the number of items they have interacted with. It calculates the intersection over the union of items rated by both users. It is especially useful when dealing with binary data (e.g., like/dislike or purchase/no purchase).

2.1.4 Spearman Rank Correlation

- It evaluates the strength and direction of the monotonic relationship between two users' ratings. Unlike Pearson correlation, Spearman rank correlation assesses the association between ranks of the ratings rather than the raw ratings themselves. It is less sensitive to outliers and deviations from linearity.

The choice of similarity measure in user-based collaborative filtering depends on factors such as the nature of the dataset (e.g., sparsity, scale of ratings), computational efficiency, and the specific characteristics of the recommendation problem being addressed. Different measures may yield different results and performance in recommending items to users, and often a combination or hybridization of measures is used to enhance the accuracy of recommendations.

2.2 User-Item Interaction Matrix

The User-Item Interaction Matrix is a fundamental data structure used in recommendation systems, specifically in collaborative filtering-based approaches. It represents the interactions or engagements between users and items within a system.
In simple terms, this matrix is a two-dimensional grid where rows represent users and columns represent items. The cells of the matrix contain information about the interactions or preferences of users for specific items. For example, in a movie recommendation system, the matrix might contain user ratings for different movies, where each cell represents a user's rating for a particular movie.
This matrix is typically sparse because not all users interact with or rate all items available in the system. It serves as the foundational dataset for collaborative filtering algorithms, allowing them to analyze patterns in user-item interactions to make personalized recommendations.
By analyzing this matrix, recommendation algorithms identify similarities between users or items based on their interaction patterns. These similarities are used to predict missing ratings or recommend new items to users based on the preferences of similar users or items.
Overall, the User-Item Interaction Matrix is a key component in collaborative filtering methods, enabling the systems to understand and predict user preferences by capturing their interactions with items in a structured manner.

3. Item-Based Collaborative Filtering

Item-Based Collaborative Filtering (CF) is a technique used in recommendation systems to provide personalized suggestions to users based on similarities between items rather than users themselves.
Item-Based Collaborative Filtering is advantageous in scenarios where the number of items is smaller compared to the number of users, as it involves computing item-item similarities rather than user-user similarities. It tends to perform well in scenarios where users have stable preferences over time and new items

are added to the system. However, Item-Based CF might face challenges in dynamic environments where user preferences shift rapidly or when the number of items is exceedingly large, as it requires the computation of item similarities for each item in the catalog.
Item-Based Collaborative Filtering is an item-centric approach that exploits similarities between items based on user interactions to offer personalized recommendations, contributing to improved user experiences in various recommendation domains.

3.1 Item Similarity Metrics

Item similarity metrics are mathematical measures used in item-based collaborative filtering to quantify the similarity or closeness between items in a recommendation system. These metrics are crucial in determining the relationships and connections between items based on user interactions or attributes.

3.1.1 Cosine Similarity

- Evaluates the similarity between item vectors, emphasizing common users' interactions.
- This metric calculates the cosine of the angle between two item vectors in the user-item interaction matrix. It measures the similarity in user-item rating patterns across items. Items with similar ratings by the same set of users tend to have a higher cosine similarity.

3.1.2 Adjusted Cosine Similarity

- Accounts for user rating biases, enhancing the accuracy of item similarity measurements.
- Adjusted cosine similarity accounts for user biases by normalizing item ratings based on the average ratings given by users. It subtracts the user-specific average ratings from the item ratings before calculating cosine similarity. This normalization helps in handling variations in users' rating scales.

3.1.3 Jaccard Index

- The Jaccard Index calculates similarity based on the number of users who have interacted with or rated two items. It measures the intersection over the union of users who have engaged with both items. It is particularly useful in binary scenarios (e.g., like/dislike).

3.1.4 Pearson Correlation Coefficient

- This metric measures the linear correlation between ratings given to two items by users. It captures the strength and direction of the linear relationship between the ratings, indicating how items' ratings move together.

3.1.5. Euclidean Distance or Manhattan Distance

- These distance-based metrics compute the geometric distance between items in a multidimensional space. They quantify the dissimilarity or proximity between items based on their rating patterns.

The choice of item similarity metric often depends on the characteristics of the dataset, the nature of user-item interactions, and the specific recommendation scenario. Different similarity metrics might yield different results in terms of item similarity and consequently affect the quality of recommendations in collaborative filtering systems. Hybrid approaches or combinations of these metrics are also used to improve the accuracy and effectiveness of item-based recommendation systems.

3.2 Recommendations in Item-Based CF

Identifying similar items to those a user has interacted with and recommending based on those similarities.
Recommendations in Item-Based Collaborative Filtering (CF) refer to the process of suggesting items to users based on the similarities between items within a recommendation system.
In Item-Based CF:

- Similarity Calculation: The system first computes similarities between items based on user interactions. It determines how closely related or similar items are to each other using similarity metrics such as cosine similarity, adjusted cosine similarity, or others.
- Item Neighborhood: For each item in the system, a set of similar or related items, known as the item's neighborhood, is identified based on their calculated similarities. This neighborhood comprises items that share substantial similarities with the chosen item.
- Recommendation Generation: When a user expresses interest in or interacts with a particular item, the system identifies the neighborhood of that item and recommends other items from that neighborhood to the user. These recommended items are those that share high similarity scores with the chosen item.
- User Preferences Consideration: Recommendations are tailored to the specific user based on their interactions or preferences. The system suggests items that are similar to those the user has already shown interest in or interacted with, aiming to provide personalized and relevant suggestions.

The primary objective of recommendations in Item-Based CF is to enhance user experiences by offering personalized and relevant item suggestions. By leveraging similarities between items based on user interactions, the system aims to predict and recommend items that a user might be interested in but has not yet encountered within the system.

Item-Based CF recommendations are widely used in various domains such as e-commerce, content platforms, and entertainment services to assist users in discovering new items or products that align with their preferences or past interactions.

4. Comparative Analysis

A comparative analysis of User-Based and Item-Based Recommender Systems involves evaluating these two distinct approaches to understand their strengths, weaknesses, and performance in providing recommendations. Here is a structured framework for conducting such a comparative analysis:

4.1 Accuracy Metrics

When conducting an accuracy metrics comparative analysis between User-Based and Item-Based Recommender Systems, several key evaluation metrics can be utilized to assess the performance of each approach. These metrics provide insights into the effectiveness and accuracy of the recommendations generated. Here are some common accuracy metrics and how they can be compared between User-Based and Item-Based systems:

4.1.1 Root Mean Squared Error (RMSE)

- Quantifying the differences between predicted and actual ratings for both methods.
- User-Based CF: Calculate the RMSE to measure the average difference between predicted and actual ratings for user-item pairs in the test dataset.
- Item-Based CF: Similarly, compute the RMSE for item-based recommendations. Compare and contrast RMSE values obtained from both approaches to evaluate prediction accuracy.

4.1.2 Precision, Recall, and F1-Score

- Assessing the effectiveness of recommendations based on user interactions.
- User-Based CF: Evaluate precision (correctly recommended items out of all recommended items), recall (correctly recommended items out of all relevant items), and F1-score (harmonic mean of precision and recall) for user-based recommendations.
- Item-Based CF: Perform the same precision, recall, and F1-score analysis for item-based recommendations. Compare these metrics to assess the accuracy of both systems in terms of relevance and recommendation quality.

4.1.3 Mean Absolute Error (MAE):

- User-Based CF: Calculate the MAE, which measures the average absolute differences between predicted and actual ratings.
- Item-Based CF: Compute the MAE for item-based recommendations and compare the results with user-based MAE values.

4.1.4 Coverage:

- User-Based CF: Evaluate the coverage, which measures the proportion of items for which recommendations can be made.
- Item-Based CF: Compare coverage metrics between user-based and item-based systems to understand the breadth of recommendations each approach can offer.

4.1.5 Hit Rate and Hit Ratio:

- User-Based CF: Assess the hit rate (percentage of successful recommendations) and hit ratio (average number of recommended items per user that were positively received) for user-based recommendations.
- Item-Based CF: Similarly, analyze hit rate and hit ratio for item-based recommendations and compare their effectiveness.

4.1.6 Ranking Metrics (e.g., Mean Reciprocal Rank - MRR, Normalized Discounted Cumulative Gain - NDCG):

- User-Based CF: Compute ranking metrics to evaluate the position of relevant items in the recommendation list.
- Item-Based CF: Compare the performance of ranking metrics in both approaches to assess the quality of top-ranked recommendations.

When comparing accuracy metrics between User-Based and Item-Based Recommender Systems, it's essential to consider the specific context, dataset characteristics, and the nature of the recommendation task to draw meaningful conclusions regarding which approach performs better in terms of accuracy and recommendation quality.

4.2 Scalability and Computational Efficiency

Analyzing the computational demands and scalability considerations for user-based and item-based approaches.

When conducting a scalability and computational efficiency comparative analysis between User-Based and Item-Based Recommender Systems, several factors need consideration to evaluate their performance in handling larger datasets and computational requirements. Here's how you can compare them:

4.2.1 Computational Complexity:

- User-Based CF: Evaluate the computational complexity of calculating user-user similarity measures across the entire user base.
- Item-Based CF: Assess the computational complexity of computing item-item similarity measures for the entire item catalog.

4.2.2 Memory Requirements:

- User-Based CF: Analyze the memory requirements for storing the user-item interaction matrix and similarity matrices for users.
- Item-Based CF: Compare memory usage for item-item interaction matrix and item similarity matrices.

4.2.3 Scalability with Data Size:

- User-Based CF: Evaluate how the computational time and resource requirements scale with an increasing number of users in the system.
- Item-Based CF: Similarly, examine how the computational demands increase as the number of items in the catalog grows.

4.2.4 Efficiency in Sparse Data:

- Assess how User-Based and Item-Based CF methods handle sparse data scenarios where users have interacted with only a small fraction of items.
- Evaluate how each method's computational efficiency is affected by the presence of sparse or incomplete user-item interaction data.

4.2.5 Algorithmic Complexity in Recommendations:

- Analyze the computational complexity of generating recommendations for users based on
- Compare the computational efforts involved in generating personalized recommendations for users in both approaches.

4.2.6 Impact of Optimization Techniques:

- Consider any optimization strategies or algorithms used to improve computational efficiency in either approach (e.g., data pruning, matrix factorization, caching mechanisms).
- Evaluate how these optimization techniques impact the scalability and computational requirements of User-Based and Item-Based CF.

4.2.7 Real-Time Recommendation Speed:

- Measure the time taken to generate recommendations in real-time scenarios for both User-Based and Item-Based systems.
- Assess the efficiency of recommendation generation in response to user queries or interactions.

4.2.8 Parallelization and Distributed Computing:

- Investigate the potential for parallelization or distributed computing to enhance the scalability of either approach.
- Assess the efficiency gains achieved by parallel or distributed implementations in both User-Based and Item-Based CF systems.

By comparing these factors, including computational complexity, memory requirements, scalability with data size, and efficiency in handling sparse data, you can gain insights into the performance and scalability characteristics of User-Based and Item-Based Recommender Systems. These insights will aid in understanding their suitability for large-scale systems and resource-constrained environments.

5. Practical Considerations

5.1 Cold Start Problem

Addressing the challenge of making accurate recommendations for new or sparsely interacting users/items. The cold start problem poses a significant challenge in recommender systems, specifically when dealing with new users or items that have limited or no interaction history within the system. Both User-Based and Item-Based Collaborative Filtering face hurdles in providing accurate recommendations in such scenarios.
User-Based CF: When encountering new users who have not yet provided sufficient ratings or interactions, the user-based approach struggles to identify similar users, limiting the system's ability to offer personalized

recommendations. This lack of historical data for new users impedes the calculation of accurate user similarities.
Item-Based CF: Similarly, when new items are introduced into the system, item-based collaborative filtering encounters difficulties in establishing similarity with existing items. The absence of interaction data for new items restricts the recommendation capabilities based on item similarities, impacting the accuracy of suggestions.
Addressing the cold start problem requires innovative strategies such as hybrid approaches, content-based methods, or utilizing demographic information to overcome the scarcity of user or item data. Employing techniques that incorporate metadata or auxiliary information can aid in making initial recommendations for new users or items, gradually mitigating the cold start issue.

5.2 Serendipity and Diversity

Evaluating the ability of each approach to provide diverse and unexpected recommendations. Serendipity and diversity in recommendations play a crucial role in enhancing user satisfaction and exploration of new items beyond users' existing preferences.
User-Based CF: User-based collaborative filtering tends to offer recommendations based on similar users' preferences, potentially leading to homogeneity in suggestions. While it excels in providing personalized recommendations aligned with a user's known tastes, it might lack diversity in offering unexpected or diverse suggestions.
Item-Based CF: Contrarily, item-based collaborative filtering is inclined towards recommending items similar to those a user has interacted with. This approach might exhibit limited serendipity by suggesting items closely related to a user's past preferences, potentially overlooking diverse or unexpected recommendations.
To address the challenge of limited serendipity and diversity, a balanced approach involving the incorporation of novelty or diversity-enhancing algorithms within both user-based and item-based methodologies is essential. Techniques such as incorporating randomness, exploring long-tail items, or integrating diversity-aware recommendation models can enrich the recommendation landscape, fostering user engagement and satisfaction.

6. Case Studies and Real-world Applications

6.1 E-commerce Platforms

E-commerce platforms heavily rely on recommender systems to personalize product suggestions for users, enhancing their shopping experiences. Comparing User-Based and Item-Based methods in this domain provides valuable insights into their performance in recommending products.
User-Based CF in E-commerce: User-based collaborative filtering excels in suggesting products based on similarities among users with similar purchase histories or preferences. In an e-commerce setting, this method may effectively recommend products based on what similar users have bought or viewed.
Item-Based CF in E-commerce: Item-based collaborative filtering demonstrates proficiency in suggesting products similar to those a user has interacted with previously. By identifying similar products based on user interactions, this method recommends items comparable to those already purchased or viewed.
Assessing the performance of both approaches in an e-commerce setting involves evaluating metrics like conversion rates, click-through rates, and user engagement to determine the effectiveness of product recommendations. These evaluations shed light on which method aligns better with the platform's objectives in providing accurate and personalized product recommendations to users.

6.2 Movie and Music Recommendations

Movie and music recommendation systems extensively use collaborative filtering to suggest entertainment content tailored to individual preferences.

User-Based CF in Entertainment Recommendations: In movie and music recommendation systems, user-based collaborative filtering suggests items based on similar users' preferences. It excels in recommending movies or songs liked by users with similar tastes.

Item-Based CF in Entertainment Recommendations: Item-based collaborative filtering recommends movies or songs similar to those a user has interacted with in the past. It identifies similarities between items and suggests related content based on item-item associations.

Comparing the effectiveness of both methods in movie and music recommendations involves analyzing user engagement metrics like watch time, ratings, or playlist creation. Additionally, exploring diversity and novelty in suggestions can provide insights into which approach introduces users to a wider range of entertainment choices beyond their known preferences.

7. Challenges and Future Directions

7.1 Integration of Deep Learning

Deep learning techniques present promising opportunities to enhance recommendation systems by leveraging neural networks' capabilities for learning complex patterns and representations.
Deep Learning in Recommendation Systems: Integrating deep learning methodologies, such as neural collaborative filtering, deep neural networks, or recurrent neural networks, offers avenues to capture intricate user-item interactions and latent features. These techniques can model non-linear relationships, implicit feedback, and sequential patterns in user behavior more effectively.
Exploring the integration of deep learning in user-based and item-based recommender systems involves investigating architectures that can handle sparse and high-dimensional data, as well as exploring attention mechanisms or sequence-based models for better temporal recommendations.
Understanding the challenges associated with computational resources, model interpretability, and cold start scenarios in deep learning-based recommender systems is crucial. Additionally, comparative studies between traditional collaborative filtering and deep learning models can provide insights into their relative strengths and limitations.

7.2 Context-Aware Recommender Systems

Context-aware recommendation systems aim to incorporate additional contextual information beyond user-item interactions, such as temporal, spatial, or situational data, to improve recommendation accuracy and personalization.
Incorporating Contextual Information: Context-aware recommender systems leverage contextual data, including location, time, device, user demographics, or user intent, to tailor recommendations more precisely. By considering contextual factors, these systems can adapt recommendations to dynamic user preferences in various situations.
Exploring the integration of context-awareness in both user-based and item-based recommender systems involves understanding how different contexts impact recommendation quality. Techniques such as hybrid models, multi-modal learning, or reinforcement learning can be explored to effectively incorporate diverse contextual signals into the recommendation process.
Challenges related to data sparsity, privacy concerns, and feature engineering for context representation need to be addressed when implementing context-aware recommender systems. Additionally, evaluating the trade-offs between increased personalization and potential user intrusiveness is essential for successful adoption.

8. Conclusion

8.1 Key Findings

Through the comparative examination of User-Based and Item-Based Recommender Systems, several key findings have emerged:

- Performance Differences: User-Based CF tends to excel in scenarios with dense user interactions, offering personalized recommendations based on similar users. Meanwhile, Item-Based CF is effective when dealing with a vast item catalog and stable item preferences.
- Accuracy and Scalability: While both approaches demonstrate strengths in providing recommendations, User-Based CF might face challenges in scalability with increasing users, whereas Item-Based CF might struggle with larger item datasets.
- Cold Start and Diversity: Both methods encounter challenges in handling the cold start problem for new users/items. Additionally, maintaining diversity and serendipity in recommendations remains a challenge, particularly in User-Based CF.
- Future Directions: Exploring the integration of deep learning techniques and context-awareness offers promising avenues for enhancing recommendation systems. Deep learning can capture complex patterns, while context-aware systems can tailor recommendations based on situational or temporal factors.

8.2 Practical Implications

The comparative analysis reveals several practical implications for designing and implementing recommendation systems:

- Adaptive Strategies: System designers must adopt adaptive strategies based on the system's characteristics, such as the size of the user or item base, data sparsity, and the need for personalization.
- Hybrid Approaches: Considering hybrid models that combine User-Based and Item-Based CF or integrate other techniques like deep learning or context-awareness might yield more robust and accurate recommendation systems.
- Contextual Considerations: Incorporating contextual information, where applicable, can significantly enhance recommendation accuracy and personalization, addressing the limitations of traditional CF methods.
- Continuous Evaluation: Continuous evaluation of recommendation performance using metrics relevant to the application domain is crucial for optimizing and refining recommendation algorithms.

In conclusion, while User-Based and Item-Based Recommender Systems offer distinct advantages, their effectiveness depends on the specific application domain, data characteristics, and scalability requirements. Integrating innovative techniques and maintaining a balance between accuracy, scalability, and diversity remains a focus for future research and practical implementations in recommendation systems.

Acknowledgments

I would like to express my sincere gratitude to all those who have supported and contributed to this research endeavor. Special thanks to my advisor Dr. Pratik A. Vanjara for their guidance and valuable insights throughout this study. I extend my appreciation to the ATMIYA University, Rajkot for providing the necessary resources and facilities essential for conducting this research. Furthermore, I wish to thank my colleagues and peers for their constructive feedback, discussions, and assistance, which have significantly enriched the content and perspectives presented in this paper. Finally, I extend my appreciation to the academic community and authors whose work served as a foundation for this comparative analysis of user-based and item-based recommender systems.

References

1. Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering.
2. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). Analysis of recommendation algorithms for e-commerce.
3. Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering.
4. Linden, G., Smith, B. J., & York, J. (2003). Amazon.com recommendations: item-to-item collaborative filtering.
5. Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions.
6. Park, S. T., & Chu, W. (2009). Pairwise preference regression for cold-start recommendation.
7. Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model.
8. Bell, R. M., & Koren, Y. (2007). Scalable collaborative filtering with jointly derived neighborhood interpolation weights.
9. Resnick, P., & Varian, H. R. (1997). Recommender systems.
10. Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook.
11. Melville, P., & Sindhwani, V. (2010). Recommender systems.
12. Desrosiers, C., & Karypis, G. (2011). A comprehensive survey of neighborhood-based recommendation methods.
13. Lops, P., de Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends.
14. Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques.
15. Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems.
16. Ekstrand, M. D., Riedl, J. T., & Konstan, J. A. (2011). Collaborative filtering recommender systems.
17. Burke, R. (2002). Hybrid recommender systems: Survey and experiments.
18. Hidasi, B., & Karatzoglou, A. (2018). Recurrent neural networks with top-k gains for session-based recommendations.
19. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering.
20. Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for YouTube recommendations.
21. Wang, H., Wang, N., & Yeung, D. Y. (2015). Collaborative deep learning for recommender systems.
22. Campos, P. G., Fernandez-Luna, J. M., & Huete, J. F. (2014). A comparative study of collaborative filtering algorithms.
23. Konstan, J. A., & Riedl, J. T. (2012). Recommender systems: from algorithms to user experience.
24. Gao, J., Li, Y., Nie, L., & Huang, J. (2015). Multimedia recommendation with implicit feedback.
25. Salakhutdinov, R., & Mnih, A. (2007). Probabilistic matrix factorization.
26. Zhang, Y., Chen, Z., Chen, L., & Zheng, V. W. (2019). A survey of deep learning based recommendation systems.
27. Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems.
28. Shani, G., & Gunawardana, A. (2011). Evaluating recommendation systems.
29. Kang, S., Kim, J., & Kim, H. (2020). A survey on recommender systems based on deep learning.
30. Lee, H., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization.