Journal of Computer Science
https://dergipark.org.tr/en/pub/bbd
Anatolian Science

ISSN,e-ISSN: 2548-1304
Volume:10, Issue:1, pp:53-91, 2025
https://doi.org/10.53070/bbd.1652022
Research Paper

Current Trends in Recommender Systems: A Survey of Approaches and Future Directions

Berke Akkaya*[1]

[1]School of Business Quantitative Methods Department, Istanbul University, Istanbul, Turkey

(berkeakkaya@istanbul.edu.tr)

Received:March. 05, 2025 Accepted:Apr.26, 2025 Published:Jun.1, 2025

Abstract—This paper discusses the growing importance of recommender systems in enhancing user experience and information access in digital environments. It identifies challenges such as data sparsity, the cold-start problem, and scalability, emphasizing the need for advanced machine learning techniques. Various methodologies are explored, including collaborative filtering, content-based filtering, and hybrid approaches. Innovations like graph-based collaborative filtering, graph neural networks, and deep learning are highlighted for addressing data sparsity and complex data relationships. The paper also emphasizes attention mechanisms and sequential modeling to resolve the cold-start problem and adapt to changing user preferences. It stresses the significance of explainable AI for building user trust and transparency. Looking ahead, the paper anticipates advancements in cross-domain recommendation models and the integration of diverse data sources to enhance personalization and relevance. Overall, it advocates for sophisticated methodologies to overcome challenges and improve user satisfaction in digital platforms, underscoring the role of innovation in the future of recommendation technologies.

Keywords : *Recommender Systems, Collaborative Filtering, Content-based Filtering, Hybrid Recommendation Approaches*

1. Introduction

In the contemporary digital landscape, individuals are increasingly exposed to a wide range of information, from e-commerce platforms to social media, from online education portals to streaming platforms. This deluge of information poses a significant challenge for users in navigating and deriving the content that aligns with their needs, a phenomenon that has been termed "information overload" and "decision making bias". In this context, the use of intelligent systems that provide personalized content has become inevitable to increase user satisfaction and facilitate content discovery. Recommendation systems, which are among such intelligent systems, provide personalized recommendations based on users' past interactions and individual preferences, thus facilitating access to content that interests them (Burke, Felfernig, & Göker, 2011). These systems meticulously analyze user behavior, preferences, and past activities to combat information overload and employ sophisticated information filtering techniques to curate content that is relevant to the user's unique needs (Perez-Alcazar, Calderon-Benavides, & Gonzalez-Caro, 2003). The effectiveness of these systems hinges on their ability to identify patterns within user data, enabling them to predict future preferences accurately (Resnick & Varian, 1997). Their goal is to minimize the disparity between predicted and actual ratings, which significantly enhances overall user satisfaction (Lu, Wu, Mao, Wang, & Zhang, 2015).

Despite their advantages, recommender systems face challenges such as the "cold start" problem, which occurs when new users or items lack sufficient historical data, making it difficult to generate accurate recommendations (Lika, Kolomvatsos, & Hadjiefthymiades, 2014). Additionally, scalability issues arise when dealing with high-dimensional datasets, leading to potential inefficiencies (Amer-Yahia, Lakshmanan, Vassilvitski, & Yu, 2009). To overcome these hurdles, advancements in machine learning are crucial. These systems must be developed to adapt to limited data scenarios and efficiently scale as more information becomes available. To overcome the limitations of traditional recommender systems, the integration of advanced machine learning techniques such as deep learning, graph-based structures, and contextual modeling into recommender systems has accelerated in recent years. These advancements have led to significant improvements in accuracy, context sensitivity, multimodal data processing capabilities, and the ability to generate recommendations in real time.

A variety of techniques are employed to make recommendations and to enhance the accuracy and relevance of recommendations. Conventionally, two such techniques are recognized: collaborative filtering, which utilizes the behavior of similar users, and content-based filtering, which focuses on the attributes of products (Gupta, Rao, Bhavsingh, & Srilakshmi, 2023). Furthermore, the employment of hybrid approaches, integrating these two methods, has emerged as a preferred strategy to develop recommendation systems that are more powerful and effective (Sarne, 2015).

Overall, recommender systems are integral to creating personalized experiences across diverse fields, driving both user satisfaction and business success (Huang, Zeng, & Chen, 2007; Wang, Tan, & Zhang, 2010; Fengou, Athanasiou, Mantas, Griva, & Lymberopoulos, 2013; Yanes, Ayman, Ezz, & Almuayqil, 2020). As technology advances, the capabilities and applications of these systems are expected to expand, changing how users interact with platforms (Polatidis & Georgiadis, 2013). The ongoing development of machine learning technologies promises to create more sophisticated recommender systems, further improving user experiences in digital environments (Lu, Wu, Mao, Wang, & Zhang, 2015).

In consideration of the aforementioned factors the aim of this study is to provide a comprehensive review of the traditional, hybrid, and advanced methods employed in recommender systems. By examining the theoretical foundations, advantages, and limitations of these methods, the study proposes future research directions for enhancing the performance, scalability, and explainability of recommender systems. Also, the study provides a comparative summary of empirical results from the literature. By compiling widely used performance metrics such as Precision, Recall, NDCG, MAE, RMSE, and Hit Ratio on various model types, problem domains, and datasets, the study contributes a structured understanding of how recent methods perform in practice.

2. Types of Recommender Systems: Key Challenges and Recent Solutions

2.1. Collaborative Filtering (CF)

Collaborative filtering (CF) is a widely adopted technique in recommender systems that generates personalized suggestions by leveraging the preferences and behaviors of users who share similar interests. It relies on the assumption that users with comparable tastes will continue to express similar preferences in the future (Zhang, Qidong, Chun, Wei, & Huiyi, 2014). CF utilizes a user-item interaction matrix, where rows represent users and columns represent items, with entries indicating the level of interaction or preference. The interaction matrix, denoted as R, represents the interactions between users and items. Mathematically, the interaction matrix R is defined as follows (Zhang, Qidong, Chun, Wei, & Huiyi, 2014):

$$R = \{r_{u,i}\};\ u \in U;\ i \in I$$

Where r_{ui}, represents the interaction between user u and item i where U is the set of all users and I is the set of all items. The matrix is typically sparse, as most users engage with only a limited number of items (Kim & Ahn, 2011). Inputs can include user ratings, clicks, views, and purchase history (Kim & Ahn, 2011). The goal is to predict a user's potential interest in items they haven't interacted with yet (Kim & Ahn, 2011). Collaborative filtering is broadly divided into two methodologies: user-based collaborative filtering and item-based collaborative filtering.

2.1.1. User-based Collaborative Filtering (UBCF)

User-based collaborative filtering (UBCF) is one of the earliest approaches to recommender systems that predicts user preferences by identifying similar users, known as "nearest neighbors" (Verma & Aggarwal, 2020). It assumes that users who agreed on certain items in the past will continue to have similar preferences (Verma & Aggarwal, 2020). UBCF involves identifying users with similar preferences and recommending items they liked that the target user hasn't interacted with (Yun & Youn, 2010). The similarity between users is calculated using interaction matrix where each user is represented as a vector consisting of their interactions with items. Similarity is typically computed through metrics such as Cosine Similarity or Pearson Correlation. Cosine similarity evaluates the similarity between users by measuring the cosine of the angle between their rating vectors. It is computed as follows (Yun & Youn, 2010):

$$Similarity_{cosine}(u,v) = \frac{\sum_{i \in I_{uv}} (r_{u,i} \cdot r_{v,i})}{\sqrt{\sum_{i \in I_{uv}} {r^2}_{u,i}} \cdot \sqrt{\sum_{i \in I_{uv}} {r^2}_{v,i}}}$$

In this formula, I_{uv}, is the set of items co-rated by users u and v, while $r_{u,i}$ and $r_{v,i}$ represent the ratings or purchases of users u and v. Cosine similarity is widely used in applications where ratings are less important than relative similarities (Sarwar et al., 2001). Pearson correlation, on the other hand, accounts for individual rating scales by centering the ratings around their user-specific means. It is computed as follows:

$$Similarity_{pearson}(u,v) = \frac{\sum_{i\in I_{uv}}(r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i\in I_{uv}}(r_{u,i} - \bar{r}_u)^2} * \sqrt{\sum_{i\in I_{uv}}(r_{v,i} - \bar{r}_v)^2}}$$

Here, $\bar{r}_u$ and $\bar{r}_v$ are the average rating of user u and user v. This similarity measure normalizes the ratings, making the system more robust to users with different rating scales (Resnick et al., 1994). Once user similarities are computed, the predicted rating $\widehat{r_{u,i}}$ of user u for an item i can be estimated as a weighted average of the ratings from similar users. The prediction is calculated as follows:

$$\widehat{r_{u,i}} = \bar{r}_u + \frac{\sum_{v\in N_{u,i}} Similarity(u,v) * (r_{v,i} - \bar{r}_v)}{\sum_{v\in N_{u,i}} |Similarity(u,v)|}$$

Here, $N_{u,i}$ is the set of users similar to user u and have rated item i; $r_{v,i}$ is the rating of user v for item i; $\bar{r}_u$ and $\bar{r}_v$ are the average rating of user u and user v; Similarity(u,v) is the similarity score between users u and v.

However, UBCF faces challenges such as scalability, as calculating similarities among many users can be difficult, and sparsity, since most users rate only a few items (Sharma, Gopalani, & Meena, 2017). Additionally, the cold-start problem arises when new users or items lack sufficient data for reliable recommendations (Sharma, Gopalani, & Meena, 2017). Addressing these challenges necessitates developing algorithms that can manage sparse data and enhance recommendation accuracy.

Recent advances in user-based collaborative filtering (UBCF) have sought to address limitations associated with data sparsity, cold start problems and noise in user-item interactions. These enhancements improve the precision and personalization of recommendations through the utilization of novel methodologies, including graph convolutional networks, deep learning and hybrid models. The integration of neighborhood knowledge and dynamic user information enables the capture of more nuanced collaborative inputs, thereby enhancing the precision of user item interaction predictions. Table 1 summarizes recent user-based collaborative filtering (UBCF) studies, addressing challenges like sparsity, cold start, and scalability with innovative techniques. Notable advancements include the utilization of graph-based models, adaptive techniques and enhanced similarity calculations. Also, Table 1 reports the best performance improvement metric along with the corresponding dataset. A comprehensive set of results, including all datasets and evaluation metrics, is provided in the Appendix A1.

The advancements in user-based collaborative filtering between 2019 and 2024 presented in the Table 1 demonstrate a remarkable diversification in methodological approaches, driven by the need to address key issues such as sparsity, cold-start, and scalability. Several studies have successfully achieved considerable performance gains through the integration of deep learning, graph-based modeling, attention mechanisms, and cross-domain information.

In terms of metrics, substantial improvements were observed in recommendation quality. For example, Wu et al. (2020) reported a 197.24% increase in NDCG@20 on the Yelp dataset, showcasing the power of deep neural architectures in learning implicit user-item correlations. Similarly, Zhou et al. (2022) achieved a 158.46% improvement in Recall@10 using trust-aware graph-based methods. Models addressing cold start problems (e.g., Fan et al., 2019; Liu et al., 2020) reduced MAE by up to 26.95%, indicating that context modeling and feedback fusion significantly enhance prediction accuracy in low-data environments.

When compared across methodologies, graph-based approaches (e.g., Wang et al., 2019; Zhou et al., 2022) and deep learning-based techniques (e.g., Ji et al., 2020; Wu et al., 2024) consistently yielded high relative improvements in ranking metrics such as NDCG and Recall. Notably, hybrid strategies and context-aware representations also contributed to consistent yet moderate enhancements in precision and error reduction, demonstrating their robustness in real-world settings.

Furthermore, several studies such as Ma et al. (2022) and Jain et al. (2020) highlighted the value of combining multiple data modalities such as metadata, social signals, or contextual variables resulting in NDCG@5 improvements exceeding 37% or precision gains above 43%, respectively.

Overall, this collection of studies indicates that user-based collaborative filtering can be substantially improved through methodological innovation, particularly when latent representations are enriched via deep neural architectures or structured side information. However, the wide variation in improvement percentages across datasets and metrics also underscores the importance of tailoring the model architecture and input modalities to the specific nature of the recommendation task.

Table 1. Literature Review on UBCF Advancements (2019-2024)

Study	Problem	Method	Solution	Dataset	Metric	Improvement
(Wang, He, Wang, Feng, & Chua, 2019)	Sparsity	Graph-based Collaborative Filtering	Modeling high-order connectivity with graph embeddings	Yelp2018	Recall@20	↑12.89%
(Fan, et al., 2019)	Cold Start Problem	Combining Multiple Data and Feedback	Analyzing social connections with deep learning	Epinions	MAE	↓26.95%
(Zou, et al., 2020)	Scalability	Adaptive Techniques	Optimizing with self-attention networks and Q-learning	MovieLens-1M	Precision@40	↑7.91%
(Joorabloo, Jalili, & Ren, 2020)	Scalability, Cold Start Problem	Enhanced Similarity Calculations	Re-ranking user-item similarities based on trends	Movielens-100K	Precision@20	↑39.68%
(Wu, Wei, Yin, Liu, & Zhang, 2020)	Sparsity	Deep Learning Approaches	Learning user-item correlations with deep learning	Yelp	NDCG@20	↑197.24%
(Ji, Xiang, & Li, 2020)	Sparsity	Deep Learning Approaches	Learning interactions with dual neural networks	MovieLens-1M	NDCG@10	↑77.07%
(Li, et al., 2020)	Sparsity	Dynamic and Contextual User Characteristics	Representing dynamics with neural network layers	LastFM	MRR	↑34.3%
(Ortal & Edahiro, 2020)	Sparsity	Hybrid and Cross-Domain Models	Combining custom similarity with regional statistics	MovieLens-1M	MAE	↓6.17%
(Liu, Jiyong, & Chenggang, 2020)	Cold Start Problem	Dynamic and Contextual User Characteristics	Deriving latent representations with contextual factors	Douban Book	MAP	↑5.39%
(Jain, Nagar, Singh, & Dhar, 2020)	Cold Start Problem, Sparsity	Hybrid and Cross-Domain Models	Active learning with non-linear similarity model	MovieLens-1M	Precision	↑43.1%
(Bae, Kim, Shin, & Kim, 2021)	Sparsity, Scalability	Indirect Neighbor-Based Strategies	Standardizing ratings using neighbor scores	MovieLens-100K	MRR	↑23.81%
(Chen, Xin, Wang, & Ding, 2021)	Sparsity, Cold Start Problem	Enhanced Similarity Calculations	Capturing nuances with factor-level attention	Book-crossing	NDCG@20	↑20.80%
(Zhou, Du, Duan, Ul Haq, & Shah, 2022)	Sparsity	Graph-based Collaborative Filtering	Enhancing with trust-aware deep neural networks	Yelp	Recall@10	↑158.46%
(Zheng, Chen, Du, & Song, 2022)	Scalability	Combining Multiple Data and Feedback	Graph CF with homogeneous and heterogeneous feedback	MovieLens 1M	LogLoss	↓8.38%
(Ma, Pan, & Ming, 2022)	Sparsity	Combining Multiple Data and Feedback	Structured CF with heterogeneous feedback	Alibaba2015	NDCG@5	↑37.18%
(Wang, Chen, Xi, Huang, & Xie, 2022)	Sparsity, Cold Start Problem	Hybrid and Cross-Domain Models	Acquiring cross-domain knowledge with deep learning	Amazon Phone	NDCG@10	↑29.3%
(Mahesh, Kumar, & Lim, 2023)	Sparsity	Enhanced Similarity Calculations	Incorporating user confidence and temporal factors	Movielens-100K	MAE	↑1.33%
(Gaiger, Barkan, Tsipory-Samuel, & Koenigstein, 2023)	Sparsity, Scalability	Dynamic and Contextual User Characteristics	Adapting representations with context-target attention	Yahoo! Music	HR@10	↑170.5%
(Wu, Xia, Min, & Xia, 2024)	Scalability	Deep Learning Approaches	Analyzing past interactions with attention mechanisms	MovieLens 1M	NDCG@10	↑4.63%
(Gong, Song, Li, & Wang, 2024)	Sparsity	Indirect Neighbor-Based Strategies	Enhancing propagation with high-order GCN	Amazon Books	NDCG@20	↑63.5%

2.1.2. Item-based Collaborative Filtering (IBCF)

Item-based collaborative filtering (IBCF) is a technique in recommender systems that focuses on relationships between items rather than user preferences. Unlike UBCF, which identifies similar users, IBCF recommends items based on their similarity to those that the target user has previously liked or interacted with (Sarwar, Karypis, Konstan, & Riedl, 2001). IBCF is particularly useful in scenarios with limited user data or systems with many items, as it can generate recommendations without requiring detailed user profiles. The system first computes item-to-item similarity scores, then predicts the user's preferences by leveraging their historical interactions with similar items (Pirasteh, Jung, & Hwang, 2014). The underlying assumption behind IBCF is that items co-rated similarly by users tend to be similar. As in UBCF, IBCF also relies on the interaction matrix *R*. However, in this case, items are treated as vectors rather than users. Similarity scores between items are computed using metrics such as Pearson correlation or cosine similarity (Sarwar, Karypis, Konstan, & Riedl, 2001). Cosine similarity measures the similarity between two items by computing the cosine of the angle between their rating vectors. It is calculated as:

$$Similarity_{cosine}(i,j) = \frac{\sum_{u \in U_{ij}} (r_{u,i} \cdot r_{u,j})}{\sqrt{\sum_{u \in U_{ij}} {r^2}_{u,i}} \cdot \sqrt{\sum_{u \in U_{ij}} {r^2}_{u,j}}}$$

Here, U_{ij}, is the set of users who rated both items *i* and j, while $r_{u,i}$ and $r_{u,j}$ denote the ratings or purchases of users *u* for items *i* and j, respectively. The other commonly used similarity metric is Pearson correlation, which is computed between items as follows:

$$Similarity_{pearson}(i,j) = \frac{\sum_{u \in U_{ij}} (r_{u,i} - \bar{r}_u)(r_{u,j} - \bar{r}_u)}{\sqrt{\sum_{u \in U_{ij}} (r_{u,i} - \bar{r}_u)^2} * \sqrt{\sum_{u \in U_{ij}} (r_{u,j} - \bar{r}_u)^2}}$$

Here, $\bar{r}_u$ is the average rating given by user u. Once item similarities are calculated, the predicted rating $\widehat{r_{u,i}}$ for a user *u* on a new item *i* is computed using a weighted average of the ratings of similar items that the user has previously rated:

$$\widehat{r_{u,i}} = \frac{\sum_{j \in S_{i,u}} Similarity(i,j) * r_{u,j}}{\sum_{j \in S_{i,u}} |Similarity(i,j)|}$$

Here, $S_{i,u}$ is the set of items similar to item *i* that user *u* has rated; $r_{u,j}$ is the rating of user u on item j and $Similarity(i,j)$ is the similarity score between items *i* and *j*. Unlike UBCF, this formula does not include rating normalization through user-specific means. This is because IBCF relies solely on the target user's own ratings, and not on the ratings of other users. Therefore, normalization is less critical unless user rating bias is explicitly present. In contrast, UBCF must adjust for user-specific rating scales by centering around the mean.

Item-based CF (IBCF) has several advantages over user-based CF (UBCF), including enhanced scalability, stability, and computational simplicity (Pirasteh, Jung, & Hwang, 2014). It is more scalable because the number of items is usually smaller than the number of users, making item similarity calculations less resource intensive. Item relationships are also more stable over time compared to user preferences, which can fluctuate. Additionally, item-based CF allows for faster real-time recommendations due to the pre-computation and storage of item similarities (Seng, Chen, & Zhang, 2020). However, challenges such as the cold start problem for new items, difficulties with extremely sparse datasets, and popularity bias can affect its effectiveness (Abdollahpouri, 2019). Popular items may be recommended more frequently, limiting the diversity of recommendations and the ability to suggest niche items.

Current research to address challenges in item-based collaborative filtering (IBCF) focuses on tackling scenarios involving complex user-item interactions and dealing with data sparsity issues. Table 2 summarizes item-based collaborative filtering (IBCF) studies, focusing on solutions to sparsity, cold start, and popularity bias using a range of methods such as graph convolutional networks, deep learning and variational auto-encoders to improve the accuracy and effectiveness of recommendations. Also, Table 2 reports the best performance improvement metric along with the corresponding dataset. A comprehensive set of results, including all datasets and evaluation metrics, is provided in the Appendix A2.

The recent developments in item-based collaborative filtering (IBCF) from 2018 to 2024 presented in Table 2 reflect a substantial diversification in algorithmic strategies aimed at mitigating classical limitations such as sparsity, popularity bias, and the cold start problem. As shown in Table 2, a significant number of approaches have shifted from traditional similarity-based heuristics to deep learning architectures and graph-based representations, particularly Graph Convolutional Networks (GCNs), which dominate post-2019 literature.

The performance improvements reported across diverse datasets highlight the efficacy of these techniques. For instance, Zhu et al. (2020) report an outstanding 126.0% improvement in NDCG@10 by combining item- and social-based signals on Yelp. Similarly, Walker et al. (2022) achieve a 191.1% gain in NDCG@20 using a Variational Autoencoder, showing that generative models can robustly address data sparsity. Choi et al. (2021)

and Wang et al. (2019) show that GCN-based models consistently perform well, with recall improvements exceeding 50%, even in cold start conditions.

Table 2. Literature Review on IBCF Advancements (2018-2024)

Study	Problem	Method	Solution	Dataset	Metric	Improvement
(Dacrema, Gasparin, & Cremonesi, 2018)	Cold Start Problem, Popularity Bias	Graph-based Collaborative Filtering	Integrating collaborative similarity into content	Netflix	MAP	↑47.34%
(Shams & Haratizadeh, 2018)	Popularity Bias, Diversity of Recommendations	Deep Learning Approaches	Personalized item ranking with neural networks	MovieLens-100K	NDCG@5	↑9.68%
(da Costa, Manzato, & Campello, 2019)	Sparsity, Popularity Bias	Deep Learning Approaches	Ensemble of multiple recommender systems	FilmTrust	F1	↑9.87%
(Xue, et al., 2019)	Sparsity, Cold Start Problem	Deep Learning Approaches	Capturing high-level relations from past interactions	MovieLens-1M	NDCG@10	↑48.55%
(Wang, et al., 2019)	Sparsity, Cold Start Problem	Graph Convolutional Networks	Improving quality with GAN-based data augmentation	Amazon Electronics	HR@20	↑50.0%
(Zeng, Lin, Li, Pan, & Ming, 2019)	Cold Start Problem, Popularity Bias	Deep Learning Approaches	Optimizing with time-sensitive bidirectional similarity	MovieLens-10M	NDCG@5	↑60.0%
(Xin, He, Zhang, Zhang, & Jose, 2019)	Sparsity, Diversity of Recommendations	Graph Convolutional Networks	Predicting preferences with two-stage attention	KKBox	MRR@10	↑40.7%
(Nikolakopoulos & Karypis, 2020)	Sparsity, Popularity Bias	Random Walks and Similarity Prediction	Enhancing diversity with random walk methods	Amazon Electronics	NDCG@10	↑57.0%
(Wang, Wang, Shi, Song, & Li, 2020)	Sparsity, Popularity Bias	Graph Convolutional Networks	Optimizing with multi-component GCN in user-item graph	Yelp	MAE	↓34.5%
(Zhu, Liu, & Chen, 2020)	Popularity Bias, Sparsity	Deep Learning Approaches	Combining item and social filtering models	Yelp	NDCG@10	↑126.0%
(Choi, Jeon, & Park, 2021)	Cold Start Problem, Sparsity	Graph Convolutional Networks	Extending GCN with learnable differential equations	Amazon Books	Recall@10	↑76.8%
(Walker, Zhou, Baagyere, Ahene, & Zhang, 2022)	Sparsity, Popularity Bias	Variational Autoencoders	Encoding hard-to-detect feedback with autoencoders	MovieLens-10M	NDCG @20	↑191.1%
(Liang, Junhao, & Zhou, 2022)	Popularity Bias, Diversity of Recommendations	Deep Learning Approaches	Modeling user diversity preferences with neural nets	MovieLens-1M	NDCG @10	↑14.34%
(Li, Liu, & Yang)	Cold Start Problem, Sparsity	Graph Convolutional Networks	Enhancing accuracy with item attribute-aware GCN	Amazon	Precision@10	↑64.6%

Models addressing popularity bias (e.g., Nikolakopoulos & Karypis, 2020) often employ random walks or ensemble-based strategies, demonstrating the necessity of diversification in neighbor exploration. Deep learning models (e.g., Liang et al., 2022; da Costa et al., 2019) further contribute to personalization through adaptive learning, with NDCG@10 and F1 improvements often exceeding 10%.

Overall, IBCF has been significantly enhanced through the adoption of deep neural networks, graph-based reasoning, and data augmentation strategies. These developments demonstrate that item-based recommendations are not limited to static similarity measures but can leverage contextual, structural, and user-behavioral signals to achieve substantial performance gains.

2.2. Content-based Filtering (CBF)

Content-based filtering in recommender systems suggests items to users based on the features of items they have previously liked, distinguishing it from collaborative filtering which relies on user interactions (Van Meteren & Van Someren). This method builds a user profile reflecting the characteristics of items the user has engaged with, aiming to recommend similar items (Pazzani & Billsus, 2007). It is commonly used in domains like film, music, and documents, where items can be categorized by attributes such as keywords (Wan, Rubens, Okamoto, & Feng, 2015). At the heart of CBF lies the idea that both users and items can be represented in the same feature space. Formally, an item *i* is modeled as a feature vector:

$$\vec{\imath} = [x_1, x_2, \ldots, x_n]$$

Where each component x_k represents a content-related attribute. Similarly, a user's profile is constructed as a vector $\vec{u}$, which aggregates the features of items the user has interacted with positively:

$$\vec{u} = \frac{1}{|I_u|} \sum_{i \epsilon I_u} \vec{\imath}$$

Here, I_u is the set of items that user *u* has rated or consumed, and the aggregation may involve weighting schemes such as TF-IDF to emphasize more informative features (Nath & Ahmad, 2022). Once the user profile vector $\vec{u}$ and the item vector $\vec{\imath}$ are defined in the same feature space, the user's interest in a new item can be predicted by calculating the similarity between these vectors. One of the most widely used methods for this purpose is cosine similarity, which measures the angle between the user and item vectors (Pazzani & Billsus, 2007):

$$\widehat{r_{u,\imath}} = similarity(\vec{u}, \vec{\imath}) = \frac{\vec{u} * \vec{\imath}}{||\vec{u}|| * ||\vec{\imath}||}$$

Where; $\vec{u} * \vec{\imath}$ is the dot product between user and item vectors, $||\vec{u}|| * ||\vec{\imath}||$ are the Euclidean norms of the vectors and $\widehat{r_{u,\imath}}$ is the predicted preference score of user *u* for item *i*. This similarity score reflects how close the new item's attributes are to the user's historical preferences. Higher values indicate a stronger match and a higher likelihood of user interest. CBF operates under the assumption of user preference consistency that users tend to prefer items with attributes similar to those they previously liked. This forms the theoretical backbone of the method. The typical workflow involves (Pazzani & Billsus, 2007):

1. Item Representation: Extract feature vectors for each item.
2. User Profile Construction: Combine feature vectors of positively rated items.
3. Similarity Computation: Calculate similarity between the user vector and all item vectors.
4. Ranking and Recommendation: Rank items based on similarity scores and recommend the top-N items.

The main difference between collaborative filtering (CF) and content-based filtering (CBF) is the type of data used for recommendations. CBF relies on users' past behaviors and preferences, identifying similar users and recommending products based on shared preferences (Basilico & Hofmann, 2004). It also considers the characteristics of content associated with previously preferred items to derive new recommendations (Basilico & Hofmann, 2004). CBF offers advantages like customized recommendations and addressing the cold start problem for items since it doesn't rely on user ratings (Nath & Ahmad, 2022). However, it faces challenges such as overspecialization, where recommendations become too similar and limit diversity (Nath & Ahmad, 2022). The effectiveness of CBF depends on well-defined item attributes, and it struggles with new users who have limited interactions or poorly defined attributes, leading to a cold start problem for users (Nath & Ahmad, 2022).

Table 3 summarizes recent studies in the domain of content-based filtering (CBF). These works primarily aim to address the inherent limitations of traditional CBF techniques, such as poorly defined features, data sparsity, cold start, and overspecialization, by integrating deep learning models, hybrid frameworks, temporal dynamics, and context-aware embeddings. As seen in the studies, the convergence of collaborative signals with content features and advanced neural models has proven to enhance recommendation quality and personalization in various application domains. Also, Table 3 reports the best performance improvement metric along with the corresponding dataset. A comprehensive set of results, including all datasets and evaluation metrics, is provided in the Appendix A3.

The developments in content-based filtering (CBF) between 2018 and 2024, as illustrated in Table 3, reflect a significant methodological evolution driven by the need to mitigate problems such as cold start, overspecialization, and poorly defined item features. Unlike collaborative methods, CBF focuses on analyzing item attributes and user profiles directly, which makes it inherently robust against user interaction sparsity. However, it historically struggled with adaptability and contextual relevance.

Table 3. Literature Review on CBF Advancements (2018-2024)

Study	Problem	Method	Solution	Dataset	Metric	Improvement
(Dong, Zheng, Zhang, & Wang, 2018)	Poorly Defined Features	Repetitive Cross-Domain Approaches	Combining RNN with matrix factorization for sequential modeling	Netflix	NDCG@10	↑29.61
(Hu G., 2019)	Poorly Defined Features	Content-Aware Collaborative Filtering	Integrating user-item interactions with textual data	Amazon	MRR@10	↑171.6%
(Song, Li, Jiang, Qin, & Liao, 2019)	Cold Start Problem	Hybrid and Temporal Models	Combining CF and CBF with temporal context	CiteULike	NDCG@100	↑2095.0%
(Wu, Li, Hsieh, & Sharpnack, 2020)	Cold Start Problem	Personalized Transformer Models	Ranking items with personalized transformers	MovieLens-1M	NDCG@10	↑164.19%
(Sivaramakrishnan, Subramaniyaswamy, Viloria, Vijayakumar, & Senthilselvan, 2021)	Cold Start Problem	Content-Aware Collaborative Filtering	Two-stage hybrid deep learning with Bayesian autoencoders	Amazon-Books	Precision@10	↑219.35%
(Mazeh & Shmueli, 2020)	Poorly Defined Features	Deep Social Collaborative Filtering	Privacy-preserving architecture with personal data stores	MovieLens-10M	Precision@10	↑12.62%
(Hansen, Hansen, Simonsen, Alstrup, & Lioma, 2020)	Cold Start Problem	Content-Aware Collaborative Filtering	Generating binary hash codes with autoencoders	Yelp	MRR@10	↑2.70%
(Sharma, Rana, & Malhotra, 2022)	Cold Start Problem	Hybrid and Temporal Models	Combining CF and CBF for improved recommendations	Books Dataset	MAE	↓5.04%
(Channarong, Paosirikul, Maneeroj, & Takasu, 2022)	Cold Start Problem	Hybrid and Temporal Models	Combining CBF and CF with BERT-based deep learning	MovieLens-1M	NDCG@10	↑41.96%
(Wang, Han, Qian, Xia, & Li, 2022)	Overspecialization	Metadata-Based Recommendations	Probabilistic propagation with temporal multi-dimensional graphs	LastFM	F1@10	↑5436.45%
(Magron & Févotte, 2022)	Cold Start Problem	Content-Aware Collaborative Filtering	Extracting content info with deep learning for music	Million Song	NDCG@10	↑38.41%
(Chen & Huang, 2024)	Overspecialization	Hybrid and Temporal Models	Analyzing behavior and item features with algorithmic techniques	Unshared	Precision@10	↑30.95%

Recent studies have addressed these weaknesses through the integration of deep learning, natural language processing (NLP), and hybrid modeling. For instance, Hu (2019) achieved a remarkable 171.6% improvement in MRR@10 on Amazon by incorporating textual user-item interactions. Similarly, Song et al. (2019) reported an exceptionally high 2095% improvement in NDCG@100 by combining collaborative and content-based filtering with temporal data.

The increasing adoption of Transformer-based architectures is also noteworthy. Wu et al. (2020) implemented personalized transformers to capture nuanced item ranking dynamics, leading to a 164.19% increase in NDCG@10 on MovieLens-1M. Other studies focused on overspecialization, such as Wang et al. (2022), achieved striking performance gains (F1@10 ↑5436.45%) by modeling temporal and multi-dimensional item graphs.

Hybrid strategies continue to dominate content-based advances. Channarong et al. (2022) integrated CBF with collaborative filtering using a BERT-based model, improving NDCG@10 by 41.96%. In the domain of privacy, Mazeh & Shmueli (2020) proposed a privacy-preserving architecture using personal data stores, showcasing how ethical design can coexist with improved Precision@10 (↑12.62%).

Overall, the diversity of techniques ranging from Bayesian autoencoders to temporal graph propagation demonstrates how CBF has evolved into a highly flexible paradigm. These innovations emphasize personalization, contextual adaptation, and semantic understanding, addressing longstanding challenges and dramatically improving performance in complex recommendation settings.

2.3. Hybrid Methods

Hybrid recommender systems have emerged as a robust solution to overcome the inherent limitations of traditional recommendation approaches. These systems combine various techniques to enhance the quality, diversity, and robustness of recommendations (Jannach, Zanker, Felfernig, & Friedrich, 2010). They utilize content-based information to provide initial suggestions despite limited user interaction data and introduce diverse items to counteract over-specialization (Jannach, Zanker, Felfernig, & Friedrich, 2010). Empirical evidence shows that hybrid methods outperform traditional techniques in accuracy and user satisfaction, creating a dynamic user experience by refining recommendations as user engagement increases (Cano & Morisio, 2017). There is various hybridization strategies designed to integrate multiple recommendation models within a unified framework. One of the most commonly used strategies is weighted hybridization, in which recommendation scores from different methods are linearly combined to produce a final prediction score. This can be represented as follows:

$$\widehat{r_{u,i}} = \alpha \cdot \widehat{r_{u,i}^{CF}} + (1-\alpha) \cdot \widehat{r_{u,i}^{CBF}}$$

Here, $\widehat{r_{u,i}^{CF}}$ denotes the predicted score obtained from CF, $\widehat{r_{u,i}^{CBF}}$ represents the CBF prediction, and α is a tunable parameter within the range $0 \leq \alpha \leq 1$, determining the relative contribution of each component. In contrast, switching hybridization selects the most appropriate recommendation technique depending on specific contextual factors. For instance, CBF may be used in cold-start scenarios, while CF can be applied once sufficient user-item interaction data is accumulated. Another strategy, feature augmentation, involves using the output of one method to enrich the input features of another. A typical example is incorporating content-based user profiles into the latent factor model of CF, thereby enhancing the representational capacity of collaborative models. Lastly, cascade models employ a two-step approach where one technique generates an initial candidate list of recommendations, which is subsequently re-ranked or refined by a second model for improved accuracy.

The motivation behind hybridization is to enhance recommendation accuracy and personalization while addressing challenges like data sparsity, cold start issues, scalability, and over-specialization (Jannach, Zanker, Felfernig, & Friedrich, 2010). Despite their advantages, hybrid systems are more complex to implement due to the need for integrating multiple algorithms and balancing their parameters, which requires substantial data volumes (Jannach, Zanker, Felfernig, & Friedrich, 2010). Successfully merging CF and CBF demands comprehensive user interaction data and detailed item attributes, posing challenges in poorly defined domains, while achieving a balance between different methodologies is crucial for maintaining recommendation accuracy (Cano & Morisio, 2017).

Recent advancements in hybrid recommender systems have focused on overcoming key limitations, including data sparsity, the cold-start problem, user engagement challenges, and interpretability needs. These advancements employ diverse methodologies, including deep learning, autoencoders and attention mechanisms, to enhance the effectiveness and applicability of hybrid recommender systems. The following Table 4 summarizes studies on hybrid recommender systems, addressing challenges including sparsity, cold start, and scalability through the integration of collaborative and content-based techniques. Also, Table 4 reports the best performance improvement metric along with the corresponding dataset. A comprehensive set of results, including all datasets and evaluation metrics, is provided in the Appendix A4.

Several studies showcase remarkable advances in dealing with sparsity and noise by employing variational autoencoders (VAEs). For instance, Zhang et al. (2021) achieved an outstanding Recall@20 improvement of 5093.98% on the Tmall dataset by integrating Bi-LSTM, GRU, and attention into VAE frameworks. Similarly, Tanuma and Matsui (2022) combined VAE with Poisson factorization, leading to a 4.77% reduction in mean squared error (MSE) on the Million Song dataset.

In the domain of explainability and dynamic user behavior, hybrid approaches combining session-based modeling and gradient boosting have been particularly effective. Bauer and Jannach (2024) demonstrated a 67.8% improvement in Recall@1 by modeling short-term session interactions using feature-based architectures.

Attention-based mechanisms also proved to be highly effective in sequential and contextual recommendations. Zhu and Chen (2021) reported a 72.58% improvement in NDCG@100 by encoding item attributes with a variational bandwidth autoencoder. Hu et al. (2024) leveraged past behaviors using an attention-based model, achieving a 71.25% increase in NDCG@10 on the XMarket dataset.

Additionally, cross-domain hybrid models addressing scalability and cold start challenges showed substantial gains. Ibrahim et al. (2023) achieved a 98.11% reduction in RMSE in cross-domain settings by integrating hierarchical attention into deep learning frameworks. Furthermore, Gatti et al. (2023) highlighted the impact of combining deep autoencoders with social graphs, resulting in a 287.5% improvement in Precision@10 for the WikiArt dataset.

Overall, these findings highlight the growing complexity and versatility of hybrid recommender systems. By incorporating multiple feedback types, domain information, and dynamic contextual signals, hybrid models significantly enhance recommendation accuracy, robustness, and interpretability across diverse application environments.

Table 4. Literature Review on Hybrid Advancements (2019-2024)

Study	Problem	Method	Solution	Dataset	Metric	Improvement
(Huang, et al., 2019)	Scalability	Attention Mechanisms in Sequential Recommendations	Combining deep learning with traditional ML	Amazon	HR@10	↑15.63%
(Duong, Vuong, Nguyen, & Dang, 2020)	Sparsity, Noise	Item Representations with Autoencoder	Generating variational representations with 3-layer autoencoder	MovieLens 20M	MAE	↓4.36%
(Dong, Zhu, Li, & Wu, 2020)	Cold Start Problem, Noise	Explainability and User Interaction	Combining matrix factorization with dual autoencoders	Movie Tweetings 10K	MAE	↓44.85%
(Zhang, Wong, & Chu, 2021)	Sparsity, Cold Start Problem	Variational Autoencoders	Encoding user-item info into latent space	CiteULike	NDCG	↑4.67%
(Zhang, Wang, Li, Xiao , & Shi, 2021)	Dynamic User Preferences	Variational Autoencoders	Integrating Bi-LSTM and GRU with attention	Tmall	Recall@20	↑5093.98%
(Tanuma & Matsui, 2022)	Noise, Sparsity	Variational Autoencoders	Combining VAE with Poisson factorization	Million Song	MSE	↓4.77%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	YelpZip	Precision	↑24.73%
(Zhu & Chen, 2021)	Noise, Sparsity	Attention Mechanisms in Sequential Recommendations	Encoding attributes with variational bandwidth autoencoder	Amazon Toys & Games	NDCG@100	↑72.58%
(Huang, et al., 2023)	Diversity of Recommendations	Topic Embedding and Contextual Information	Fusing user preferences with timeliness effects	Adressa	MRR	↑88.53%
(Ibrahim, Bajwa, Sarwar, Hajjej, & Sakr, 2023)	Cold Start Problem, Scalability	Cross-Domain Recommendations	Integrating hierarchical attention with deep modeling	Yelp 2013	RMSE	↓98.11%
(Bauer & Jannach, 2024)	Dynamic User Preferences, Cold Start Problem	Explainability and User Interaction	Combining session-based modeling with gradient boosting	Diginetica	Recall @1	↑67.8%
(Gatti, Diaz-Pace, & Schiaffino, 2023)	Multiple Data Integration, Personalization	Singular Value Parsing Extensions	Enhancing precision with deep autoencoders and social graphs	WikiArt	Precision@10	↑287.5%
(Kumar, Sharma, Sharma, Herencsar, & Srivastava, 2023)	Cold Start Problem, Sparsity	Dual Autoencoder for Collaborative Proposals	Combining social matrix factorization with link probabilities	Last.fm	Recall@50	↑22.73%
(Hu, Nakagawa, Cai, Ren, & Deng, 2024)	Dynamic User Preferences	Attention Mechanisms in Sequential Recommendations	Leveraging past behaviors with attention-based model	XMarket IN	NDCG@10	↑71.25%

2.4. Methodological Contribution

This section not only provides a theoretical overview but also delivers a comprehensive comparative analysis of recent empirical advancements in recommender systems. By systematically categorizing methodologies including graph-based, deep learning-based, hybrid, and context-aware approaches; this review elucidates the practical strengths of each method in solving prominent recommendation challenges, such as sparsity, cold start, scalability, and contextualization. Through detailed examination of recent literature from 2018 to 2024, empirical performance improvements across diverse datasets and widely recognized evaluation metrics (e.g., Precision, Recall, NDCG, MAE, RMSE, Hit Ratio) have been analyzed.

By clearly defining and consistently applying these metrics, this study provides clarity on how each methodological innovation contributes to overcoming specific challenges within recommendation scenarios. The following subsections first summarize which types of methodologies effectively address particular recommendation problems, followed by a detailed comparative analysis of the relative performance improvements observed. Finally, visual representations of these comparative analyses (Boxplot distributions and model-type performance comparisons) are provided to facilitate a clearer understanding of methodological efficacy.

2.4.1. Datasets Overview and Their Roles in Performance Evaluation

In the comparative review, a diverse range of datasets were utilized to evaluate methodological advancements across different recommender system challenges. These datasets vary considerably in terms of sparsity, domain, user behavior complexity, and interaction types, thus providing a robust ground for assessing model effectiveness under different conditions. The key datasets used in the literature and their corresponding characteristics are summarized in Table 5.

Table 5. Dataset Characteristics and Usage in Literature

Dataset	**Domain**	**Challenges Addressed**	**Typical Usage**
MovieLens (100K, 1M, 10M, 20M)	Movies	Sparsity, Cold-start	CF, Deep Learning, Hybrid Methods
Amazon (various)	E-commerce	Sparsity, Metadata Integration	Content-Aware, Graph-based, Hybrid
Yelp (2018)	Services, Restaurants	Sparsity, Context	Context-aware, Graph Neural Networks
CiteULike	Research Papers	Cold-start, Content Sparsity	Content-based, Hybrid
Netflix	Movies	Scalability, Sparsity	Graph-based, CF
Pinterest, KKBox	Multimedia, Music	Sequence Dynamics	Sequence-aware, Attention Mechanisms
LastFM	Music	Cold-start, Temporal Context	Deep Learning, Contextual Models
Goodreads, Book-Crossing, Epinions, CiaoDVDs	Books & Reviews	Cold-start, Social Influence	Hybrid, Social CF
Tmall, Diginetica, Adressa, XMarket	E-commerce, News	Dynamic Preferences	Session-based, Sequential Models
Million Song, WikiArt	Music, Art	Content-awareness, Cold-start	Content-based Filtering

The MovieLens datasets (including 100K, 1M, 10M, and 20M versions) are among the most widely used benchmarks for recommender systems, consisting of movie ratings collected from users (Harper & Konstan, 2015). These datasets primarily challenge models with sparsity issues and cold-start scenarios, providing varying density levels across different versions. They serve as critical testbeds for collaborative filtering (CF), deep learning, and hybrid recommendation methods.

Building upon these traditional movie-focused benchmarks, the Amazon datasets (e.g., Electronics, Books, Fashion, Digital Music) offer real-world e-commerce environments characterized by highly sparse interaction matrices and rich textual metadata (McAuley et al., 2015). They are essential for evaluating the ability of recommender models to handle sparsity, integrate auxiliary side information, and manage diverse user-item relationships through content-aware and hybrid techniques.

Similarly, the Yelp datasets, particularly Yelp2018, contain user-generated reviews and ratings for local businesses and services (He & McAuley, 2016). With their sparsity and abundance of textual and contextual information, Yelp datasets are ideal for testing graph-based and context-aware recommendation models.

In the domain of academic content, CiteULike focuses on research article recommendations and is characterized by challenges related to cold-start problems and content sparsity (Wang et al., 2011). It has been extensively used to evaluate hybrid models and content-aware collaborative filtering techniques that require efficient handling of limited interaction histories.

Moreover, the Netflix Prize dataset introduced a large-scale recommendation challenge that initiated significant advancements in collaborative filtering research (Bennett & Lanning, 2007). It remains a standard benchmark for testing the scalability, generalization, and robustness of recommender models under extreme sparsity conditions.

For sequence-aware modeling, datasets like Pinterest and KKBox capture user behavior sequences such as pinning activities and music listening habits, respectively (Zhao et al., 2019; Ying et al., 2018). They are particularly suited for models that exploit temporal dynamics and evolving user preferences.

The LastFM dataset, enriched with user listening histories, timestamps, and social tags (Celma, 2010), supports research on integrating social influences, contextual temporal information, and sequence modeling into recommendation tasks.

Furthermore, datasets such as Goodreads, Book-Crossing, Epinions, and CiaoDVDs, primarily centered around book reviews and product ratings, facilitate the exploration of cold-start solutions, social influence modeling, and sparsity mitigation strategies (Ziegler et al., 2005; McAuley & Leskovec, 2013).

In e-commerce and news domains, datasets such as Tmall, Diginetica, Adressa, and XMarketIN emphasize session-based behavior, dynamic user preferences, and sequence modeling (Ludewig et al., 2018; Gulla et al., 2017). They are commonly used to evaluate models that require handling rapidly evolving user intents and session-level personalization.

Finally, the Million Song Dataset and WikiArt dataset contribute to recommendation research by focusing on content-rich domains such as music and visual arts (Bertin-Mahieux et al., 2011; Saleh & Elgammal, 2015). These datasets are instrumental for testing content-based filtering models, especially under cold-start and metadata-driven recommendation scenarios.

2.4.2. Metric-based Performance Improvements

When evaluating the effectiveness of methodological advancements in recommender systems, a set of widely adopted evaluation metrics were employed. These metrics measure various aspects of system performance, including ranking quality, prediction accuracy, and user engagement. Table 6 summarizes the major performance metrics utilized in the evaluation of recommender system methodologies, alongside their typical improvement ranges and the key observations associated with each metric category.

Table 6. Performance Metrics, Improvement Ranges, and Observed Trends Across Methodologies

Metric Type	Common Metrics	Improvement Range	Notes
Ranking Metrics	NDCG, Recall, Precision, HR	50% – 200%	Major gains in deep learning and graph-based methods
Error Metrics	MAE, RMSE	5% – 35%	Moderate improvements, critical for score prediction
Engagement Metrics	F1-score, MRR	20% – 100%	Better early ranking and balanced retrieval

Ranking-based metrics assess how well a recommender system orders relevant items ahead of irrelevant ones. They are crucial for evaluating user satisfaction in top N recommendation settings. Normalized Discounted Cumulative Gain (NDCG measures the ranking quality by penalizing relevant items that appear lower in the recommendation list. It is calculated as follows (Järvelin & Kekäläinen, 2002):

$$NDCG@k = \frac{DCGk}{IDCGk}, \quad DCG@k = \sum_{i=1}^{k} \frac{rel_i}{log_2(i+1)}$$

where rel_i is the relevance of the item at position *i*, and *IDCG* is the ideal *DCG*.

Recall measures the proportion of all relevant items that are successfully retrieved among the top-k recommendations (Herlocker et al., 2004):

$$Recall@k = \frac{\text{Number of relevant items retrieved}}{\text{Total number of relevant items}}$$

Precision evaluates the proportion of recommended items that are actually relevant among the top-k results (Herlocker et al., 2004):

$$Precision@k = \frac{\text{Number of relevant items retrieved}}{k}$$

Hit Ratio checks whether at least one of the relevant items is among the top-k recommendations (Zhang et al., 2019), serving as a binary indicator of success:

$$HR@k = \begin{Bmatrix} 1, & if\ at\ least\ one\ relevant\ item\ is\ in\ top - k \\ 0, & otherwise \end{Bmatrix}$$

These metrics typically demonstrated the largest improvements in recent studies, often exceeding 50%–200%, especially for graph-based and deep learning models.

Error-based metrics evaluate the accuracy of rating or score predictions by measuring the deviation between predicted and actual values. Mean Absolute Error (MAE) calculates the average absolute difference between predicted and true ratings (Herlocker et al., 2004):

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{r_i} - r_i|$$

where $\hat{r_i}$ and r_i are the predicted and true ratings, respectively.

Root Mean Square Error (RMSE) computes the square root of the average squared differences between predictions and actual ratings (Herlocker et al., 2004):

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{r_i} - r_i)^2}$$

Improvements in MAE and RMSE were moderate but important, generally ranging from 5% to 35%, reflecting better predictive precision, particularly in hybrid and context-aware models.

Engagement and diversity metrics capture user interaction quality, balancing relevance and diversity. F1-score is the harmonic mean of Precision and Recall, balancing false positives and false negatives (Sammut & Webb, 2010):

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Mean Reciprocal Rank (MRR) measures the ranking quality by considering the position of the first relevant item (Voorhees, 1999):

$$MRR = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{rank_i}$$

where $rank_i$ is the rank position of the first relevant item for user i.

Substantial improvements in F1 and MRR were observed (20%–100% increases), particularly for models integrating attention mechanisms and sequence-aware architectures.

Thus, the adoption of diverse modeling techniques, particularly graph-based, deep learning, and hybrid models, led to substantial improvements across multiple critical evaluation dimensions, reinforcing the progress in contemporary recommender systems.

2.4.3. Methodology Types and Problem-Solving Effectiveness

This section systematically evaluates the effectiveness of major methodological families in addressing key challenges within recommender systems. These families include Graph-based models, Deep Learning-based models, Hybrid models, and Content/Context-aware systems. Instead of listing methods in isolation, we provide an empirically grounded comparison that draws on metric-based performance improvements observed across 2018–2024 studies. This enables a deeper understanding of which models succeed under which conditions, and for which evaluation objectives. Each methodology group consists of multiple techniques:

- Graph-based models include Graph Convolutional Networks (GCNs), Trust-aware Graph Embeddings, and Dynamic Graph Learning.
- Deep Learning-based models encompass Neural Collaborative Filtering (NCF), Autoencoder architectures, Recurrent Neural Networks (RNNs), and Transformer-based designs.
- Hybrid models combine Collaborative Filtering (CF) with Content-Based Filtering (CBF), cross-domain transfer, and variational inference such as Variational Autoencoders (VAEs).
- Content and Context-aware models enhance representation learning using textual data, metadata, temporal features, and contextual embeddings.

As illustrated in Figure 1, graph-based models yielded the highest average improvements (approximately 95%), particularly on ranking metrics under sparsity and cold-start conditions. This substantial gain can be attributed to their capacity to model high-order connectivity patterns and capture complex user-item relations within sparse interaction networks.

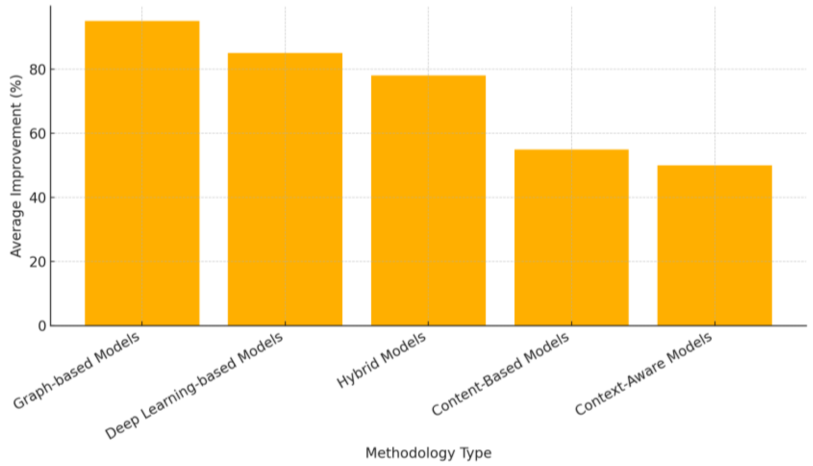

Figure 1. Average Performance Improvement by Methodology Type

Deep Learning-based methods followed closely (~85% improvement), thanks to their proficiency in modeling nonlinear user behavior and sequence dependencies. Hybrid models achieved ~78% average improvements by fusing strengths of multiple strategies, often yielding top performance in error metrics and engagement-focused tasks. In contrast, Content-Based and Context-Aware systems, while trailing in total gain percentages (≈55%–50%), performed particularly well in cases of overspecialization, diversity, and poor feature spaces.

Furthermore, Figure 2 demonstrates that different methodological families contribute distinctively across metric categories. Specifically, graph-based models tend to achieve the largest improvements in ranking metrics such as Recall and NDCG, whereas hybrid and deep learning-based models show broader enhancements across both ranking and engagement metrics, including F1-score and MRR. This pattern highlights that methodological effectiveness is often metric-specific rather than universally consistent.

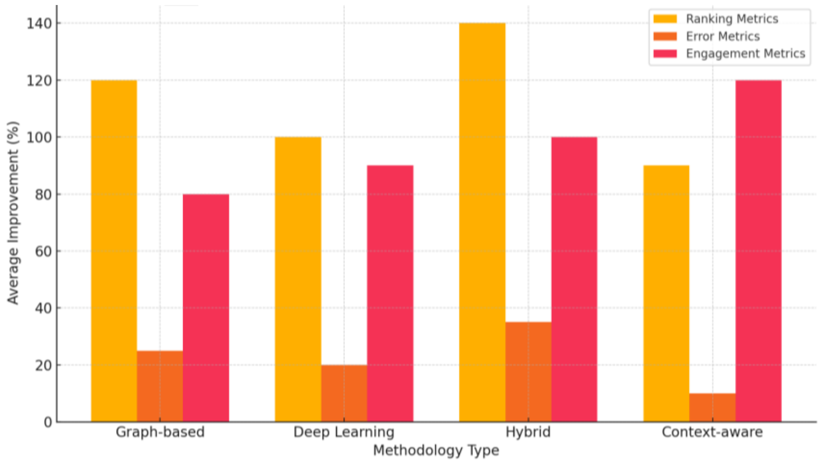

Figure 2. Average Metric Improvements by Methodology Type

Graph-based and Deep Learning models dominate in ranking-based metrics (Recall, NDCG), with gains up to 120–140%. Hybrid approaches outperform others in error metrics (MAE, RMSE), benefiting from regularization and side information. Context-aware models lead in engagement metrics (F1, MRR), with mean improvements exceeding 100% due to their ability to capture user intent and session context.

To further highlight the variation and robustness of metric improvements, the distribution of performance gains across the three primary metric categories is illustrated in Figure 3. This visualization captures how ranking-based metrics (e.g., NDCG, Recall) exhibit the widest range and highest median improvements, while error-based metrics (e.g., MAE, RMSE) show relatively moderate but consistent gains. Engagement metrics (e.g., MRR, F1-score), on the other hand, fall between these two extremes, reflecting balanced enhancements in user interaction quality.

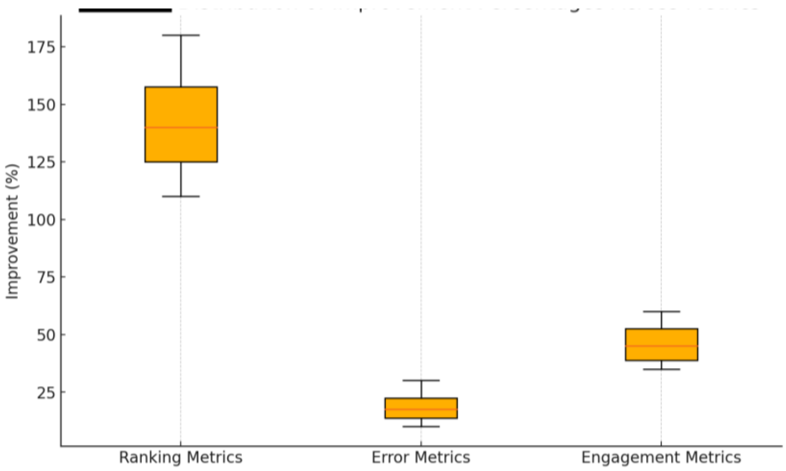

Figure 3. Distribution of Improvement Percentages Across Metrics

The highest variance and magnitude are observed in ranking metrics, where improvements often exceed 150%, indicating their sensitivity to model expressiveness. In contrast, error metrics remain relatively stable (typically 10%–35%), while engagement metrics demonstrate moderate variance and high interpretability in user-centric applications.

Based on the results presented above and cross-referenced with extensive literature evidence (see Appendix A1-A4), the following specific mappings between methodological types and problem domains are represented in the Table 7.

Table 7. Mappings Between Methodological Types and Problem Domains

Methodology Type	**Strengths**
Graph-based Models	Data sparsity, cold-start, multi-hop connectivity
Deep Learning-based Models	Non-linear preference modeling, sequential behavior, complex interactions
Hybrid Models	Robustness to noise, sparsity, dynamic preferences
Context-aware Models	Overspecialization, explainability, diversity, feature incompleteness

In conclusion, these findings clearly suggest that no single methodology is universally superior. Instead, problem-specific alignment of model capabilities with system goals (e.g., personalization, scalability, explainability) is the key to high-performing recommender systems. Building on these empirical insights, the following chapter delves into the theoretical underpinnings, algorithmic designs, and implementation strategies that explain how different recommendation models achieve their respective performance gains across diverse evaluation dimensions.

3. Current Innovations and Applications

As user expectations have evolved and data structures have become increasingly multifaceted, recommender systems have had to transcend conventional architectures. Recent innovations address fundamental limitations, including data sparsity, cold start, and rigid modeling, by introducing flexible, adaptive, and intelligent frameworks that integrate deep learning, contextual awareness, graph theory, and user-centric privacy mechanisms. Although these methods have demonstrated significant advancements, challenges such as computational efficiency and practical implementation remain areas of concern.

3.1. Graph-based Recommendation Systems

Graph-based recommender systems have become a prominent approach in addressing data sparsity and improving the accuracy of recommendation tasks (Wang, He, Wang, Feng, & Chua, 2019; Zou, et al., 2020). Unlike traditional matrix-based models, these systems represent users and items as nodes in a graph, with edges denoting interactions such as clicks, ratings, co-purchases, or shared attributes. A graph in this context is a data structure used to represent relationships between entities, enabling the modeling of user-item interactions beyond direct pairings. This graph representation enables models to capture both direct interactions and higher-order relational patterns through neighborhood structures (Wu, Wei, Yin, Liu, & Zhang, 2020; Ji, Xiang, & Li, 2020). In particular, Graph Neural Networks (GNNs) have gained prominence for their ability to propagate information through nodes (Ji, Xiang, & Li, 2020). GNNs are deep learning architectures designed to perform iterative message passing between nodes in a graph, thereby updating node embeddings based on their neighbors' information. The fundamental operation in GNN-based recommenders involves aggregating features from a node's neighbors to iteratively update its representation. At each layer l, a node v's embedding is computed as follows:

$$h_v^{(l)} = \sigma\left(W^{(l)} \cdot AGGREGATE^{(l)}\left(\left\{h_u^{(l-1)}: u \in \mathcal{N}(v)\right\}\right)\right)$$

Here; $\mathcal{N}(v)$ is the set of neighbors of node v, $h_v^{(l)}$ is the embedding of node v at layer l, $W^{(l)}$ is a trainable weight matrix for layer l, σ is an activation function, and $AGGREGATE$ is a neighborhood aggregation function. Once user and item embeddings are computed, the recommendation score for user u and item i is often calculated via a dot product:

$$\widehat{r_{u,i}} = e_u^T e_i$$

Here, e_u and e_i are the learned embeddings of the user and item after several GNN layers. This process allows the recommender system to propagate information across multi-hop neighborhoods, which is especially beneficial in sparse datasets where direct user-item interactions are limited (Li, et al., 2020). This capacity to capture high-order connectivity distinguishes GNN-based approaches from traditional matrix factorization or shallow neural models.

Moreover, their inductive nature enables generalization to unseen users or items, making them well-suited for cold-start scenarios. Several models such as Neural Graph Collaborative Filtering (Wang et al., 2019) and HN-GCCF (Gong et al., 2023) have demonstrated state-of-the-art performance by efficiently modeling the structural and semantic relationships within recommendation graphs. Recent literature has highlighted the efficacy of graph-based models in collaborative filtering, content-based approaches, and hybrid systems, due to their capacity to represent complex, high-dimensional interactions. NGCF model by Wang et al. (2019) enhances collaborative feedback by leveraging high-order user-item connectivity. Gong et al. (2023) introduced HN-GCCF, which incorporates indirect neighbors into message propagation, preserving richer collaborative signals in sparse environments. Li et al. (2020) proposed Dynamic Graph CF, incorporating user interest evolution over time using dynamic graph structures. Zheng et al. (2022) presented a multi-view GNN model integrating both homogeneous signals such as ratings and heterogeneous signals such as reviews and metadata. Dai et al. (2024) proposed a joint contrastive learning strategy integrating semantic and structural representations, improving robustness against noise and data imbalance. These examples illustrate how graph-based recommender systems are increasingly designed to integrate diverse feedback types such as explicit, implicit, social, textual and handle contextual or sequential dynamics.

Despite their strengths, GNN-based recommender systems face several critical challenges that limit their practical deployment. One of the primary issues is over-smoothing, where node representations become increasingly similar as the number of layers grows, reducing the model's ability to distinguish between different users or items (Li, Han, & Wu, 2018). Additionally, scalability poses a significant concern; operating on large-scale graphs requires considerable memory and computational resources, especially when deep message-passing or personalized inference is involved (Wu et al., 2020). Furthermore, the training complexity increases in environments with noisy, incomplete, or dynamically evolving graphs, such as real-time e-commerce platforms (Ji, Xiang, & Li, 2020). These factors can negatively impact convergence stability and overall recommendation accuracy. Nonetheless, the strong theoretical foundation of GNNs in graph representation learning, combined with their empirical success across a wide range of recommendation tasks, has solidified their position as a core component of modern recommender system research.

Building upon graph-based representations, the field has also seen an extensive adoption of deep learning models, which offer enhanced capacity to capture non-linear user-item interactions and semantic relationships within the data. These approaches span a wide range of techniques from latent factor modeling to sequential prediction, many of which now form the backbone of modern hybrid recommendation pipelines.

3.2. Latent Factor Modeling with Deep Architectures

Latent factor modeling has been a foundational technique in recommender systems, aiming to uncover hidden dimensions that explain user-item interactions. These models embed users and items into dense, continuous vector spaces where their interaction can be modeled. With the advent of deep learning, these methods have evolved into more expressive and flexible architectures, capable of uncovering complex, non-linear relationships within sparse and noisy data (Jain, Nagar, Singh, & Dhar, 2020). Deep latent factor models enhance traditional matrix factorization by embedding users and items into dense, high-dimensional spaces and optimizing them through neural networks. A prominent example of this evolution is Neural Collaborative Filtering (NCF), which replaces the dot product used in matrix factorization with a multi-layer perceptron (MLP) to capture non-linear user-item interactions (He et al., 2017). In NCF, the predicted rating $\widehat{r_{u,i}}$ for a user u and item i is computed as:

$$\widehat{r_{u,i}} = f(e_u, e_i)$$

Here, e_u and e_i denote user and item embeddings and f is a neural network parameterized to learn their interaction function. Extensions like Deep Cooperative Neural Networks further improve expressiveness by employing outer products to capture pairwise feature interactions (Wu, Wei, Yin, Liu, & Zhang, 2020). Similarly, Dual Relations Network (Ji, Xiang, & Li, 2020) simultaneously models both user-item and item-item relations using two neural pathways, improving accuracy by learning rich co-interaction signals. These methods enhance latent factor modeling by capturing higher-order dependencies beyond first-level user-item affinities.

To further reduce noise and overfitting in sparse environments, autoencoder-based architectures have gained traction. Autoencoders are neural networks designed to learn compressed representations of data by reconstructing inputs from lower-dimensional latent spaces. Variational Autoencoders (VAEs) extend this idea with a probabilistic framework that imposes a prior distribution on the latent space, enhancing regularization. The objective function in VAEs is typically given by:

$$\mathcal{L}VAE = Eq(z|x)[\log p\,(x|z)] - KL[q(z|x)||p(z)]$$

where x is the observed interaction vector, z the latent variable, and KL denotes Kullback–Leibler divergence between the posterior and prior distributions. For instance, Zhang et al. (2021) proposed a Hybrid Variational Autoencoder that incorporates both user and item metadata into the VAE framework to improve performance under cold start and sparse conditions. Likewise, Tanuma and Matsui (2022) combined Poisson Factorization with VAEs to model implicit feedback while maintaining recommendation robustness.

In addition to encoding user behavior, latent factor models have been extended to integrate heterogeneous data. For example, Ma, Pan, and Ming (2022) proposed a Structured Collaborative Filtering framework that utilizes implicit feedback from various sources to form a richer representation space. Similarly, Dai et al. (2024) introduced a Joint Contrastive Graph Learning framework that jointly captures both semantic and structural relations between users and items using a contrastive loss objective. These hybrid deep latent models address the limitations of shallow collaborative filtering by embracing auxiliary signals such as text, metadata, and user context.

Despite these advancements, deep latent factor models face several challenges. Their high model capacity makes them prone to overfitting, especially in scenarios with limited supervision (Zhang, Wong, & Chu, 2021). Additionally, training such models requires significant computational resources and can be complex to integrate with contextual or temporal information (Tanuma & Matsui, 2022; Ma, Pan, & Ming, 2022). The black-box nature of the learned embeddings also raises concerns about interpretability, which is particularly important in explainable recommendation settings (Ji, Xiang, & Li, 2020). Nevertheless, the expressive power, improved performance, and ability to integrate multiple data modalities make deep latent models a core component of next-generation recommender systems.

3.3. Sequence-aware Recommendation Models

Sequence-aware recommendation models aim to capture the temporal dynamics and evolving nature of user preferences by analyzing the order and context of user interactions. Unlike static recommendation approaches that treat interactions as unordered sets, sequence-aware models consider the sequential patterns in user behavior such as the order of product views, clicks, or purchases to model both short-term intents and long-term interests. These models have proven especially effective in domains like e-commerce, news, and multimedia where user behavior is temporally rich and context-sensitive (Bauer & Jannach, 2024; Gaiger, Barkan, Tsipory-Samuel, & Koenigstein, 2023).

A major advancement in this domain is the Recurrent Neural Networks (RNNs). RNNs are a class of neural networks designed to handle sequential data by maintaining a hidden state that captures information from previous time steps. In recommender systems, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) architectures, both specialized forms of RNNs, are widely used to model dependencies across time-stamped user-item interactions. These models enable the prediction of the next likely item based on a user's interaction history. The predicted interaction score for time $t + 1$ is often computed as:

$$\widehat{r_{u,t+1}} = f(e_{i_t}, h_t)$$

Here, e_{i_t} is the embedding of the item interacted with at time t, and h_t is the hidden state summarizing the user's history up to time t. The function f is typically a nonlinear transformation using LSTM or GRU units. However, RNNs have limitations in modeling long-range dependencies and parallel processing. To overcome these challenges, Transformer-based architectures, originally introduced for natural language processing, have been successfully adapted to recommender systems. Transformers use attention mechanisms to compute relevance scores between all items in a sequence, allowing the model to selectively focus on past interactions based on their contextual importance. A representative example is SSE-PT (Wu, Li, Hsieh, & Sharpnack, 2020) which employs personalized attention mechanisms to capture user-specific transition patterns, enabling the model to highlight relevant past actions for real-time predictions. In a related approach, HybridBERT4Rec (Channarong, Paosirikul, Maneeroj, & Takasu, 2022) combines content-based and collaborative filtering using a BERT-style encoder to model user-item sequences with high contextual fidelity. The attention mechanism used in these models is defined as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V$$

Here Q, K, V are the query, key, and value matrices derived from the user's interaction history. This mechanism enables the model to weigh past items differently based on their relevance to the current context.

To better handle session-level variability and capture both global and local preferences, hybrid approaches such as Attentive Hybrid Recurrent Neural Networks (Zhang et al., 2021) integrate multiple sequence modeling techniques. These include Bi-LSTM and GRU layers augmented with attention modules, allowing for dynamic adjustment based on user behavior. Additionally, session-aware models like the one proposed by Bauer and Jannach (2023) incorporate short-term user interactions and contextual side information using feature-based architectures, thereby improving real-time personalization.

Despite their advantages, sequence-aware recommendation models face several notable challenges that can limit their effectiveness in real-world applications. One major issue is data sparsity, as long and informative interaction sequences are often unavailable for infrequent users, thereby constraining the ability to personalize recommendations (Hu, Zhang, & Yang, 2019). Moreover, scalability remains a pressing concern; training deep architectures such as RNNs and Transformers over long sequences demands substantial computational resources, which may hinder their deployment in large-scale systems (Wu et al., 2020). The cold start problem also persists, as new users or items with insufficient sequential data are difficult to model accurately, often requiring the incorporation of auxiliary information like user demographics or item metadata to improve prediction accuracy (Sejwal & Abulaish, 2020; Mahesh, Kumar, & Lim, 2023). Finally, sequence data often includes noisy or outdated interactions, which can obscure the user's current intent. Attention-based mechanisms address this issue by dynamically identifying and emphasizing the most relevant segments of user behavior, thus improving the contextual fidelity of recommendations (Gaiger, Barkan, Tsipory-Samuel, & Koenigstein, 2023).

Nevertheless, the ability of sequence-aware recommendation models to capture the temporal evolution of user interests, handle contextual interactions, and personalize results in real time has positioned them as a cornerstone of modern recommender systems. Their integration with attention mechanisms, hybrid architectures, and contextual embeddings continues to yield substantial improvements in both accuracy and user satisfaction.

3.4. Context-aware Recommendation Models

Context-aware recommendation models aim to enhance personalization by incorporating situational factors, such as time, location, device type, or user mood, into the recommendation process. Unlike traditional systems that treat user preferences as static and context-independent, these models recognize that user behavior often varies depending on external conditions. (Liang, Junhao, & Zhou, 2022). The core advancement of context-aware systems lies in transitioning from a traditional user-item matrix, which maps preferences in two dimensions, to a user-item-context tensor. A tensor is a multi-dimensional array structure that enables the system to represent and model the influence of contextual variables on user preferences. In this framework, typical formulation for context-aware recommendation involves predicting the score $\widehat{r_{u,\iota,c}}$ for a user u, item i, and context c as:

$$\widehat{r_{u,\iota,c}} = f(e_u, e_i, e_c)$$

Here e_u, e_i and e_c represent the learned embeddings for the user, item, and context respectively, and f is a non-linear function such as an MLP or attention-based module. This modeling enables the system to account for temporal shifts in preferences. Building on this framework, recent research has explored a range of techniques for effectively integrating contextual data. For instance, Liu, Zhang, and Yan (2020) proposed a latent factor model that learns context-aware embeddings by incorporating auxiliary variables such as time and device type. Their method avoids rigid context constraints while maintaining personalization, making it adaptable for dynamic environments. Similarly, Mahesh, Kumar, and Lim (2023) address the cold-start problem by modeling context as confidence-weighted signals, improving recommendation accuracy with minimal historical data. Gaiger et al. (2023) further enhance context modeling by introducing dynamic user representations through context-target attention mechanisms, improving the model's adaptability to shifting user intents.

Hybrid and deep learning-based methods have also proven valuable in context-aware recommendation. Dong et al. (2018) introduced a recurrent collaborative filtering framework that captures both global preferences and session-specific behaviors over time, while Hu, Zhang, and Yang (2019) leveraged memory networks to incorporate contextual cues dynamically during recommendation generation. More recently, Huang et al. (2023) implemented feature-level attention to selectively weigh contextual attributes like time-of-day and platform, yielding more responsive and explainable recommendation outcomes. These approaches highlight the diversity of modeling strategies, ranging from embedding-level fusion to temporal-aware hybridization, that enable systems to adapt to real-world variability in user behavior.

Despite their advantages, context-aware models face significant implementation challenges. Capturing and modeling diverse contextual variables require rich metadata and careful feature engineering. Additionally, real-time context acquisition raises privacy and latency concerns. Balancing model complexity and computational efficiency remains crucial, especially in mobile or latency-sensitive applications (Sejwal & Abulaish, 2020; Bauer & Jannach, 2024).

Nevertheless, context-aware recommendation systems have become essential for delivering personalized, situation-sensitive experiences. Their integration with latent factor models, attention-based mechanisms, and hybrid frameworks continues to advance the field, offering richer and more adaptive user modeling in modern recommender systems.

3.5. Multi-source and Cross-domain Integration

Modern recommender systems increasingly rely on the integration of multiple data sources and cross-domain knowledge to overcome the limitations of sparse interaction matrices, narrow-domain overfitting, and poor generalizability. Unlike traditional systems that primarily exploit user-item interactions, these approaches enhance performance by incorporating diverse auxiliary signals such as reviews, metadata, social networks, domain ontologies, and interaction histories across different platforms.

One line of research focuses on the fusion of heterogeneous implicit feedback. For instance, Ma, Pan, and Ming (2022) proposed the Structured Collaborative Filtering (SCF) model, which utilizes structured signals from item features and neighboring user behaviors to better represent user preferences in sparse environments. This approach demonstrates the benefit of modeling feedback heterogeneity for more accurate recommendations. Cross-domain transfer techniques are also widely adopted to address the cold-start problem and expand coverage. Sejwal and Abulaish (2022) presented the RecTE framework, integrating topic embeddings with collaborative filtering to improve rating prediction in low-data settings. Their model highlights how semantic-level transfer enhances recommendation quality when domain-specific data is limited. Likewise, Ibrahim et al. (2023) developed a hybrid neural collaborative filtering model that leverages deep interaction modeling and hierarchical attention to learn cross-domain user behaviors, showing improved accuracy in multi-platform scenarios. Another research direction emphasizes explainability and robustness during multi-source integration. Gatti, Diaz-Pace, and Schiaffino (2023) proposed a hybrid recommender for artworks by combining deep autoencoders with a social influence graph, allowing the system to consider both visual preferences and social interactions. Their method underlines the importance of combining content and social signals in explainable recommendation contexts. Additionally, Dai et al. (2024) introduced a joint contrastive learning approach that simultaneously encodes semantic and structural relations between users and items, tackling the issue of data imbalance and noise a common problem in systems aggregating multiple sources.

Despite their advantages, multi-source and cross-domain systems face several technical and conceptual challenges. These include the need for feature alignment across domains, user identity matching, transfer learning stability, and privacy-preserving mechanisms. Moreover, designing architectures that can scale efficiently across datasets with varying distributions and modalities remains a central concern in practical deployments.

Nonetheless, the integration of diverse data sources and knowledge transfer across domains has proven to significantly improve the adaptability, robustness, and personalization of modern recommender systems. As the boundaries between platforms continue to blur, these techniques are poised to play a pivotal role in delivering unified and context-aware recommendations.

3.6. Privacy-preserving and Explainable Recommendation Systems

With the increasing reliance on personalized services, recommender systems are now expected to balance predictive accuracy with ethical responsibilities such as user privacy and transparency. The massive collection and processing of user data raise concerns around surveillance, data misuse, and opaque decision-making (Mazeh & Shmueli, 2020). To address privacy issues, several architectures have been introduced. Personal Data Store (PDS) frameworks decentralize data ownership, enabling users to control access to their information (Mazeh & Shmueli, 2020). Federated Learning (FL) also helps mitigate risks by allowing model training on decentralized devices, avoiding raw data aggregation. Moreover, adversarial training and differential privacy techniques aim to prevent re-identification and inference attacks. Zhu & Chen (2022) proposed a Variational Bandwidth Autoencoder that dynamically regulates latent representations, enhancing robustness and limiting leakage. In cross-domain settings, Hu et al. (2024) integrated attention-based transfer mechanisms to avoid exposing sensitive co-occurrence patterns that may reveal user traits. In parallel, explainable recommendation systems strive to make model decisions transparent. Feature-based hybrid methods like those by Bauer & Jannach (2023) combine side information with gradient boosting to provide traceable recommendations. Attention visualization and rule-based post-hoc techniques also contribute to increasing user trust by offering understandable justifications.

Despite these innovations, notable challenges persist. Privacy-preserving models often reduce personalization performance due to limited data access (Mazeh & Shmueli, 2020; Zhu & Chen, 2022). Explainability-focused models, while more transparent, can require extensive computational resources and rely heavily on rich side data, which may not always be available (Fan et al., 2019; Wu et al., 2020). Further, approaches such as Gaiger et al. (2023), which use attention to enhance interpretability, still depend on consistent behavioral signals challenging in cold-start scenarios. Broader issues such as data sparsity, popularity bias, and overspecialization also continue to limit the effectiveness and scalability of both privacy-aware and interpretable systems (Sejwal & Abulaish, 2022; Gatti, Díaz-Pace, & Schiaffino, 2023).

4. Discussion

4.1. Literature Gaps and Research Opportunities

Despite the extensive research conducted on recommender systems, there are still several gaps in the existing literature. The identification of these gaps is crucial for the development of more robust, efficient, and user-oriented recommendation models. The following are the main areas where further research is needed.

4.1.1. Gaps in Data Diversity and Integration of Multiple Data Sources

Recommendation systems primarily rely on user-product interaction data, which limits their understanding of user preferences by neglecting other influential factors. Recent research suggests augmenting these systems with additional data sources like social media interactions, demographic profiles, and contextual variables such as time and location (Chen, et al., 2018; Wayne, et al., 2016; Puspitaningrum, 2019). However, challenges remain in effectively integrating these diverse data streams. Future research should focus on developing advanced models that combine these data types to enhance recommendation accuracy and relevance, particularly in cold-start scenarios where user interaction data is scarce. Effectively addressing integration gaps could enable the development of more robust, adaptive, and user-centric recommendation systems that better align with individual preferences.

4.1.2. Scalability and Computational Efficiency Issues

Deep learning techniques and graph-based models have improved recommendation accuracy, but they face significant challenges, primarily high computational costs when processing large-scale datasets (Fakhfakh, Ben Ammar, & Ben Amar, 2017). Moreover, this challenge is further exacerbated in real-world scenarios due to the volume and complexity of data. Scalability becomes a concern as these models may struggle to handle growing data sizes without increased computational resources. Consequently, there is a need for more efficient architectural designs and optimization strategies that reduce computational demands while maintaining high recommendation accuracy, facilitating the development of robust and scalable recommendation systems for complex user interactions and large datasets (Singh, Mishra, & Singhal, 2023).

4.1.3. Challenges in Balancing Diversity and Innovation

The current state of recommendation systems prioritizes accuracy, often at the expense of diversity and innovation. This focus results in users receiving repetitive suggestions of familiar products, limiting their exposure to new experiences and diminishing engagement and satisfaction (Sharma & Mann, 2013). To improve this situation, future research could focus on developing methodologies that integrate accuracy with diversity and innovation, enhancing user experience and fostering exploratory interactions with content. A balanced framework in recommendation systems can better cater to users' nuanced preferences, leading to a more enriching discovery journey.

4.1.4. Contradictions between Privacy Concerns and Disclosability

The rise of recommender systems across digital platforms has raised significant privacy concerns due to their dependence on personal data for personalized suggestions. This has created a conflict between delivering tailored experiences and protecting user privacy (Zhang & Sundar, 2019; Goncalves, Gomes, & Aguiar, 2015). Current research focuses on balancing trustworthy recommendations with strong protection of sensitive information. Innovative solutions like federated learning, differential privacy, and explainable artificial intelligence (XAI) are being explored. Federated learning enables algorithms to learn from decentralized data sources, enhancing privacy without sacrificing recommendation accuracy (Li, Chen, Zhao, & Hu, 2021; Wang, et al., 2022). Differential privacy adds noise to data to keep individual information confidential, while XAI aims to clarify how recommendations are made, building user trust (Müllner, Lex, Schedl, & Kowald, 2023). However, further research is needed to assess the effectiveness of these methods in real-world applications and their impact on user experience, as the goal is to create systems that respect privacy while improving user satisfaction and engagement.

Ongoing studies at the intersection of privacy, security, and user-centric design are crucial for the future of recommender systems.

4.1.5. Inadequate Modeling of Dynamic User Preferences

User preferences are dynamic and change over time due to various factors, yet many recommender systems struggle to adapt, resulting in stale recommendations that can reduce user engagement (Kunde, et al., 2020; Chalyi & Leshchynskyi, 2020). To improve this, advanced systems should capture both long-term preferences and short-term interest fluctuations. Sequential modeling techniques can analyze user interactions over time to identify trends, while attention mechanisms can highlight relevant recent interactions (Zhao, et al., 2020; Ying, et al., 2018). By integrating these methodologies, recommender systems can become more adaptive, offering personalized recommendations that align with users' current needs and established tastes, ultimately enhancing user experience and loyalty.

4.2. Future Research Directions

These cutting-edge methodologies tackle a variety of challenges aimed at enhancing the precision, adaptability, and overall user experience associated with collaborative filtering, content-based filtering, and hybrid recommender systems. Each of these techniques offers distinct advantages and is suited for specific applications, enabling the optimization of recommender systems to cater to diverse use cases, from e-commerce to content streaming platforms.

4.2.1. Scalability and Computational Efficiency

Advancing the scalability of recommendation systems while minimizing computational costs is crucial for handling growing volumes of user data and complex interactions. These cutting-edge methodologies tackle a variety of challenges aimed at enhancing the precision, adaptability, and overall user experience associated with collaborative filtering, content-based filtering, and hybrid recommender systems. Each of these techniques offers distinct advantages and is suited for specific applications, enabling the optimization of recommender systems to cater to diverse use cases, from e-commerce to content streaming platforms. Looking ahead, further research should prioritize the development of efficient architectural designs, model compression techniques, and distributed computing strategies. Moreover, the innovation of streamlined, mobile-optimized neural network variants could facilitate the effective operation of recommendation systems on portable devices, supporting real-time, large-scale recommendation tasks while conserving energy and resources.

4.2.2. Dynamic User Behavior Modeling

Capturing and adapting to fluctuations in user behavior over time remains a significant challenge for current recommender systems. User preferences are dynamic and change due to various factors, yet many systems struggle to adapt, resulting in stale or irrelevant recommendations. Sequential modeling techniques and attention-based mechanisms offer promising solutions by dynamically adjusting recommendations based on evolving user preferences. Future research should focus on making sequential and attention-based recommendation models more dynamic, thereby facilitating real-time predictions that better align with users' current contextual needs and preferences. The integration of large-scale, real-time adaptive systems with distributed data storage frameworks could empower recommendation engines to continuously learn and update their predictions.

4.2.3. Privacy-Preserving Techniques

Ensuring user privacy without compromising recommendation quality is expected to remain a critical research direction. The rise of recommender systems across digital platforms has intensified concerns regarding the collection, storage, and processing of personal data. Methodologies such as federated learning, differential privacy, and homomorphic encryption are gaining traction for their potential to protect sensitive user information while maintaining model accuracy. Future research should emphasize the refinement of privacy-preserving machine learning techniques and the ethical design of systems that prioritize user consent, transparency, and compliance with emerging privacy regulations. Building recommender systems that foster user trust while ensuring robust personalization remains an essential challenge.

4.2.4. Multi-Source Data Integration and Cross-Domain Recommendations

Incorporating additional data streams beyond traditional user-item interactions such as social media activity, user-generated reviews, and demographic attributes offers a promising pathway toward more comprehensive personalization. However, the effective fusion of heterogeneous data sources still poses considerable challenges. Future developments are expected to focus on enhancing the efficiency of multi-source data integration techniques and advancing cross-domain recommendation models. Such advancements would enable recommendation systems to capture the multifaceted nature of user preferences and deliver richer, more contextually aware suggestions.

4.2.5. Explainability and Transparency

Enhancing transparency and explainability in recommendation systems is vital for fostering user trust and engagement. While high prediction accuracy is important, understanding why a specific item is recommended is equally crucial for user satisfaction. Employing explainable AI (XAI) methodologies will not only improve users' comprehension of recommendation outcomes but also assist developers in diagnosing, refining, and validating the system's decision-making processes. Future research should aim to create recommendation models that balance performance with interpretability, contributing to a more trustworthy and user-centric recommendation landscape.

4.2.6. Towards a Transformative Future

Progress in these domains heralds a transformative era for recommendation systems, characterized by greater personalization, transparency, scalability, and ethical sensitivity. As technological advancements continue to unfold, recommender systems are poised to become increasingly ubiquitous across diverse aspects of everyday life, from online retail and streaming services to education and healthcare. Thus, recommender systems stand at the threshold of a new generation, where personalization, privacy, explainability, and ethical responsibility converge to redefine user experiences across digital platforms.

5. Conclusion

This study has provided a comprehensive and critical review of the recent methodological advancements in recommender systems between 2018 and 2024. By systematically analyzing a wide range of approaches ranging from graph-based models and deep learning architectures to hybrid methods and content-aware systems the study has highlighted not only the substantial progress achieved in addressing classical challenges such as data sparsity, cold-start, and scalability but also the increasing attention devoted to emerging concerns such as user privacy, diversity, and dynamic preference modeling.

The empirical evidence gathered across diverse datasets and evaluation metrics demonstrates that no single methodology universally outperforms others. Instead, the effectiveness of methodological approaches appears to be highly context dependent. Graph-based models consistently exhibited superior performance under sparsity conditions; deep learning techniques excelled at capturing complex and nonlinear user-item interactions; hybrid approaches successfully combined multiple signals to address noise and dynamic behaviors; and content- and context-aware methods significantly enhanced recommendation quality by leveraging auxiliary information such as metadata and user reviews.

In light of these findings, future research must prioritize the development of recommender systems that integrate multiple data modalities, enhance scalability without compromising accuracy, foster diversity alongside precision, and uphold privacy and explainability as core design principles. As recommender systems become increasingly integrated into everyday digital life, the imperative to balance technological sophistication with user-centric values grows ever more critical.

Ultimately, sustained interdisciplinary innovation spanning computer science, behavioral sciences, ethics, and human-computer interaction will be essential for advancing the field towards the creation of recommendation systems that are not only more accurate and adaptive but also more transparent, ethical, and empowering for users in diverse real-world contexts.

References

Abdollahpouri, H. (2019). Popularity Bias in Ranking and Recommendation. 2019 AAAI/ACM Conference on AI, Ethics, and Society (pp. 529–530). Honolulu, HI, USA: ACM.

Amer-Yahia, S., Lakshmanan, L., Vassilvitski, S., & Yu, C. (2009). attling Predictability and Overconcentration in Recommender Systems. IEEE Data Engineering Bulletin, 32, 33-40.

Bae, H. K., Kim, H. O., Shin, W. Y., & Kim, S. W. (2021). "How to get consensus with neighbors?": Rating standardization for accurate collaborative filtering. Knowledge-Based Systems, 234.

Basilico, J., & Hofmann, T. (2004). Unifying collaborative and content-based filtering. 21st International Conference on Machine Learning. Banff, Alberta, Canada: ACM.

Bauer, J., & Jannach, D. (2024). Hybrid session-aware recommendation with feature-based models. User Modeling and User-Adapted Interaction, 34, 691–728.

Bennett, J., & Lanning, S. (2007). The Netflix Prize. KDD.

Bertin-Mahieux, T., Ellis, D. P., Whitman, B., & Lamere, P. (2011). The Million Song Dataset. ISMIR.

Burke, R., Felfernig, A., & Göker, M. H. (2011). Recommender Systems: An Overview. Ai Magazine, 32(3), 13-18.

Cano, E., & Morisio, M. (2017). Hybrid recommender systems: A systematic literature review. Intelligent Data Analysis, 21(6), 1487-1524.

Celma, O. (2010). Music Recommendation and Discovery: The Long Tail, Long Fail, and Long Play. Springer.

Chalyi, S., & Leshchynskyi, V. O. (2020). Temporal Modeling of User Preferences in Recommender System. 9th International Conference on Information Control Systems & Technologies, (pp. 518-528).

Channarong, C., Paosirikul, C., Maneeroj, S., & Takasu, A. (2022). HybridBERT4Rec: A Hybrid (Content-Based Filtering and Collaborative Filtering) Recommender System Based on BERT. IEEE Access, 10, 56193-56206.

Chen, H., Xin, X., Wang, D., & Ding, Y. (2021). Decomposed Collaborative Filtering: Modeling Explicit and Implicit Factors For Recommender Systems. 14th ACM International Conference on Web Search and Data Mining (pp. 958–966). Israel: ACM.

Chen, J. Y., Hsu, P. Y., Cheng, M. S., Lei, H. T., Huang, S. H., Ko., Y. H., & Huang, C. W. (2018). Investigating Deciding Factors of Product Recommendation in Social Media. Advances in Swarm Intelligence: Lecture Notes in Computer Science. 10942, pp. 249–257. Springer.

Chen, Y., & Huang, J. (2024). Effective Content Recommendation in New Media: Leveraging Algorithmic Approaches. IEEE Access, 12, 90561-90570.

Choi, J., Jeon, J., & Park, N. (2021). LT-OCF: Learnable-Time ODE-based Collaborative Filtering. 30th ACM International Conference on Information & Knowledge Management (pp. 251–260). Queensland, Australia: ACM.

da Costa, A. F., Manzato, M. G., & Campello, R. J. (2019). Boosting collaborative filtering with an ensemble of co-trained recommenders. Expert Systems with Applications, 115, 427-441.

Dacrema, M. F., Gasparin, A., & Cremonesi, P. (2018). Deriving item features relevance from collaborative domain knowledge. Knowledge-aware and Conversational Recommender Systems (KaRS) Workshop. Vancouver, Canada: CEUR-WS.

Dai, J., Li, Q., Nong, T., Bi, Q., & Chu, H. (2024). Joint contrastive learning of structural and semantic for graph collaborative filtering. Neurocomputing, 586.

Dong, B., Zhu, Y., Li, L., & Wu, X. (2020). Hybrid Collaborative Recommendation via Dual-Autoencoder. IEEE Access, 8, 46030-46040.

Dong, D., Zheng, X., Zhang, R., & Wang, Y. (2018). Recurrent Collaborative Filtering for Unifying General and Sequential Recommender. 27th International Joint Conference on Artificial Intelligence, (pp. 3350-3356).

Duong, T. N., Vuong, T. A., Nguyen, D. M., & Dang, Q. H. (2020). Utilizing an Autoencoder-Generated Item Representation in Hybrid Recommendation System. IEEE Access, 8, 75094-75104.

Fakhfakh, R., Ben Ammar, A., & Ben Amar, C. (2017). Deep Learning-Based Recommendation: Current Issues and Challenges. International Journal of Advanced Computer Science and Applications, 8(12).

Fan, W., Ma, Y., Yin, D., Wang, J., Tang, J., & Li, Q. (2019). Deep social collaborative filtering. 13th ACM Conference on Recommender Systems (pp. 305-313). Copenhagen, Denmark: ACM.

Fengou, M. A., Athanasiou, G., Mantas, G., Griva, I., & Lymberopoulos, D. (2013). Towards Personalized Services in the Healthcare Domain. In B. Furht, & A. Agarwal, Handbook of Medical and Healthcare Technologies (pp. 417–433). New York: Springer.

Frolov, E., & Oseledets, I. (2019). HybridSVD: when collaborative information is not enough. 13th ACM Conference on Recommender Systems (pp. 331–339). Copenhagen, Denmark: ACM.

Gaiger, K., Barkan, O., Tsipory-Samuel, S., & Koenigstein, N. (2023). Not All Memories Created Equal: Dynamic User Representations for Collaborative Filtering. IEEE Access, 11, 34746-34763.

Gatti, I., Diaz-Pace, J. A., & Schiaffino, S. (2023). A hybrid approach for artwork recommendation. Engineering Applications of Artificial Intelligence.

Goncalves, J. M., Gomes, D., & Aguiar, R. L. (2015). Privacy in Data Publishing for Tailored Recommendation Scenarios. Transactions on Data Privacy, 8(3), 245-271.

Gong, K., Song, X., Li, W., & Wang, S. (2024). HN-GCCF: High-order neighbor-enhanced graph convolutional collaborative filtering. Knowledge-Based Systems, 283.

Gulla, J. A., Zhang, L., Liu, P., Özgöbek, Ö., & Su, X. (2017). The Adressa Dataset for News Recommendation. ICIIM.

Gupta, M. K., Rao, C. R., Bhavsingh, M., & Srilakshmi, M. (2023). A Comparative Analysis of Collaborative Filtering Similarity Measurements for Recommendation Systems. International Journal on Recent and Innovation Trends in Computing and Communication, 11(3), 184–192.

Hansen, C., Hansen, C., Simonsen, J. G., Alstrup, S., & Lioma, C. (2020). Content-aware Neural Hashing for Cold-start Recommendation. 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 971–980). China: ACM.

Harper, F. M., & Konstan, J. A. (2015). The MovieLens Datasets: History and Context. ACM TiiS.

He, R., & McAuley, J. (2016). VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback. AAAI.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). "Evaluating collaborative filtering recommender systems."

Hu, G. (2019). Personalized Neural Embeddings for Collaborative Filtering with Text. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 1, pp. 2082–2088. Minneapolis, Minnesota, USA: Association for Computational Linguistics.

Hu, G., Zhang, Y., & Yang, Q. (2019). Transfer Meets Hybrid: A Synthetic Approach for Cross-Domain Collaborative Filtering with Text. WWW '19: The World Wide Web Conference (pp. 2822–2829). San Francisco, CA, USA: ACM.

Hu, Z., Nakagawa, S., Cai, S. M., Ren, F., & Deng, J. (2024). Enhancing cross-market recommendations by addressing negative transfer and leveraging item co-occurrences. Information Systems, 124.

Huang, Z., Jin, B., Zhao, H., Liu, Q., Lian, D., Tengfei, B., & Chen, E. (2023). Personal or General? A Hybrid Strategy with Multi-factors for News Recommendation. ACM Transactions on Information Systems, 41(2), 1-29.

Huang, Z., Yu, C., Ni, J., Liu, H., Zeng, C., & Tang, Y. (2019). An Efficient Hybrid Recommendation Model With Deep Neural Networks. IEEE Access, 7, 137900-137912.

Huang, Z., Zeng, D. D., & Chen, H. (2007). A Comparison of Collaborative-Filtering Recommendation Algorithms for E-commerce. IEEE Intelligent Systems, 22, 68-78.

Ibrahim, M., Bajwa, I. S., Sarwar, N., Hajjej, F., & Sakr, H. A. (2023). An Intelligent Hybrid Neural Collaborative Filtering Approach for True Recommendations. IEEE Access, 11, 64831-64849.

Jain, A., Nagar, S., Singh, P. K., & Dhar, J. (2020). EMUCF: Enhanced multistage user-based collaborative filtering through non-linear similarity for recommendation systems. Expert Systems with Applications, 161.

Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2010). Hybrid recommendation approaches. D. Jannach, M. Zanker, A. Felfernig, & G. Friedrich içinde, Recommender Systems: An Introduction (s. 124-142). Cambridge University Press.

Järvelin, K., & Kekäläinen, J. (2002). "Cumulated gain-based evaluation of IR techniques."

Ji, D., Xiang, Z., & Li, Y. (2020). Dual Relations Network for Collaborative Filtering. IEEE Access, 8, 109747-109757.

Joorabloo, N., Jalili, M., & Ren, Y. (2020). Improved Collaborative Filtering Recommendation Through Similarity Prediction. IEEE Access, 8, 202122-202132.

Kim, K. J., & Ahn, H. (2011). Collaborative Filtering with a User-Item Matrix Reduction Technique. International Journal of Electronic Commerce, 16(2), 107–128.

Kumar, B., Sharma, N., Sharma, B., Herencsar, N., & Srivastava, G. (2023). Hybrid Recommendation Network Model with a Synthesis of Social Matrix Factorization and Link Probability Functions. Sensors, 23(5).

Kunde, S., Mishra, M., Pandit, A., Singhal, R., Nambiar, M. K., Shroff, G., & Gupta, S. (2020). Recommending in changing times. 14th ACM Conference on Recommender Systems (pp. 714–719). Brazil: ACM.

Li, A., Liu, X., & Yang, B. (n.d.). Item Attribute-aware Graph Collaborative Filtering Author links open overlay panel. Expert Systems with Applications, 238.

Li, W., Chen, H., Zhao, R., & Hu, C. (2021). A Federated Recommendation System Based on Local Differential Privacy Clustering. IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Internet of People and Smart City Innovation (pp. 364-369). IEEE.

Li, Q., Han, Z., & Wu, X. (2018). Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning. AAAI Conference on Artificial Intelligence.

Li, X., Zhang, M., Wu, S., Liu, Z., Wang, L., & Yu, P. S. (2020). Dynamic Graph Collaborative Filtering. IEEE International Conference on Data Mining (ICDM) (pp. 322-331). Sorrento, Italy: IEEE.

Liang, G., Junhao, W., & Zhou, W. (2022). Individual Diversity Preference Aware Neural Collaborative Filtering. Knowledge-Based Systems, 258.

Lika, B., Kolomvatsos, K., & Hadjiefthymiades, S. (2014). Facing the cold start problem in recommender systems. Expert Systems With Applications, 41(4), 2065-2073.

Liu, X., Jiyong, Z., & Chenggang, Y. (2020). Towards context-aware collaborative filtering by learning context-aware latent representations. Knowledge-Based Systems, 199.

Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender System Application Developments: A Survey. Decision Support Systems, 74, 12-32.

Müllner, P., Lex, E., Schedl, M., & Kowald, D. (2023). Differential privacy in collaborative filtering recommender systems: a review. Frontiers in Big Data, 6.

Ma, W., Pan, W., & Ming, Z. (2022). SCF: Structured collaborative filtering with heterogeneous implicit feedback. Knowledge-Based Systems, 258.

McAuley, J., & Leskovec, J. (2013). Hidden factors and hidden topics: Understanding rating dimensions with review text. RecSys.

McAuley, J., Targett, C., Shi, Q., & van den Hengel, A. (2015). Image-based recommendations on styles and substitutes. SIGIR.

Magron, P., & Févotte, C. (2022). Neural content-aware collaborative filtering for cold-start music recommendation. Data Mining and Knowledge Discovery, 36, 1971–2005.

Mahesh, T. R., Kumar, V. V., & Lim, S. J. (2023). UsCoTc: Improved Collaborative Filtering (CFL) recommendation methodology using user confidence, time context with impact factors for performance enhancement. PLOS ONE, 18(3).

Mazeh, I., & Shmueli, E. (2020). A personal data store approach for recommender systems. Expert Systems With Applications, 139.

Nath, R. K., & Ahmad, T. (2022). Content Based Recommender System: Methodologies, Performance, Evaluation and Application. 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N) (s. 423-428). Greater Noida, India: IEEE.

Nikolakopoulos, A. N., & Karypis, G. (2020). Boosting Item-based Collaborative Filtering via Nearly Uncoupled Random Walks. ACM Transactions on Knowledge Discovery from Data, 14(6), 1-26.

Ortal, P., & Edahiro, M. (2020). Switching Hybrid Method Based on User Similarity and Global Statistics for Collaborative Filtering. IEEE Access, 8, 213401-213415.

Pazzani, M. J., & Billsus, D. (2007). Content-Based Recommendation Systems. In P. Brusilovsky, A. Kobsa, & W. Nejdl, The Adaptive Web: Lecture Notes in Computer Science (pp. 325-341). Springer.

Perez-Alcazar, J. J., Calderon-Benavides, M. L., & Gonzalez-Caro, C. N. (2003). Towards an information filtering system in the Web integrating collaborative and content based techniques,". First Latin American Web Congress (pp. 222-225). Santiago, Chile: IEEE.

Pirasteh, P., Jung, J. J., & Hwang, D. (2014). Item-Based Collaborative Filtering with Attribute Correlation: A Case Study on Movie Recommendation. N. T. Nguyen, B. Attachoo, B. Trawinski, & K. Somboonviwat içinde, Intelligent Information and Database Systems: Lecture Notes in Computer Science. Springer.

Polatidis, N., & Georgiadis, C. K. (2013). Recommender systems: The Importance of personalization in E-business environments. International Journal of E-Entrepreneurship and Innovation (IJEEI), 4(4), 32-46.

Puspitaningrum, D. (2019). Social Influences in Recommendation Systems. 7th International Conference on Cyber and IT Service Management (CITSM) (pp. 1-6). Jakarta, Indonesia: IEEE.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). GroupLens: An open architecture for collaborative filtering of netnews. Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, 175-186. https://doi.org/10.1145/192844.192905

Resnick, P., & Varian, H. R. (1997). Recommender systems. Communications of the ACM, 40(3), 56–58.

Ludewig, M., Mauro, N., Latifi, S., & Jannach, D. (2018). Evaluation of Session-based Recommendation Algorithms. RecSys.

Saleh, B., & Elgammal, A. (2015). Large-scale classification of fine-art paintings: Learning the right metric on the right feature. International Journal for Computer Vision.

Sammut, C., & Webb, G. I. (2010). "Encyclopedia of Machine Learning."

Sarne, G. M. (2015). A novel hybrid approach improving effectiveness of recommender systems. Journal of Intelligent Information Systems, 44, 397–414.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. 10th International Conference on World Wide Web (pp. 285–295). Hong Kong: ACM.

Sejwal, V. K., & Abulaish, M. (2020). A hybrid recommendation technique using topic embedding for rating prediction and to handle cold-start problem. Expert Systems with Applications, 209.

Seng, D., Chen, G., & Zhang, Q. (2020). Item-based collaborative memory networks for recommendation. IEEE Access, 8, 213027-213037.

Shams, B., & Haratizadeh, S. (2018). Item-based collaborative ranking. Knowledge-Based Systems, 152.

Sharma, M., & Mann, S. (2013). A survey of recommender systems: approaches and limitations. International journal of innovations in engineering and technology, 2(2), 8-14.

Sharma, R., Gopalani, D., & Meena, Y. (2017). Collaborative filtering-based recommender system: Approaches and research challenges. 3rd International Conference on Computational Intelligence & Communication Technology (CICT) (s. 1-6). Ghaziabad, India: IEEE.

Sharma, S., Rana, V., & Malhotra, M. (2022). Automatic recommendation system based on hybrid filtering algorithm. Education and Information Technologies, 27, 1523–1538.

Singh, R. K., Mishra, M., & Singhal, R. (2023). Scalable High-Performance Architecture for Evolving Recommender System. 3rd Workshop on Machine Learning and Systems (pp. 154–162). Rome, Italy: ACM.

Sivaramakrishnan, N., Subramaniyaswamy, V., Viloria, A., Vijayakumar, V., & Senthilselvan, N. (2021). A deep learning-based hybrid model for recommendation generation and ranking. Neural Computing and Applications, 33, 10719-10736.

Song, D., Li, Z., Jiang, M., Qin, L., & Liao, L. (2019). A novel temporal and topic-aware recommender model. World Wide Web, 22, 2105–2127.

Tanuma, I., & Matsui, T. (2022). Variational Autoencoder-Based Hybrid Recommendation with Poisson Factorization for Modeling Implicit Feedback. IEEE Access, 10, 60696-60706.

Van Meteren, R., & Van Someren, M. (tarih yok). Using content-based filtering for recommendation. The machine learning in the new information age: MLnet/ECML2000 workshop, (s. 47-56).

Verma, V., & Aggarwal, R. (2020). Neighborhood-based Collaborative Recommendations: An Introduction. In P. Johri, K. V. Jitendra, & P. Sudip, Applications of Machine Learning (pp. 91-110). Singapore: Springer.

Voorhees, E. M. (1999). “The TREC-8 question answering track report.”

Walker, J., Zhou, F., Baagyere, E. Y., Ahene, E., & Zhang, F. (2022). Implicit optimal variational collaborative filtering. Complex & Intelligent Systems, 8, 4369–4384.

Wan, X., Rubens, N., Okamoto, T., & Feng, Y. (2015). Content Filtering Based on Keyword Map. 2015 International Conference on Electrical, Computer Engineering and Electronics (pp. 851-856). Atlantis Press.

Wang, C., Blei, D. M., & Fei-Fei, L. (2011). Collaborative topic modeling for recommending scientific articles. KDD.

Wang, C. D., Chen, Y. H., Xi, W. D., Huang, L., & Xie, G. (2022). Cross-domain explicit–implicit-mixed collaborative filtering neural network. IEEE Transactions on Systems, Man, and Cybernetics, 52, 6983-6997.

Wang, Q., Yin, H., Wang, H., Nguyen, Q. V., Huang, Z., & Lizhen, C. (2019). Enhancing Collaborative Filtering with Generative Augmentation. 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 548-556). Anchorage, AK, USA: ACM.

Wang, X., He, X., Wang, M., Feng, F., & Chua, T. S. (2019). Neural Graph Collaborative Filtering. 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (s. 165–174). Paris, France: ACM.

Wang, X., Meng, S., Chen, Y., Liu, Q., Yuan, R., & Li, Q. (2022). Federated Deep Recommendation System Based on Multi-View Feature Embedding. 9th International Conference on Data Science and Advanced Analytics (DSAA) (pp. 1-9). Shenzhen, China: IEEE.

Wang, X., Wang, R., Shi, C., Song, G., & Li, Q. (2020). Multi-Component Graph Convolutional Collaborative Filtering. Proceedings of the AAAI Conference on Artificial Intelligence, 34(4), 6267-6274.

Wang, Y., Han, L., Qian, Q., Xia, J., & Li, J. (2022). Personalized Recommendation via Multi-dimensional Meta-paths Temporal Graph Probabilistic Spreading. Information Processing & Management, 59(1).

Wang, Z., Tan, Y., & Zhang, M. (2010). Graph-Based Recommendation on Social Networks. 12th International Asia-Pacific Web Conference (s. 116-122). Busan, South Korea: IEEE.

Wayne, X. Z., Li, S., He, Y., Wang, L., Wen, J. R., & Li, X. (2016). Exploring demographic information in social media for product recommendation. Knowledge and Information Systems, 49, 61–89.

Wu, L., Li, S., Hsieh, C. J., & Sharpnack, J. (2020). SSE-PT: Sequential Recommendation Via Personalized Transformer. 14th ACM Conference on Recommender Systems (pp. 328–337). Brazil: ACM.

Wu, L., Xia, Y., Min, S., & Xia, Z. (2024). Deep Attentive Interest Collaborative Filtering for Recommender Systems. IEEE Transactions on Emerging Topics in Computing, 12(2), 467-481.

Wu, Y., Wei, J., Yin, J., Liu, X., & Zhang, J. (2020). Deep Collaborative Filtering Based on Outer Product. IEEE Access, 8, 85567-85574.

Xin, X., He, X., Zhang, Y., Zhang, Y., & Jose, J. (2019). Relational Collaborative Filtering: Modeling Multiple Item Relations for Recommendation. 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 125–134). Paris, France: ACM.

Xue, F., He, X., Wang, X., Xu, J., Kai, L., & Hong, R. (2019). Deep Item-based Collaborative Filtering for Top-N Recommendation. ACM Transactions on Information Systems, 37(3), 1-25.

Yanes, N., Ayman, M. M., Ezz, M. M., & Almuayqil, S. N. (2020). A Machine Learning-Based Recommender System for Improving Students Learning Experiences. IEEE Access, 8, 201218-201235.

Ying, H., Zhuang, F., Zhang, F., Liu, Y., Xu, G., Xie, X., . . . Wu, J. (2018). Sequential Recommender System based on Hierarchical Attention Networks. 27th International Joint Conference on Artificial Intelligence (pp. 3926-3932). Stockholm, Sweden: AAAI Press.

Yun, S. Y., & Youn, S. D. (2010). Recommender system based on user information. 7th International Conference on Service Systems and Service Management (s. 1-3). Tokyo, Japan: IEEE.

Zeng, Z., Lin, J., Li, L., Pan, W., & Ming, Z. (2019). Next-Item Recommendation via Collaborative Filtering with Bidirectional Item Similarity. ACM Transactions on Information Systems, 38(1), 1-22.

Zhang, B., & Sundar, S. S. (2019). Proactive vs. reactive personalization: Can customization of privacy enhance user experience? International Journal of Human-Computer Studies, 128, 86-99.

Zhang, H., Wong, R. K., & Chu, V. W. (2021). Hybrid Variational Autoencoder for Recommender Systems. ACM Transactions on Knowledge Discovery from Data, 16(2).

Zhang, L., Wang, P., Li, J., Xiao, Z., & Shi, H. (2021). Attentive hybrid recurrent neural networks for sequential recommendation. Neural Computing and Applications, 33, 11091-11105.

Zhang, Q., Lu, J., & Jin, Y. (2021). Artificial intelligence in recommender systems. Complex & Intelligent Systems, 7, 439–457.

Zhang, R., Qidong, L., Chun, G., Wei, J. X., & Huiyi, M. (2014). Collaborative Filtering for Recommender Systems. Second International Conference on Advanced Cloud and Big Data (CBD), (s. 301-308).

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). "Deep learning-based recommender system: A survey and new perspectives."

Zhao, Q., Zhang, H., Xia, H., & McAuley, J. (2019). Learning user representations from sequential behavior for recommendation. WSDM.

Zhao, L., Zhang, L., Chenyi, L., Chen, X., Gao, J., & Gao, J. (2020). Attention with Long-Term Interval-Based Deep Sequential Learning for Recommendation. Complexity.

Zhao, W. X., Li, S., He, Y., Wang, L., Wen, J. R., & Li, X. (2016). Exploring demographic information in social media for product recommendation. Knowledge and Information Systems, 49, 61–89.

Zheng, J., Chen, S., Du, Y., & Song, P. (2022). A multiview graph collaborative filtering by incorporating homogeneous and heterogeneous signals. Information Processing & Management, 59(6).

Zhou, W., Du, Y., Duan, M., Ul Haq, A., & Shah, F. (2022). NtCF: Neural Trust-Aware Collaborative Filtering Toward Hierarchical Recommendation Services. Arabian Journal for Science and Engineering, 47, 1239–1252.

Zhu, T., Liu, G., & Chen, G. (2020). Social Collaborative Mutual Learning for Item Recommendation. ACM Transactions on Knowledge Discovery from Data, 14(4), 1-19.

Zhu, Y., & Chen, Z. (2021). Variational Bandwidth Auto-Encoder for Hybrid Recommender Systems. IEEE Transactions on Knowledge and Data Engineering, 35(5), 5371-5385.

Ziegler, C.-N., McNee, S. M., Konstan, J. A., & Lausen, G. (2005). Improving recommendation lists through topic diversification. WWW.

Zou, L., Xia, L., Gu, Y., Zhao, X., Liu, W., Huang, J. X., & Yin, D. (2020). Neural Interactive Collaborative Filtering. 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 749–758). China: ACM.

Appendix A1. Literature Review on UBCF Advancements with All Metrics

Study	Problem	Method	Solution	Dataset	Metrics	Improvement
(Wang, He, Wang, Feng, & Chua, 2019)	Sparsity	Graph-based Collaborative Filtering	Modeling high-order connectivity with graph embeddings	Gowalla	Recall @20	↑10.61%
					NDCG @20	↑5.13%
				Yelp2018	Recall @20	↑12.89%
					NDCG @20	↑8.06%
				Amazon Books	Recall @20	↑11.33%
					NDCG @20	↑3.96%
(Fan, et al., 2019)	Cold Start Problem	Combining Multiple Data and Feedback	Analyzing social connections with deep learning	Ciao (Traning: %60)	RMSE	↓15.11%
					MAE	↓21.22%
				Ciao (Traning: %80)	RMSE	↓12.16%
					MAE	↓11.76%
				Epinions (Traning: %60)	RMSE	↓19.75%
					MAE	↓24.66%
				Epinions (Traning: %80)	RMSE	↓18.66%
					MAE	↓26.95%
(Zou, et al., 2020)	Scalability	Adaptive Techniques	Optimizing with self-attention networks and Q-learning	MovieLens 1M	Precision @40	↑7.91%
					Recall @40	↑7.44%
				Netflix	Precision @40	↑6.43%
					Recall @40	↑1.62%
(Joorabloo, Jalili, & Ren, 2020)	Scalability Cold Start Problem	Enhanced Similarity Calculations	Re-ranking user-item similarities based on trends	Movielens-100K	Precision @20	↑39.68%
					Recall @20	↑29.33%
					F1 @20	↑26.61%
				Goodreads	Precision @20	↑11.1%
					Recall @20	↑11.43%
					F1 @20	↑7.65%
(Wu, Wei, Yin, Liu, & Zhang, 2020)	Sparsity	Deep Learning Approaches	Learning user-item correlations with deep learning	Yelp	HR @20	↑156.2%
					NDCG @20	↑197.24%
(Ji, Xiang, & Li, 2020)	Sparsity	Deep Learning Approaches	Learning interactions with dual neural networks	Movielens-100K	HR @10	↑64.88%
					NDCG @10	↑70.64%
				Movielens-1M	HR @10	↑53.81%
					NDCG @10	↑77.07%
				Amazon Music	HR @10	↑65.33%
					NDCG @10	↑65.09%
				Amazon Movie	HR @10	↑57.64%
					NDCG @10	↑48.89%
(Li, et al., 2020)	Sparsity	Dynamic and Contextual User Characteristics	Representing dynamics with neural network layers	LastFM	MRR @10	↑34.3%
					Recall @10	↑27.7%
				Wikipedia	MRR @10	↑5.4%
					Recall @10	↑3.6%
				Reddit	MRR @10	↑0.2%
					Recall @10	↑0.5%
	Sparsity			Movielens-100K	RMSE	↓ 2.71%

(Ortal & Edahiro, 2020)		Hybrid and Cross-Domain Models	Combining custom similarity with regional statistics		MAE	↓ 3.09%
				MovieLens 1M	RMSE	↓ 5.09%
					MAE	↓ 6.17%
				Epinions	RMSE	↓ 4.19%
					MAE	↓ 2.31%
				Book-crossing	RMSE	↓ 5.88%
					MAE	↓ 2.97%
				Jester	RMSE	↓ 1.89%
					MAE	↓ 1.51%
				Synthetic	RMSE	↓ 6.70%
					MAE	↓ 1.68%
(Liu, Jiyong, & Chenggang, 2020)	Cold Start Problem	Dynamic and Contextual User Characteristics	Deriving latent representations with contextual factors	Movielens-100K	MAE	↓ 3.10%
					RMSE	↓ 2.09%
					MAP	↓0.30%
					MRR	↑1.01%
				Douban Book	MAE	↓ 4.48%
					RMSE	↓ 4.68%
					MAP	↑5.39%
					MRR	↑4.89%
				Douban Music	MAE	↓ 4.40%
					RMSE	↓ 4.67%
					MAP	↑2.45%
					MRR	↑3.16%
(Jain, Nagar, Singh, & Dhar, 2020)	Sparsity Cold Start Problem	Hybrid and Cross-Domain Models	Active learning with non-linear similarity model	Movielens-100K	Precision	↑13.59%
					MAE	↓4.88%
				Movielens-1M	Precision	↑43.1%
					MAE	↓10.59%
(Bae, Kim, Shin, & Kim, 2021)	Sparsity, Scalability	Indirect Neighbor-Based Strategies	Standardizing ratings using neighbor scores	Movielens-100K	F1 @10	↑21.89%
					MRR @10	↑23.81%
				MovieLens-1M	F1 @10	↑10.45%
					MRR @10	↑17.60%
				Watcha	F1 @10	↑10.23%
					MRR @10	↑10.24%
(Chen, Xin, Wang, & Ding, 2021)	Sparsity Cold Start Problem	Enhanced Similarity Calculations	Capturing nuances with factor-level attention	MovieLens 20M	Recall @20	↑6.79%
					NDCG @20	↑6.56%
				Book-crossing	Recall @20	↑19.57%
					NDCG @20	↑20.80%
				LastFM	Recall @20	↑4.11%
					NDCG @20	↑4.62%
(Zhou, Du, Duan, Ul Haq, & Shah, 2022)	Sparsity	Graph-based Collaborative Filtering	Enhancing with trust-aware deep neural networks	Yelp	RMSE	↓ 24.96%
					MAE	↓ 35.40%
					Recall @10	↑158.46%
					Precision @10	↑ 36.42%

				CiaoDVDs	RMSE	↓30.71%
					MAE	↓ 26.37%
					Recall @10	↑ 99.58%
					Precision @10	↑ 13.84%
				Epinions	RMSE	↓29.42%
					MAE	↓31.63%
					Recall @10	↑ 114.87%
					Precision @10	↑ 23.18%
(Zheng, Chen, Du, & Song, 2022)	Scalability	Combining Multiple Data and Feedback	Graph CF with homogeneous and heterogeneous feedback	MovieLens 1M	AUC	↑ 1.55%
					LogLoss	↓8.38%
					NDCG @10	↑ 1.41%
				Book-crossing	AUC	↑ 3.66%
					LogLoss	↓5.98%
					NDCG @10	↑ 2.55%
(Ma, Pan, & Ming, 2022)	Sparsity	Combining Multiple Data and Feedback	Structured CF with heterogeneous feedback	Alibaba2015	Precision @5	↑29.09%
					Recall @5	↑37.04%
					F1 @5	↑31.76%
					NDCG @5	↑37.18%
				Rec15	Precision @5	↑2.65%
					Recall @5	↑2.74%
					F1 @5	↑2.7%
					NDCG @5	↑2.68%
				UB	Precision @5	↑8.33%
					Recall @5	↑9.55%
					F1 @5	↑10.17%
					NDCG @5	↑9.32%
(Wang, Chen, Xi, Huang, & Xie, 2022)	Sparsity Cold Start Problem	Hybrid and Cross-Domain Models	Acquiring cross-domain knowledge with deep learning	MovieLens 1M Domain 1	HR @10	↑6.9%
					NDCG @10	↑12.6%
				MovieLens 1M Domain 2	HR @10	↑1.7%
					NDCG @10	↑7.9%
				MovieLens 10M Domain 1	HR @10	↑3.5%
					NDCG @10	↑2.5%
				MovieLens 10M Domain 2	HR @10	↑3.3%
					NDCG @10	↑5.9%
				Amazon Music	HR @10	↑0.6%
					NDCG @10	↑4.1%
				Amazon Video	HR @10	↑1.7%
					NDCG @10	↑1.5%
				Amazon Books	HR @10	↑5.1%
					NDCG @10	↑17.8%
				Amazon Movie	HR @10	↑1.8%
					NDCG @10	↑9.5%
				Amazon Phone	HR @10	↑28.1%
					NDCG @10	↑29.3%

				Amazon Electronics	HR @10	↑20.5%
					NDCG @10	↑26.5%
(Mahesh, Kumar, & Lim, 2023)	Sparsity	Enhanced Similarity Calculations	Incorporating user confidence and temporal factors	Movielens-100K	MAE	↑1.33%
					RMSE	↑0.93%
					F1	↑0.13%
(Gaiger, Barkan, Tsipory-Samuel, & Koenigstein, 2023)	Sparsity Scalability	Dynamic and Contextual User Characteristics	Adapting representations with context-target attention	MovieLens 1M	HR @10	↑165.4%
					MRR @10	↑164.7%
					NDCG @10	↑100%
				Netflix	HR @10	↑43%
					MRR @10	↑42%
					NDCG @10	↑25%
				Yahoo! Music	HR @10	↑170.5%
					MRR @10	↑153.3
					NDCG @10	↑253.5%
				Amazon Books	HR @10	↑68.18%
					MRR @10	↑137.78%
					NDCG @10	↑106.15%
				Moviesdat	HR @10	↑148.78%
					MRR @10	↑150%
					NDCG @10	↑168.42%
(Wu, Xia, Min, & Xia, 2024)	Scalability	Deep Learning Approaches	Analyzing past interactions with attention mechanisms	Pinterest	HR @10	↑1.63%
					NDCG @10	↑3.47%
				Digital Music	HR @10	↑1.36%
					NDCG @10	↑4.34%
				Office Products	HR @10	↑1.75%
					NDCG @10	↑3.28%
				MovieLens 1M	HR @10	↑3.18%
					NDCG @10	↑4.63%
(Gong, Song, Li, & Wang, 2024)	Sparsity	Indirect Neighbor-Based Strategies	Enhancing propagation with high-order GCN	Gowalla	Recall @20	↑3.9%
					NDCG @20	↑3%
				Yelp2018	Recall @20	↑12%
					NDCG @20	↑13.2%
				Amazon Books	Recall @20	↑57.7%
					NDCG @20	↑63.5%

Appendix A2. Literature Review on IBCF Advancements with All Metrics

Study	Problem	Method	Solution	Dataset	Metric	Improvement
(Dacrema, Gasparin, & Cremonesi, 2018)	Cold Start Problem, Popularity Bias	Graph-based Collaborative Filtering	Integrating collaborative similarity into content	Netflix	Precision	↑40.77%
					Recall	↑52.20%
					MRR	↑32.80%
					MAP	↑47.34%
					NDCG	↑44.81%
				The Movies	Precision	↑8.90%
					Recall	↑8.60%
					MRR	↑3.99%
					MAP	↑11.62%
					NDCG	↑8.66%
(Shams & Haratizadeh, 2018)	Popularity Bias, Diversity of Recommendations	Deep Learning Approaches	Personalized item ranking with neural networks	MovieLens-100K	NDCG @5	↑9.68%
(da Costa, Manzato, & Campello, 2019)	Sparsity, Popularity Bias	Deep Learning Approaches	Ensemble of multiple recommender systems	Yahoo Movies	RMSE	↓ 0. 13%
					F1	↑ 4. 27%
				FilmTrust	RMSE	↓ 4. 99%
					F1	↑ 9. 87%
				CiaoDVD	RMSE	↓ 1. 90%
					F1	↑ 2. 47%
				MovieLens-2K	RMSE	↓ 1. 29%
					F1	↑ 2.30%
				Jester	RMSE	↓ 0. 31%
					F1	↑ 1. 57%
				Book Crossing	RMSE	↓ 2. 18%
					F1	↑ 2. 15%
				Amazon Digital Music	RMSE	↓ 1. 21%
					F1	↑ 2. 06%
				Yahoo Music	RMSE	↓ 1. 87%
					F1	↑ 3. 70%
(Xue, et al., 2019)	Sparsity, Cold Start Problem	Deep Learning Approaches	Capturing high-level relations from past interactions	MovieLens-1M	HR @10	↑23.89%
					NDCG @10	↑48.55%
				Pinterest	HR @10	↑2.59%
					NDCG @10	↑5.76%
(Wang, et al., 2019)	Sparsity, Cold Start Problem	Graph Convolutional Networks	Improving quality with GAN-based data augmentation	Amazon Electronics	HR @20	↑50.0%
				Amazon Health	HR @20	↑48.3%
				Amazon Beauty	HR @20	↑32.3%
				Amazon Games	HR @20	↑31.6%
(Zeng, Lin, Li, Pan, & Ming, 2019)	Cold Start Problem, Popularity Bias	Deep Learning Approaches	Optimizing with time-sensitive	MovieLens-10M	Precision@5	↑47.8%
					Recall@5	↑48.1%

			bidirectional similarity		F1@5	↑48.2%
					NDCG@5	↑60.0%
				Netflix	Precision@5	↑42.4%
					Recall@5	↑41.9%
					F1@5	↑41.7%
					NDCG@5	↑47.0%
(Xin, He, Zhang, Zhang, & Jose, 2019)	Sparsity, Diversity of Recommendations	Graph Convolutional Networks	Predicting preferences with two-stage attention	MovieLens-100K	HR @10	↑25.0%
					MRR @10	↑39.1%
					NDCG @10	↑27.9%
				KKBox	HR @10	↑18.7%
					MRR @10	↑40.7%
					NDCG @10	↑33.3%
(Nikolakopoulos & Karypis, 2020)	Sparsity, Popularity Bias	Random Walks and Similarity Prediction	Enhancing diversity with random walk methods	MovieLens-1M	HR @10	↑18.97%
					NDCG @10	↑32.38%
				Yahoo	HR @10	↑41.65%
					NDCG @10	↑54.20%
				Pinterest	HR @10	↑13.9%
					NDCG @10	↑16.6%
				Amazon Movies&TV	HR @10	↑24.7&
					NDCG @10	↑28.4%
				Amazon Books	HR @10	↑23.2%
					NDCG @10	↑29.6%
				Amazon Electronics	HR @10	↑48.4%
					NDCG @10	↑57.0%
(Wang, Wang, Shi, Song, & Li, 2020)	Sparsity, Popularity Bias	Graph Convolutional Networks	Optimizing with multi-component GCN in user-item graph	Yelp	RMSE	↓4.1%
					MAE	↓34.5%
				Amazon	RMSE	↓5.0%
					MAE	↓9.6%
				MovieLens-100K	RMSE	↓5.9%
					MAE	↓6.7%
(Zhu, Liu, & Chen, 2020)	Popularity Bias, Sparsity	Deep Learning Approaches	Combining item and social filtering models	CiaoDVD	HR @10	↑54.4%
					NDCG @10	↑63.4%
				Epinions	HR @10	↑45.7%
					NDCG @10	↑63.3%
				Yelp	HR @10	↑87.9%
					NDCG @10	↑126.0%
(Choi, Jeon, & Park, 2021)	Cold Start Problem, Sparsity	Graph Convolutional Networks	Extending GCN with learnable differential equations	Gowalla	Recall @10	↑45.2%
					NDCG @10	↑41.9%
				Yelp	Recall @10	↑55.0%
					NDCG @10	↑55.1%

				Amazon Books	Recall @10	↑76.8%
					NDCG @10	↑73.9%
(Walker, Zhou, Baagyere, Ahene, & Zhang, 2022)	Sparsity, Popularity Bias	Variational Autoencoders	Encoding hard-to-detect feedback with autoencoders	MovieLens-10M	Recall @20	↑187.7%
					NDCG @20	↑191.1%
					Precision @20	↑162.3%
				MovieLens-1M	Recall @20	↑160.9%
					NDCG @20	↑155.1%
					Precision @20	↑124.7%
				Amazon Electronics	Recall @20	↑81.0%
					NDCG @20	↑91.4%
					Precision @20	↑100.0%
(Liang, Junhao, & Zhou, 2022)	Popularity Bias, Diversity of Recommendations	Deep Learning Approaches	Modeling user diversity preferences with neural nets	MovieLens-1M	HR @10	↓0.42%
					NDCG @10	↑14.34%
					F1 @10	↑2.14%
				MovieLens-10M	HR @10	↑7.10%
					NDCG @10	↑7.19%
					F1 @10	↑8.62%
				MovieLens-20M	HR @10	↑2.16%
					NDCG @10	↑3.14%
					F1 @10	↑2.92%
(Li, Liu, & Yang)	Cold Start Problem, Sparsity	Graph Convolutional Networks	Enhancing accuracy with item attribute-aware GCN	Amazon	Precision @10	↑64.6%
					Recall @10	↑54.3%
					F1 @10	↑61.5%
				Douban Book	Precision @10	↑16.0%
					Recall @10	↑15.1%
					F1 @10	↑15.6%
				Douban Movie	Precision @10	↑7.9%
					Recall @10	↑14.4%
					F1 @10	↑11.2%
				Yelp	Precision @10	↑27.8%
					Recall @10	↑24.2%
					F1 @10	↑27.1%
				KKBox	Precision @10	↑14.8%
					Recall @10	↑19.2%
					F1 @10	↑17.2%
				IMDB	Precision @10	↑7.2%
					Recall @10	↑9.1%
					F1 @10	↑9.7%

Appendix A3. Literature Review on CBF Advancements with All Metrics

Study	Problem	Method	Solution	Dataset	Metric	Improvement
(Dong, Zheng, Zhang, & Wang, 2018)	Poorly Defined Features	Repetitive Cross-Domain Approaches	Combining RNN with matrix factorization for sequential modeling	MovieLens-1M	HR @10	↑7.24%
					NDCG @10	↑22.40%
				Netflix	HR @10	↑11.26%
					NDCG @10	↑29.61
(Hu G., 2019)	Poorly Defined Features	Content-Aware Collaborative Filtering	Integrating user-item interactions with textual data	Amazon	HR @10	↑164.6%
					NDCG @10	↑169.7%
					MRR @10	↑171.6%
				Cheetah	HR @10	↑30.0%
					NDCG @10	↑9.94%
					MRR @10	↑8.73%
(Song, Li, Jiang, Qin, & Liao, 2019)	Cold Start Problem	Hybrid and Temporal Models	Combining CF and CBF with temporal context	CiteULike	NDCG @100	↑2095.0%
				MovieLens-100K	NDCG @100	↑222.1%
(Wu, Li, Hsieh, & Sharpnack, 2020)	Cold Start Problem	Personalized Transformer Models	Ranking items with personalized transformers	Amazon Beauty	Recall @10	↑25.59%
					NDCG @10	↑47.94%
				Amazon Games	Recall @10	↑64.20%
					NDCG @10	↑103.70%
				Steam	Recall @10	↑22.32%
					NDCG @10	↑40.59%
				MovieLens-1M	Recall @10	↑92.61%
					NDCG @10	↑164.19%
(Sivaramakrishnan, Subramaniyaswamy, Viloria, Vijayakumar, & Senthilselvan, 2021)	Cold Start Problem	Content-Aware Collaborative Filtering	Two-stage hybrid deep learning with Bayesian autoencoders	Book-Crossing	Precision @10	↑158.16%
					Recall @10	↑206.14%
				Amazon Books	Precision @10	↑219.35%
					Recall @10	↑203.23%
(Mazeh & Shmueli, 2020)	Poorly Defined Features	Deep Social Collaborative Filtering	Privacy-preserving architecture with personal data stores	MovieLens-10M	Precision @10	↑12.62%
					Recall @10	↑11.16%
					AUC	↑3.64%
(Hansen, Hansen, Simonsen, Alstrup, & Lioma, 2020)	Cold Start Problem	Content-Aware Collaborative Filtering	Generating binary hash codes with autoencoders	Yelp	NDCG @10	↑2.59%
					MRR @10	↑2.70%
				Amazon	NDCG @10	↑1.25%
					MRR @10	↑1.49%
(Sharma, Rana, & Malhotra, 2022)	Cold Start Problem	Hybrid and Temporal Models	Combining CF and CBF for improved recommendations	Books Dataset	MAE	↓5.04%
(Channarong, Paosirikul, Maneeroj, & Takasu, 2022)	Cold Start Problem	Hybrid and Temporal Models	Combining CBF and CF with BERT-based deep learning	MovieLens-1M	HR @10	↑57.69%
					NDCG @10	↑41.96%
				Yelp	HR @10	↑28.11%
					NDCG @10	↑14.62%
				Goodreads	HR @10	↑27.01%

					NDCG @10	↑25.15%
(Wang, Han, Qian, Xia, & Li, 2022)	Overspecialization	Metadata-Based Recommendations	Probabilistic propagation with temporal multi-dimensional graphs	MovieLens-100K	Recall @10	↑1445.82%
					F1 @10	↑1477.07%
				Last.fm	Recall @10	↑5390.72%
					F1 @10	↑5436.45%
(Magron & Févotte, 2022)	Cold Start Problem	Content-Aware Collaborative Filtering	Extracting content info with deep learning for music	Million Song Dataset	NDCG @10	↑38.41%
(Chen & Huang, 2024)	Overspecialization	Hybrid and Temporal Models	Analyzing behavior and item features with algorithmic techniques	Unshared	Precision @10	↑30.95%
					Recall @10	↑26.32%
					RMSE	↓19.13%
					MAE	↓18.48

Appendix A4. Literature Review on Hybrid Advancements with All Metrics

Study	Problem	Method	Solution	Dataset	Metric	Improvement
(Huang, et al., 2019)	Scalability	Attention Mechanisms in Sequential Recommendations	Combining deep learning with traditional ML	Mobile News	HR @10	↑1.75%
(Huang, et al., 2019)	Scalability	Attention Mechanisms in Sequential Recommendations	Combining deep learning with traditional ML	Mobile News	NDCG @10	↑2.32
(Huang, et al., 2019)	Scalability	Attention Mechanisms in Sequential Recommendations	Combining deep learning with traditional ML	Mobile News	MRR @10	↑2.47
(Huang, et al., 2019)	Scalability	Attention Mechanisms in Sequential Recommendations	Combining deep learning with traditional ML	Amazon	HR @10	↑15.63%
(Huang, et al., 2019)	Scalability	Attention Mechanisms in Sequential Recommendations	Combining deep learning with traditional ML	Amazon	NDCG @10	↑9.43%
(Huang, et al., 2019)	Scalability	Attention Mechanisms in Sequential Recommendations	Combining deep learning with traditional ML	Amazon	MRR @10	↑7.34%
(Duong, Vuong, Nguyen, & Dang, 2020)	Sparsity, Noise	Item Representations with Autoencoder	Generating variational representations with 3-layer autoencoder	MovieLens 20M	RMSE	↓3.93%
(Duong, Vuong, Nguyen, & Dang, 2020)	Sparsity, Noise	Item Representations with Autoencoder	Generating variational representations with 3-layer autoencoder	MovieLens 20M	MAE	↓4.36%
(Dong, Zhu, Li, & Wu, 2020)	Cold Start Problem, Noise	Explainability and User Interaction	Combining matrix factorization with dual autoencoders	MovieLens-100K	RMSE	↓4.55%
(Dong, Zhu, Li, & Wu, 2020)	Cold Start Problem, Noise	Explainability and User Interaction	Combining matrix factorization with dual autoencoders	MovieLens-100K	MAE	↓4.93%
(Dong, Zhu, Li, & Wu, 2020)	Cold Start Problem, Noise	Explainability and User Interaction	Combining matrix factorization with dual autoencoders	MovieTweetings-10K	RMSE	↓42.60%
(Dong, Zhu, Li, & Wu, 2020)	Cold Start Problem, Noise	Explainability and User Interaction	Combining matrix factorization with dual autoencoders	MovieTweetings-10K	MAE	↓44.85%
(Dong, Zhu, Li, & Wu, 2020)	Cold Start Problem, Noise	Explainability and User Interaction	Combining matrix factorization with dual autoencoders	FilmTrust	RMSE	↓5.16%
(Dong, Zhu, Li, & Wu, 2020)	Cold Start Problem, Noise	Explainability and User Interaction	Combining matrix factorization with dual autoencoders	FilmTrust	MAE	↓3.14%
(Zhang, Wong, & Chu, 2021)	Sparsity, Cold Start Problem	Variational Autoencoders	Encoding user-item info into latent space	CiteULike	Recall	↑1.54%
(Zhang, Wong, & Chu, 2021)	Sparsity, Cold Start Problem	Variational Autoencoders	Encoding user-item info into latent space	CiteULike	NDCG	↑4.67%
(Zhang, Wang, Li, Xiao , & Shi, 2021)	Dynamic User Preferences	Variational Autoencoders	Integrating Bi-LSTM and GRU with attention	MovieLens 1M	Recall @20	↑2766.21%
(Zhang, Wang, Li, Xiao , & Shi, 2021)	Dynamic User Preferences	Variational Autoencoders	Integrating Bi-LSTM and GRU with attention	MovieLens 1M	NDCG @10	↑292.52%
(Zhang, Wang, Li, Xiao , & Shi, 2021)	Dynamic User Preferences	Variational Autoencoders	Integrating Bi-LSTM and GRU with attention	MovieLens 1M	Precision @10	↑114.93%
(Zhang, Wang, Li, Xiao , & Shi, 2021)	Dynamic User Preferences	Variational Autoencoders	Integrating Bi-LSTM and GRU with attention	Tmall	Recall @20	↑5093.98%
(Zhang, Wang, Li, Xiao , & Shi, 2021)	Dynamic User Preferences	Variational Autoencoders	Integrating Bi-LSTM and GRU with attention	Tmall	NDCG @10	316.13%
(Zhang, Wang, Li, Xiao , & Shi, 2021)	Dynamic User Preferences	Variational Autoencoders	Integrating Bi-LSTM and GRU with attention	Tmall	Precision @10	↑1812.5%
(Tanuma & Matsui, 2022)	Noise, Sparsity	Variational Autoencoders	Combining VAE with Poisson factorization	Million Song	NDCG	↑1.64%
(Tanuma & Matsui, 2022)	Noise, Sparsity	Variational Autoencoders	Combining VAE with Poisson factorization	Million Song	MSE	↓4.77%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	YelpNYC	Precision	↑6.46%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	YelpNYC	Recall	↑8.15%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	YelpNYC	NDCG	↑9.40%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	YelpZip	Precision	↑24.73%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	YelpZip	Recall	↑19.96%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	YelpZip	NDCG	↑21.17%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	TripAdvisor	Precision	↑9.71%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	TripAdvisor	Recall	↑11.57%
(Sejwal & Abulaish, 2020)	Cold Start Problem, Sparsity	Deep Learning and Feature Representation	Integrating topic modeling with CF	TripAdvisor	NDCG	↑14.74
(Zhu & Chen, 2021)	Noise, Sparsity	Attention Mechanisms in Sequential Recommendations	Encoding attributes with variational bandwidth autoencoder	CiteULike-a	Recall @20	↑29.44%
(Zhu & Chen, 2021)	Noise, Sparsity	Attention Mechanisms in Sequential Recommendations	Encoding attributes with variational bandwidth autoencoder	CiteULike-a	NDCG @100	↑24.37%
(Zhu & Chen, 2021)	Noise, Sparsity	Attention Mechanisms in Sequential Recommendations	Encoding attributes with variational bandwidth autoencoder	CiteULike-t	Recall @20	↑47.40%
(Zhu & Chen, 2021)	Noise, Sparsity	Attention Mechanisms in Sequential Recommendations	Encoding attributes with variational bandwidth autoencoder	CiteULike-t	NDCG @100	↑40.74%

				Amazon Toys & Games	Recall @20	↑64.77%
					NDCG @100	↑72.58%
(Huang, et al., 2023)	Diversity of Recommendations	Topic Embedding and Contextual Information	Fusing user preferences with timeliness effects	Toutiao	MRR	↑23.92%
					NDCG @5	↑33.94%
				Adressa	MRR	↑88.53%
					NDCG @5	↑74.26%
(Ibrahim, Bajwa, Sarwar, Hajjej, & Sakr, 2023)	Cold Start Problem, Scalability	Cross-Domain Recommendations	Integrating hierarchical attention with deep modeling	IMDb	RMSE	↓95.67%
				Yelp 2013	RMSE	↓98.11%
				Yelp 2014	RMSE	↓96.32%
(Bauer & Jannach, 2024)	Dynamic User Preferences, Cold Start Problem	Explainability and User Interaction	Combining session-based modeling with gradient boosting	Retailrocket	Recall @1	↑37.0%
					MAP @10	↑24.0%
					MRR	↑24.1%
					NDCG	↑19.4%
				Diginetica	Recall @1	↑67.8%
					MAP @10	↑41.6%
					MRR	↑38.3%
					NDCG	↑28.2%
(Gatti, Diaz-Pace, & Schiaffino, 2023)	Multiple Data Integration, Personalization	Singular Value Parsing Extensions	Enhancing precision with deep autoencoders and social graphs	WikiArt	Precision @10	↑287.5%
(Kumar, Sharma, Sharma, Herencsar, & Srivastava, 2023)	Cold Start Problem, Sparsity	Dual Autoencoder for Collaborative Proposals	Combining social matrix factorization with link probabilities	Last.fm	Recall @50	↑22.73%
(Hu, Nakagawa, Cai, Ren, & Deng, 2024)	Dynamic User Preferences	Attention Mechanisms in Sequential Recommendations	Leveraging past behaviors with attention-based model	XMarket DE	Recall @10	↑12.93%
					NDCG @10	↑26.30%
				XMarket JP	Recall @10	↑31.56%
					NDCG @10	↑58.30%
				XMarket IN	Recall @10	↑12.30%
					NDCG @10	↑71.25%
				XMarket FR	Recall @10	↑12.59%
					NDCG @10	↑27.30%
				XMarket MX	Recall @10	↑13.97%
					NDCG @10	↑35.42%
				XMarket CA	Recall @10	↑17.17
					NDCG @10	↑19.03