applied sciences

MDPI

Review

A Comprehensive Survey of Recommender Systems Based on Deep Learning

Hongde Zhou [1], Fei Xiong [1] and Hongshu Chen [2,*]

[1] Key Laboratory of Communication and Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing 100044, China; 21120181@bjtu.edu.cn (H.Z.); xiongf@bjtu.edu.cn (F.X.)
[2] School of Management and Economics, Beijing Institute of Technology, Beijing 100081, China
* Correspondence: hongshu.chen@bit.edu.cn

Abstract: With the increasing abundance of information resources and the development of deep learning techniques, recommender systems (RSs) based on deep learning have gradually become a research focus. Although RSs have evolved in recent years, a systematic review of existing RS approaches is still warranted. The main focus of this paper is on recommendation models that incorporate deep learning techniques. The objective is to guide novice researchers interested in this field through the investigation and application of the proposed recommendation models. Specifically, we first categorize existing RS approaches into four types: content-based recommendations, sequence recommendations, cross-domain recommendations, and social recommendation methods. We then introduce the definitions and address the challenges associated with these RS methodologies. Subsequently, we propose a comprehensive categorization framework and novel taxonomies for these methodologies, providing a thorough account of their research advancements. Finally, we discuss future developments regarding this topic.

Keywords: recommender systems; deep learning; social networks; sequence recommendation; cross-domain recommendation

check for updates

Citation: Zhou, H.; Xiong, F.; Chen, H. A Comprehensive Survey of Recommender Systems Based on Deep Learning. *Appl. Sci.* **2023**, *13*, 11378. https://doi.org/10.3390/app132011378

Academic Editor: Kiril Tenekedjiev

Received: 19 September 2023
Revised: 5 October 2023
Accepted: 13 October 2023
Published: 17 October 2023

1. Introduction

With the rapid progression of technology and the development of the Internet, we have transitioned from an era of information scarcity to the age of big data. The growth in the amount of information available has led to the challenge of "information overload" [1]. In the context of the digital age, recommender systems (RSs) have arisen. RSs analyze data to discern user preferences for items and assist users in efficiently sifting through information [2], directing them towards the content that is most relevant and valuable for their interests. Currently, RSs are widely adopted and have provided many economic benefits. Essentially, RSs are founded on the premise that when users exhibit similar item ratings or behaviors, they are likely to demonstrate similar ratings or actions on other items [3].

Lately, due to the continuous improvement in computational capabilities, artificial neural networks (ANNs) have started to garner widespread attention [4]. This has propelled deep learning to emerge as a burgeoning field in computer science. An ANN is composed of layers of nodes. Neural networks acquire preferences by leveraging training data and improving their accuracy over time. Deep learning builds upon the foundation of machine learning, incorporating the concept of neural networks [5]. Using deep learning techniques, we can effectively process complex data and uncover deeply hidden features and the relationships between these features, thereby greatly enhancing data representation. In 2016, after years of continuous research in the video recommendation domain, YouTube integrated deep neural networks into its recommender system and applied them to video recommendations with outstanding results [6]. This achievement served as inspiration for an increasing number of recommendation models to incorporate deep learning techniques [7].

In an RS, 'item' denotes a product or service recommended by the system to its users. Recommendation of products to a user requires the analysis of the past preferences of similar users and leveraging item information. At the same time, users' preferences are dynamically evolving [8], so researchers employ time-based/sequential methods to model users' dynamic preferences and sequential patterns that change over time [9].

The integration of recommendation techniques with neural networks has paved the way for broader application prospects [10]. Users frequently interact with multiple platforms using various services. By utilizing cross-domain recommendations, insights from different domains can be leveraged to provide more comprehensive and relevant suggestions [11]. This approach effectively tackles data sparsity, mitigates the cold start problem, and captures a more holistic view of a user's multifaceted preferences. Some studies [12–14] have indicated that social relationships between users can be effectively used to alleviate issues such as the cold start problem. People have proposed integrating information-rich social networks with neural networks and incorporating the social–relational attributes of users or items (such as friendships and tag categorizations) into traditional recommendation frameworks, thereby overcoming the limitations of conventional recommendation algorithms. However, many recommendation algorithms still fall in terms of handling data sparsity and processing large-scale data, which to some extent compromises the precision of recommendations produced [15]. RS algorithms must have the ability to deal with highly sparse data and need to scale with increasing numbers of users and items. At the same time, they must provide satisfactory results and address other issues, such as data noise and privacy protection.

In the evolution of recommender systems, various challenges have arisen, such as information overload, cold start problems, long tail effects, and so on. In recent years, deep learning technology has emerged as a pivotal force in the field of recommender systems. Deep learning models, known for their exceptional performance, have become a hot spot in both academia and industry, offering solutions to many of the core challenges currently confronting recommender systems.

What makes deep learning compelling is its effectiveness in addressing the numerous challenges encountered by recommender systems. First of all, deep learning models excel at learning complex user and project features, making them adept at capturing subtle differences in user behavior. Second, deep learning models can handle large-scale data and are therefore able to cope with information overload. In addition, personalized recommendations that take advantage of potential representations of users and items can solve the cold-start problem to some extent. In addition, deep learning can improve the robustness of recommender systems and improve user experience. The reason why deep learning is so captivating lies in its effectiveness in addressing numerous challenges encountered by recommender systems. First, deep learning models can learn intricate features of users and items, allowing for a more precise capture of subtle variations in user behavior. Second, deep learning models can handle large-scale data, thus mitigating the issue of information overload. Furthermore, personalized recommendations based on the latent representations of users and items can, to some extent, alleviate the cold-start problem. Additionally, deep learning can enhance the robustness of recommender systems, improving the user experience.

In this study, we provide a comprehensive overview of recommender systems with a particular focus on their integration with deep learning. We introduce various types of recommendation methods, technological trends, and application domains that leverage deep learning. We offer detailed descriptions from four perspectives: content-based recommendation, sequence recommendation, cross-domain recommendation, and social recommendation. Additionally, we delve into the application of deep learning in recommender systems, including model principles, performance enhancements, and application cases, with the aim of providing readers with a deep understanding of this field. Furthermore, we discuss potential future trends in deep learning-based recommender systems, including model interpretability, multi-modal recommendation, privacy, and fairness considerations. The primary contributions of this paper can be summarized as follows.

- We present a comprehensive examination of recommender systems, with a specific emphasis on their integration with deep learning. We categorize them in terms of their developmental perspective, providing a comprehensive view of the evolution of the recommender systems field.
- We conduct a review of the research progress of recommender systems integrated with deep learning, focusing on methods for applying deep learning to collaborative filtering. Specifically, we perform a comprehensive analysis of four recommendation approaches that incorporate deep learning: content-based recommendation, sequence recommendation, cross-domain recommendation, and social recommendation.
- We identify future research directions in the field of deep learning-based recommender systems, contributing to the advancement of the research community.

Throughout this review study, in comparison to existing literature, we have pinpointed articles that excessively concentrate on content-based recommendation and collaborative filtering, with relatively less attention given to reviews concerning the direction of deep learning. Consequently, the following gaps have been identified: Despite the exceptional performance of deep learning models, their interpretability remains a challenge. This paper aims to explore ways to improve the interpretability of models to meet the requirements of both users and regulatory authorities. Future research could delve deeper into methods for enhancing model interpretability to bolster user trust. With the proliferation of multi-modal data, recommender systems need to better integrate diverse types of information. We believe that future research can explore methods for the more profound fusion of multi-modal data. Recommender systems should also make strides in addressing user privacy concerns and guaranteeing fairness in recommendation results. Future research should focus on how to protect user privacy and ensure fairness while providing personalized recommendations. In summary, this review aims to highlight these research gaps and offer valuable guidance for future research efforts in this field.

2. Related Work

Recommender systems are derived from advances in cognitive science, approximation theory, information extraction, and prediction theory. They have undergone rapid evolution since their birth and have become an important catalyst for the rise of the modern business economy [16]. These systems filter out redundant information from vast amounts of data, selecting items that can fulfill latent user needs. Traditional recommender systems typically focus on whether users have shown interest or rated an item [17]. By analyzing historical interaction data, these systems uncover the underlying demands, thereby achieving efficient recommendations.

Recommender systems are fundamentally based on exploiting binary relationships between users and items. By utilizing historical behaviors or similarity relations, these systems help identify items that might pique users' interests [18]. A utility function, denoted as "s", is employed to calculate the recommendation score for item "i" with respect to user "u". In this context, both users and items are characterized by a collection of distinct attribute features [19]. By calculating recommendation scores, the aim is to find the most interesting $\mathrm{i}' \in \mathrm{I}$ for every $\mathrm{u} \in \mathrm{U}$, as shown in Equation (1).

$$\forall \mathrm{u} \in U, i'_u = \mathrm{argmax} s(u,i) \quad (1)$$

In 2008, Ma et al. from the University of Hong Kong [20] took an innovative step by integrating user social interaction information with historical item rating data. This was a pioneering effort that introduced the SoRec recommendation algorithm, which was built on the foundation of probabilistic matrix factorization (PMF) [21]. Recognizing the increasing significance of social interactions in recommendations, research in this area witnessed exponential growth. In 2010, Jamali [22] delved deeper into the sphere of trust propagation among users and their social friends. They adeptly combined matrix factorization techniques with interactive trust, leading to the conception of the Social MF model. A year later, in 2011, building on the foundation laid by Social MF, Ma et al. utilized social regularization for recommendation [23]. This model was predicated on the assumption that the greater the influence of neighboring friends on a user, the more the user's latent preferences would converge towards that friend. When it comes to neural networks, Sedhain et al. of Australia's NICTA research center ingeniously applied an autoencoder model to recommender systems in 2015, giving birth to the single-hidden-layer neural network recommendation model known as AutoRec [24]. Using these operations, AutoRec has generalization and expression capabilities and it is the basic application of deep learning in this field. Pushing the boundaries further, Devooght and Bersini [25] achieved enhanced short-term prediction accuracy in 2017 by redefining collaborative filtering (CF) as a sequence prediction problem using neural networks. In summary, the evolution of recommender systems, as seen over the years, has seamlessly transitioned from leveraging social interactions to harnessing the prowess of deep neural networks. This evolution trend underscores the dynamic nature of research in this domain and the search for more refined and accurate recommendation techniques.

Graph neural networks (GNNs) form a collective term for models that apply neural networks on graphs, which have powerful graph-structured data learning capabilities. Graphs consist of nodes and edges connecting nodes and are typical non-Euclidean spatial data structures. Social networks are quintessential examples of graph data. Using GNNs, social network graph models can be seamlessly integrated into recommender systems. Hence, research on social recommendation has progressively become a focal point. By harnessing deep learning techniques to incorporate information from social networks, one can deeply understand the relevant features of users and items, achieving better accuracy of the recommendation while simultaneously improving user experience [26,27]. Based on this concept, GraphSAGE [28] is an inductive learning framework that efficiently generates embeddings for unknown vertices using their attribute information. This alleviates the issue of models being unable to directly generalize to vertices that have not appeared during the training process. Based on GraphSAGE, Ying et al. introduced an efficient GCN algorithm [29], they combine random walks with graph convolutions to produce node embeddings. GraphSAGE introduced a novel paradigm: inductive learning. Its strength lies in its ability to generalize from specific instances to broader contexts, discerning unknown data on unknown nodes. This advantage enables it to handle various scenarios in the industrial sector, where graph structures undergo dynamic changes.

The primary objective of this article is to provide scholars interested in this topic with an understanding of the main effects of employing deep learning methods in recommender systems. This work particularly delves into the motivations and developments associated with the application of various methods in such systems. Furthermore, it aims to offer solution-oriented perspectives that address the current challenges faced by recommender systems.

3. Overview of the Recommender Systems

Early recommender systems were primarily based on collaborative filtering and content-based recommendation. User-based collaborative filtering methods were introduced in 1992, while item-based collaborative filtering methods emerged in 1999. Content-based recommendation methods also began to appear during this period, utilizing item attributes and user preferences for recommendations. In the 21st century, with the rise of social networks and the development of deep learning technology, social recommendation methods gradually gained attention. Similarly, cross-domain recommendation methods also started to emerge, allowing the application of a user's historical behavior in one domain to recommendations in other domains. In recent years, with the proliferation of mobile devices and online shopping, sequence recommendation methods have become increasingly important. These methods consider user behavior sequences, such as purchase history or browsing history, to provide more personalized recommendations. Deep learning methods can handle large-scale data and multi-modal information, enhancing the effectiveness of recommendations. Reinforcement learning methods have also been applied to recommender systems, optimizing recommendation strategies via user interactions. The development history of recommender systems can be divided into several key stages, each producing different types of recommendation methods. A summary of some important recommender system methods is provided in the following Figure 1.

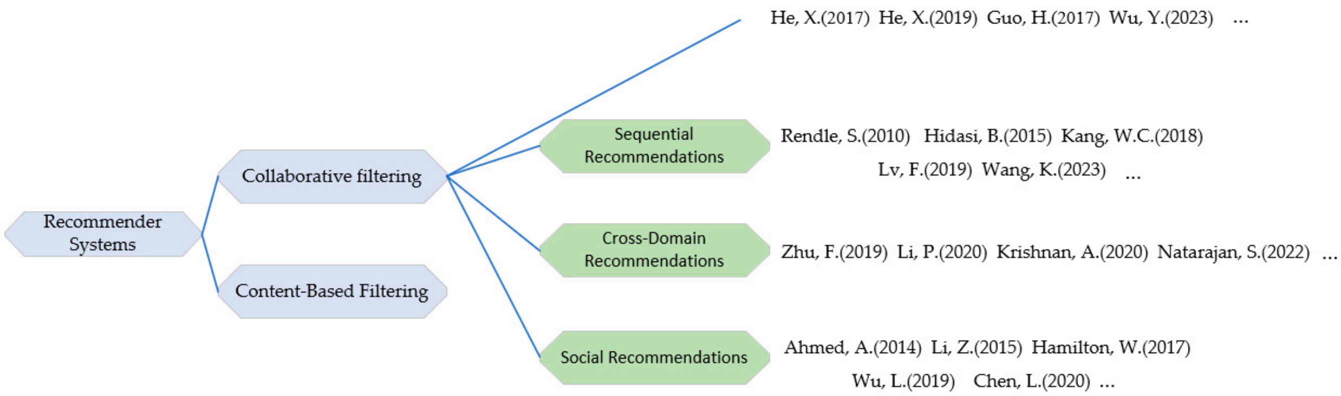

Figure 1. The development and classification of recommender system methods [28,30–46].

New technologies, compared to old ones, are not a complete revolution. New technologies are built on the foundation laid by their predecessors, incorporating improvements rather than being entirely different. The field of recommender systems is continuously evolving, with new methods and technologies emerging to adapt to evolving user demands and data environments. Currently, efforts are underway to refine the integration of deep learning for improved recommendation results. Therefore, the journey of recommender systems is a dynamic and evolving process. We will begin by introducing the most fundamental approaches, combining deep learning with collaborative filtering and content-based recommendation.

Collaborative filtering is one of the first and fundamental recommender systems. In 1992, GlodBerg et al. [47] introduced the concept of collaborative filtering. The key idea lies in finding the nearest neighbors using the interaction matrix of users and items, and then generating various forms of recommendations based on the information from these neighbors, which is essentially a process of matrix completion. The core concept of collaborative filtering algorithms is to provide product recommendations or predictions based on similar users or similar items, encompassing two dimensions: neighborhood-based and model-based approaches. The key to collaborative filtering recommendations is the computation of similarity between different users and different items.

Over the past three decades, collaborative filtering has been widely used by people. However, as time has progressed, its scalability limitations have become apparent, making it less suitable for scenarios with large datasets. People have turned to machine learning and data mining algorithms to learn and recognize complex patterns using training data. Currently, one of the most common paradigms for building deep learning recommendation algorithms is the use of multi-layer perceptions, which can introduce nonlinear transformations into existing collaborative filtering methods.

Since then, the research of recommender systems has shifted towards deep learning-based recommendation [48]. To address the complexities of user behavior, neural network-based models have become increasingly favored by researchers [49]. Among them, neural collaborative filtering (NCF) stands out as a notable attempt [30]; this method enhances the expressiveness of the developed model by expanding the inner product within a multilayer perceptron (MLP). The success of this approach has further motivated researchers to explore the integration of deep learning techniques using traditional recommendation algorithms.

This trend is effectively illustrated in the deep factorization machine (DeepFM) [31], which cleverly integrates a shallow factorization machine (FM) [50] with an MLP. This approach effectively combines the FM and deep components to achieve a unified representation of both low-order and high-order feature interactions, significantly enhancing the model's expressive power. This approach is similar to traditional machine learning, but its training is divided into two phases. Therefore, it can converge better and faster.

Wu, Y et al. proposed FedDeepFM [32], a model grounded in federated learning, which offers a mechanism that delivers high-quality recommendations while enhancing user privacy. This approach is particularly suitable for scenarios that prioritize privacy preservation or data transmission reduction. It generates training data using real data combined with interaction-based synthetic data. Built upon DeepFM, FedDeepFM offers a mechanism for delivering high-quality recommendations while enhancing user privacy.

The "multi-layer neural network + output layer" allows for a more extensive interaction of user/item vectors, introducing additional non-linear features and enhancing the learning capability for sparse features. By incorporating attention mechanisms and introducing attention layers between the embedding layer and the multi-layer perceptron, it has evolved into deep interest networks (DIN) [51]. Furthermore, there are variations such as deep interest evolution networks (DIEN) [52], which incorporate sequence models to simulate changes in user preferences, and multi-interest networks with dynamic routing (MIND) [53], which utilize capsule networks to extract diverse user interests and introduce tag-based attention mechanisms for dynamic path selection.

As one of the most widely used recommendation algorithms at present, collaborative filtering exhibits strong generality and significant effectiveness. However, collaborative filtering always faces serious issues of data sparsity and cold start. The integration of deep learning with collaborative filtering recommender systems not only simplifies engineering implementation but also alleviates these problems. Content-based recommendation, which coexisted with collaborative filtering, has also evolved in a similar manner.

3.1. Content-Based Recommendation

Content-based recommendation is the earliest and most basic type of recommendation algorithm, and it has played a pivotal role in the entire history of recommender systems [54]. Although its effect may not be as good as that of new recommendation algorithms, it is still very valuable and even essential.

Content-based recommendation algorithms form a relatively intuitive category of algorithms that primarily offer recommendations by comparing item features with users' historical preferences. In the early days, it relied primarily on specific content, such as the characteristics of text, images, or audio. Their approach entails suggesting items to users that exhibit content similarity to previously preferred items, determined by the historical preferences of these users [55,56]. A content-based recommendation model is constructed based on the target item, user information, and interactions between the user and items. Importantly, it notably neglects the actions of other users. A schematic process is shown in Figure 2 below.

To achieve content-based recommendation, the general steps include (1) constructing an item feature representation based on the content features of each item; (2) building a user feature representation based on user characteristics and behaviors; and (3) generating a recommendation list according to the level of alignment between the features of the target item and those of the user [57]. When integrated with neural networks, the entire recommendation model can be divided into an input layer, a hidden layer, an output layer, and a prediction layer. A content-based recommendation model is shown in Figure 3.

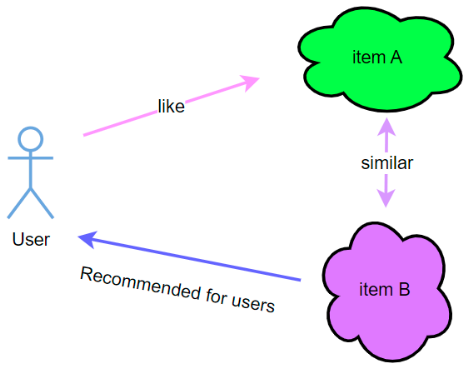

Figure 2. Diagram of a content-based recommendation algorithm.

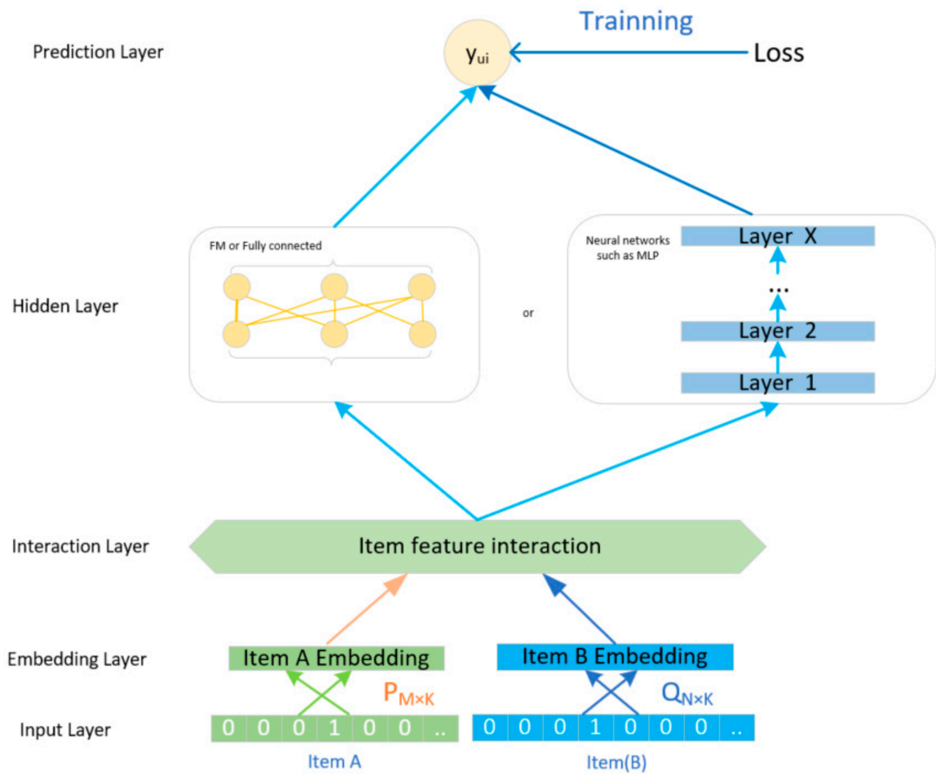

Figure 3. Content-based recommendation procedure.

The hidden layer is a fully connected layer consisting of 1–2 sublayers, and the score of the item is output [58]. This model highlights the embedding and transformation of item features. The transfer process from the input layer to the next is represented as shown in Equation (2).

$$h = f\ (\ W_{ih} x + b_{\ h}) \tag{2}$$

The transfer process between hidden layers is depicted as shown in Equation (3) (if multiple hidden layers are present).

$$h_j = f(W_{hj} h_{j-1} + b_j) \tag{3}$$

The transfer process to the output is shown in Equation (4).

$$y = f(W_{ho} h + b_o) \tag{4}$$

where x represents the input and y signifies the output. w and b are parameters, while f is a nonlinear activation function.

The process of representing items and users via tagging is becoming increasingly popular. Tags can serve as content information and be used in content-based recommendation methods [59], or they can be treated as a third entity, utilizing collaborative filtering approaches for recommendation. However, as data representations become more diverse, the preprocessing tasks for items become increasingly challenging. A one-dimensional vector representation can be overly simplistic, thereby neglecting the intrinsic relationships between various attributes of the given content. A tensor is a multidimensional data structure that is capable of representing the co-occurrence relationships of multiple types of attributes. Tensor decomposition is fundamentally a feature extraction method, and many algorithms that leverage tensor decomposition for model construction have been developed in the past. Peng et al. [60] constructed a tensor for user–item–tag triples and employed Tucker decomposition to obtain abstract representations of users. The core tensor obtained from the decomposition procedure encapsulates the associations between different attributes, thereby taking the cooccurrence relationships between items and tags into consideration.

Unlike traditional methods such as prod2vec, which typically only considers user interactions to create embeddings, Meta-prod2vec not only utilizes tags but also incorporates additional information, such as product descriptions and reviews, to generate richer product embeddings [61]. It utilizes product-side information to produce product embeddings. By integrating meta-information with interaction data, new embedding vectors are generated for products. The future of content-based recommender systems is inextricably linked with advancements in deep learning. Furthermore, feedback loops derived from users can enhance the adaptability of the utilizing model, ensuring continuous user profile learning and updating.

When dealing with large and complex user-item interaction matrices, traditional matrix factorization models tend to underperform [62]. The paper titled "Deep learning" [63], published in *Nature*, serves as a fundamental contribution to the field of deep learning. It presents the concept of deep learning by demonstrating how computational models with multiple processing layers can effectively represent data with varying levels of abstraction. These methods achieve significantly improved performance in various tasks. The paper delineates the fundamental equations of neural networks, backpropagation, CNNs, and RNNs, offering readers a comprehensive overview of the domain's evolution, key concepts, and challenges.

Currently, the content-based recommendation is extensively employed in industrial-scale recommender systems, as it possesses the following advantages.

1. Personalized recommendations: These recommendations are based on the user's historical interests, ensuring that the recommended content aligns with the user's preferences.
2. Simple principle with strong interpretability: Content-based recommendations can be made based on label dimensions or by embedding items into a vector space using similarity, making this strategy easy to implement. It is also readily accepted and validated by users.
3. Addresses the cold-start problem to some extent: As long as sufficient content attributes are available, new items can be effectively handled without relying on other users' behaviors.

However, some of the drawbacks of content-based recommendation led to limitations in its effectiveness and scope of application. It has a narrow recommendation scope, and its novelty is not pronounced. The results obtained from content-based recommender systems tend to converge on categories of items that the user has previously shown interest in. New users, without sufficient interaction histories, might not receive effective recommendations. Moreover, the comprehensiveness, integrity, and accuracy of content understanding can impact the efficacy of the recommendation process. Additionally, based on practical experience, the recommendation accuracy of this approach is not particularly high.

Overall, content-based recommendation is a fundamental technique within recommender systems. It suggests items by analyzing item feature vectors and user interest vectors. The aim is to propose new items to users that match their preferences. But people's preferences change over time. So, sequence recommendations came into being.

3.2. Sequential Recommendation

Traditional recommender systems focus on identifying static connections between users and items, often neglecting dynamic shifts in user preferences over time. Sequential recommendation approaches posit that the recommendations given to a user at a specific moment should be determined based on the user's prior behaviors prior to that moment. Sequential recommendation captures the dynamic nature of user behaviors, revealing certain relationships between items over a given time span. Sequential recommender systems can capture dynamic changes in user interests by modeling their interaction sequences. The issue of sequential recommendation entails time-based learning to anticipate user interactions with items. The aim of sequential recommendation is to explicitly model users' sequential behaviors. We delve into the intricacies of sequence-based recommendation models. To offer a more intuitive understanding of this concept, Figure 4 presents the architectural diagram of a typical sequence-based recommendation model.

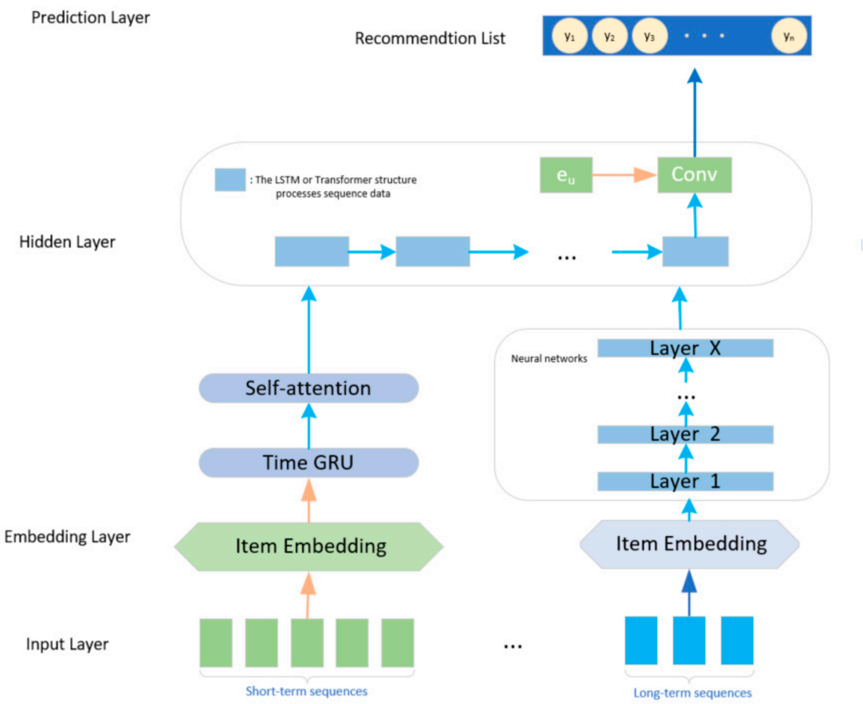

Figure 4. Sequential recommendation procedure.

As illustrated in Figure 4, sequence-based recommendation models are capable of capturing the dynamic changes in user behaviors, thereby generating more personalized and timely recommendations. In its input layer, such a model receives a sequence of user behaviors, which commonly represent the user's historical interactions. These sequential data are subsequently encoded into high-dimensional vectors. Advanced feature extraction is then performed via attention mechanisms or neural networks. Following this step, in the hidden layer, a range of more complex network architectures, such as long short-term memory (LSTM) or transformer models, are employed to capture the temporal dependencies and other latent patterns within the sequence. Finally, the output layer typically comprises a fully connected layer and an activation function, employed to compute scores or probabilities for each potential recommended item.

In summary, the development of deep learning in recommender systems has witnessed an orchestrated blend of innovations, each building upon its predecessor. These advancements, rooted in complex mechanisms and models, have progressively enhanced the precision and context awareness of recommendations. Sequence recommender systems

(SRS) can be categorized into three groups: standard sequence recommendation, long-short term sequence recommendation, and multi-objective sequence recommender systems.

3.2.1. Standard Sequence Recommendation

Standard sequence recommendation refers to recommendation algorithms that extract user representations using a single behavior sequence [64]. Traditional sequence recommendation algorithms incorporate various prevalent sequence modelling techniques, such as pooling, RNNs, CNNs, memory networks, and attention mechanisms. This approach has been employed in Google's recommendation model. Due to its simplicity and efficacy, it stands as one of the most prevalent techniques for leveraging sequence features. Neural networks have also found extensive applications in sequence recommendation tasks.

In the initial stages of sequence-based recommendation models, the primary methods employed are Markov chains [33] and feature-based matrix factorization [65,66]. Compared to deep learning techniques, Markov chains have limitations in terms of handling complex data. Researchers employ personalized Markov chains to model individual users, leading to enhanced efficiency. Some people have pointed out that the Markov chain approach underperforms when modelling joint-level sequence patterns and fails to capture the effects of skip behaviors between users and items on recommendation outcomes. While factorization-based methods might model sequences by summing up sequence item vectors, they neglect the influence of order within the user interaction sequences on the recommendation results. Traditional sequence recommendation methods primarily utilize Markov chains and their derivatives for predicting a user's next action. They operate under the assumption that a user's subsequent action is solely dependent on their recent few actions.

Compared to traditional recommendation models, deep learning models have demonstrated superior capabilities with respect to capturing the evolution trends of user sequences. The evolution of deep learning in sequential recommendation can be delineated via a series of transformative advancements. Recurrent neural networks (RNNs) have demonstrated their potent capabilities [67] and have been extensively utilized in sequence-based recommendation. Sequence recommendation models based on RNNs can be represented as Equation (5).

$$h_t = \sigma\left(W_{hh}\, h_{t-1} + W_{hx} x_t + b_h \right) \tag{5}$$

where x_t represents the input and h_t denotes the hidden state. Finally, a different activation function for output is selected. Some people introduced CNNs to sequence recommendation tasks [68]. They underlined the fact that the existing Markov chain models can only model point-level sequential patterns and fall short when representing set-level patterns. They employ convolutional layers to extract local features from the user's historical behavior sequence, capturing these features from different embedding perspectives (vertical and horizontal). Different convolutional kernels are utilized to extract features from the input sequence. These features are subsequently used to generate the final recommendation list. An attention mechanism offers a notable solution to this gap.

Based on this foundation, SASRec [34] introduced a sequence recommendation method founded on self-attention to improve its effect. Their approach employs a multihead self-attention mechanism and positional encoding to handle the input user behavior sequence. However, for an extensive sequence, earlier interactions might be overlooked. Consequently, researchers have begun to investigate recommendation models that integrate both long- and short-term interactions.

3.2.2. Long- and Short-Term Sequence Recommendation

The evolution of long- and short-term sequence recommendation reflects the progression of recommender systems from simple collaborative filtering methods to sophisticated deep learning models. With continual advancements in technology, models are becoming increasingly adept at distinguishing and integrating two user preferences, thereby facilitating more accurate and personalized recommendations.

Long-term preferences reflect their relatively stable interests, while their recent preferences indicate their recent interest changes [69]. Although user preferences change dynamically over time [70], some long-term stable preferences that influence user behaviors remain. The existing approaches tend to combine two preferences [71].

Taking into account that interactions between two behaviors may exert diverse effects on the user's present interests [72]. Compared to recent interaction sequences, long-term interactions tend to evolve more slowly [73], thereby reducing the demand for real-time responsiveness [74]. SHAN [75] categorizes user behaviors into two behaviors, employing a multilayer attention network for modelling purposes. By coupling users' two preferences, an optimal user representation is generated, enhancing the output recommendation results. This hybrid representation is calculated as shown in Equation (6).

$$u_t^{hybrid} = \beta_0 u_{t-1}^{long} + \sum_{j \in S_t^u} \beta_j \mathrm{v}_j \tag{6}$$

where $\mathrm{v_j}$ represents the rating embedding of an item, which is indicative of the short-term embedding, while $\mathrm{u_{t-1}}$ denotes the long-term embedding. β denotes varying attention scores. This formulation captures the dynamic nature and differentiates between the contributions of items towards predicting the next.

RNNs encounter challenges like gradient vanishing and gradient explosion when dealing with lengthy sequences, diminishing their proficiency in handling extended sequences. To tackle this problem, long short-term memory networks (LSTMs) [76] were initially introduced, introducing the fundamental concept of gating mechanisms that effectively addressed the issue of vanishing gradients.

Recommender systems often use LSTM or gate recurrent units (GRUs) for user modelling. LSTM is an enhanced version of an RNN [35]. It has been demonstrated that LSTM outperforms conventional RNNs in sequence recommendation tasks. A GRU, another variant of LSTM, is computationally simpler yet equally effective. Recognizing that the two-term impacts of preferences might differ, attention mechanisms have also been employed. The attention mechanisms used in deep learning stem from the idea that humans are drawn to the significant parts of a target. Their origin can be traced back to the research performed by Bahdanau et al. [77]. They used attention mechanisms to model the importance of the output using different parts of the input sentence. Building upon this foundation, vanilla attention was introduced to function as an RNN decoder, and it has been extensively used in sequence recommendation [78]. However, the self-attention mechanism (originating from the transformer in Google's 2017 neural machine translation work [79]) has also been deployed in sequence recommendation. Contrary to vanilla attention, self-attention does not involve an RNN structure, but it has achieved better performance in recommender systems than RNN-based models [80].

Building upon this idea, a GRU [81] offered streamlined versions of the LSTM gating processes. Venturing into the realm of natural language processing, the transformer model [82] gained widespread acclaim. It established unparalleled standards in terms of discerning the dependencies between sequences, which paved the way for swift parallel computations and accelerated sequence information extraction [83].

This foundation was ingeniously adopted in Alibaba's DIN model. The self-attention mechanism of this model was integrated within the recommendation domain, amalgamating both fundamental user details and context information. SDM [36] takes a nuanced approach by segregating user interests. It employs LSTM, complemented by multi-head attention mechanisms, to home in on immediate user inclinations. The culmination of this model witnessed the amalgamation of both fleeting and lasting user interests, encapsulating the user's essence in a vector representation. Compared to previous models, this approach employs gating mechanisms to act as weights, ultimately outputting a user behavior vector, as shown in Equations (7) and (8).

$$G_u^t = sigmoid\ (W^{\,1} e_u + W^{\,2} s_t^u + W^{\,3} p^u + b) \tag{7}$$

$$o_t^u = (1 - G_t^u) \odot p^u + G_t^u \odot s_t^i \tag{8}$$

Here, e_u is the user embedding, while s_u and q_u are data derived from user embeddings processed via different attention networks. This facilitates a more effective data processing strategy. Unlike prior works that simply used parameters as weights, this current approach enables short-term behaviors to be better integrated.

Finally, the Trans2D model [84] showcased a pioneering expandable attention mechanism. This module adeptly discerns user predilections for specific item attributes, which are gleaned from their behavioral sequence data, incrementally elevating the efficacy of the recommendation process. Compared to recommendation algorithms based on RNNs, those relying on convolutional neural networks (CNNs) present greater challenges. Tuan et al. [85] devised a 3DCNN model for high-dimensional modelling, demonstrating its efficacy when the given user sequences are associated with intricate element features. Yan et al. [86] encoded interaction sequences into three-dimensional vectors, employing 2D convolutional filters to learn local characteristics.

Tang and Wang et al. [87] proposed the Caser model, building upon the foundations of prior convolutional techniques. They employed both horizontal and vertical convolutional filters to capture continuous patterns in pointwise, conjoint, and skip behaviors, achieving promising results. The vertical convolution operation is shown in Equations (9) and (10).

$$c_i^k = \Phi_c(E_{i:i+h-1} \odot F^k) \tag{9}$$

$$\widetilde{c}_i^k = \sum_{l=1}^{L} \widetilde{F}_l^k \cdot E_l \tag{10}$$

where E serves as a matrix formed by stacking embeddings to facilitate the convolution operations. F^k slides over E from top to bottom during the horizontal convolution process, capturing the user's behavioral patterns. Behavioral patterns are how users interact with different items at different points in time. The vertical convolution operates in the embedding space, focusing on how to group or categorize different items based on their embedding representations. By combining these two types of convolutions, the model is capable of capturing user behavior patterns in two distinct dimensions. It also adapts the embedding representations according to the dynamics of the user behaviors. However, this approach separately learns embeddings for users and items, failing to capture inherent interaction information and overlooking contextual details.

GRU4Rec [37] integrates an RNN into session-based recommender systems, treating the interactions within a session as the historical context for sequence modelling. The highlight of this work is its rational optimization of sequence recommendation based on an RNN. With continued research advancements, a multitude of improved models have emerged.

These models aim to delve deeper into item information by optimizing their model architectures and training methodologies, thereby enhancing the produced recommendation outcomes. One such enhanced RNN model evolved from GRU4Rec employs data augmentation techniques to bolster the stability of the model training process [88]. Another modified RNN model adjusts for batch generation by considering dwell time to more precisely capture user behaviors, discovering the potential impact of dwell time on the results [89]. However, a significant challenge is that when all candidate items rank above popular items based on a popularity-driven sampling method, the learning speed of the developed model is constrained, particularly when recommending long-tailed items. Addressing this issue, an RNN model with top-k gains [90] introduced a novel sampling strategy for GRU4Rec, blending uniform sampling with popularity-based sampling to markedly improve performance. To enhance the modeling of item information, not only was the item's ID considered but other features such as text descriptions and images were also incorporated. This was effectively realized using a parallel RNN (p-RNN) architecture for efficient recommendation [91]. In terms of training strategies, the authors innovatively introduced methods such as alternating training, residual training, and interlaced training to optimize the performance of the p-RNN. Furthermore, Dietmar explored multiple approaches to combine a session-based KNN classifier with GRU4Rec, such as switching, cascading, and weighted blending, further enhancing the accuracy of the output recommendations [92].

The RUM model [93] incorporates a user memory module to preserve such interaction information. Utilizing a dual-layer memory mechanism, the model employs an external memory mechanism to simulate the user's memory process, thereby capturing both the long-term and short-term preferences of the user. Recurrent structures are employed to capture the user's recent activities, whereas an external memory unit stores the user's historical behaviors. Attention mechanisms are applied to select the most relevant historical information. A multilayer network architecture is finally used to capture deep interaction information. By dynamically updating the memory unit, the model reflects user preference changes, as indicated in the following Equation (11).

$$add_i = \tan\,\mathrm{h}\,(A^{\,T} q_{\,i} + b_a), m_k^u \leftarrow m_k^u + z_{i\,k} \cdot add_i \tag{11}$$

The above equation is utilized to update the user's preference memory. A and $\mathrm{b_a}$ act as trainable additive parameters with the aim of erasing existing information prior to the incorporation of new data. Transformers have achieved significant breakthroughs in NLP tasks, with substantial pretrained models such as BERT leading the way. Bert4Rec [94] adapts this architecture for recommender systems.

To achieve optimal recall results, it is essential for the employed system to account for both the long- and short-term interests of the target user. In practical applications, recommendation models can discern these two preferences from the user's historical sequence and ultimately merge them using gated unit modules.

3.2.3. Multi-Objective Sequence Recommendation

Researchers have recognized that in addition to relevance, diversity and novelty are also important objectives in recommender systems. This implies that these systems should not only recommend items that users are likely to enjoy but also those they have not previously encountered. In the realm of multiobjective sequence recommendation, understanding and modelling user behaviors is one of the core tasks. Traditional methods often overlook the personalized needs of users when addressing different learning objectives [95]. As time has evolved, researchers have started to realize that user interests are not monolithic and often span multiple domains, prompting the emergence of a series of methods. These methods attempt to capture this diversity by encoding the target user's sequential behavior into different interest representation vectors [96]. In simple terms, the input is subjected to grouped convolution operations, which are determined by coupling coefficients obtained via an iterative dynamic routing process and subsequent weighted embedding vectors. Through such convolutional operations, each group of outputs can be regarded as an embedding vector corresponding to a specific interest.

Complexity increases when users exhibit a diverse array of behavioral sequences in real-world applications, such as clicks, shares, and purchases. Researchers need to model multiple types of behavioral sequences to uncover users' genuine preferences more comprehensively [97]. This issue also underscores the critical importance of explicitly considering users' historical interactions to enhance performance [98]. Further studies have revealed that the actions within a user's behavioral sequence are often heterogeneous and polysemic [99,100]. To capture the underlying intents of these actions more effectively, researchers have started assigning varying weights to different types of behaviors. For instance, a purchase action is typically considered a stronger indicator of a user's genuine interest and preference for a product or service compared to a mere click action. Therefore, distinguishing different types of behaviors becomes particularly critical [101]. This concept has been concretely implemented in the CBS model. CBS categorizes the given behavioral sequence into target sequences and supporting sequences based on the types of behaviors observed [102]. This enables the target sequences that are closely related to the most predictive types of behavior (e.g., purchasing) to be highlighted. The authors proposed a novel approach that integrates a basket representation into a recurrent layer to capture sequential effects. The implicit recurrent representation of h_t is presented as shown in Equation (12).

$$h_t = g(\Phi_b\, b_t + \Phi_h h_{t-1} + \Omega_h) \tag{12}$$

This approach takes both temporal ordering and continuity into account. The formula explicitly incorporates basket representation, placing particular emphasis on the influence that user choices at specific time steps exert on the hidden state. This distinguishes it from more traditional sequence recommendation methods.

A similar line of thinking has also been implemented in BINN [103]. BINN aims to capture users' current interests by utilizing all types of behaviors, such as clicks, purchases, and favorites, while exclusively employing behaviors related to purchases (e.g., buying, adding to cart, and favoriting) to reflect users' long-term preferences.

In another study [104], the authors attempted to integrate the specific representations of each type of behavior with the corresponding item embedding vectors, aiming to capture users' interests more comprehensively. The researchers integrated a masked beam search and determinantal point process (DPP) selection to produce a high-quality and diversified bundle list with an appropriate bundle size. The formula is presented in Equation (13).

$$masked\_soft\max(h_t, E, m_t)_j = \frac{\exp(h_t^T e_j - m_{t,j})}{\sum_{\hat{j}=1}^{N} \exp(h_t^T \hat{e}_j - m_{t,\hat{j}})} \tag{13}$$

Specifically, the beam search retains the top k most promising candidate bundle lists in each step and subsequently extends these lists in the next step to progressively construct longer recommended bundle lists. This paper employed a beam search to generate more accurate and relevant bundle recommendation lists while also ensuring the computational efficiency of the model.

After all, with the continual advancements achieved in deep learning and natural language processing technologies, sequential recommender systems are well-primed for further optimization and development. More advanced model architectures, such as transformers and BERT, can be employed to capture complex sequential patterns. Moreover, integrating these techniques with other recommendation approaches, such as those based on knowledge graphs, may enhance both the diversity and accuracy of sequential recommender systems. As the Internet of Things and smart home technologies continue to proliferate, the utilization of sequential recommendation is expected to further broaden in everyday life.

Recent research on sequential recommendation remains highly active, with some researchers considering temporal information in sequential recommendation [105]. They aggregate sequential information and collaborative signals in user behavior sequences, thereby taking a more comprehensive approach to considering the information in sequential behaviors. In order to enhance the transition probabilities between items in sequences, researchers employ contrastive learning for sequential recommendation [106]. They introduce two informative augmentation operators leveraging item correlations to create high-quality views for contrastive learning.

Traditionally, recommender systems have often employed static strategies to characterize the interactions between users and items, relying on long-term historical behaviors to infer user interests. However, the selection of a particular item by a user is not solely based on their long-term stable preferences but is increasingly driven by their fluctuating short-term interests [107]. In fact, user interests dynamically evolve over time. This is precisely where sequential recommendation comes into play. Sequential recommendation models the user-product interaction history as a dynamic sequence and utilizes the temporal dependencies within this sequence to capture the evolving user preferences, leading to more precise and timely recommendations. Deep learning enables the hierarchical treatment of complex problems, swiftly identifying the latent patterns and relationships between different layers of data. The integration of deep learning with a sequential recommendation can also adjust recommendations based on dynamic user preference changes, further enhancing their effectiveness [108].

Sequential recommender systems excel at accurately predicting users' immediate interests and dynamically adapting to their behavior changes and shifting preferences. This approach not only guarantees content diversity but also furnishes recommendations that are highly pertinent to users' recent activities, even if these suggestions may not fully correspond with users' earlier behavioral patterns. While sequential recommendation undoubtedly offers unique advantages, it is important to mention that handling time-series data can introduce additional computational demands on the model [109]. Furthermore, an undue emphasis on users' short-term actions carries its own set of risks, potentially leading to a neglect of their long-term interests and preferences.

3.3. Cross-Domain Recommendation

In the preceding sections, our discussion was primarily focused on generating recommendations based on users' historical behaviors and content characteristics, usually within a single application scenario or domain. However, to reap benefits from a variety of services, users frequently engage with multiple social media platforms [110]. Cross-domain recommendation (CDR) has been introduced to utilize richer information from multiple domains, aiming to improve recommendation performance. Studies indicate strong correlations between user-generated data across different domains [111]. Traditional data mining techniques are primarily designed for single-domain analysis, and by overlooking data from other domains, they often encounter issues related to data sparsity [112]. Consequently, addressing the utilization of cross-domain data to enhance the comprehensiveness and accuracy of recommendations has become a prominent research focus. By aggregating data from different domains, CDR methods not only compensate for the missing information in a single domain but also holistically harness the value embedded in social media data. The framework is shown in Figure 5 below.

As depicted in Figure 5, the input layer is designed to accommodate user behavior and item attributes from disparate domains. After performing feature extraction, the data acquired from these diverse domains are aggregated into a shared hidden layer, which is tasked with capturing cross-domain patterns and relationships. Within the hidden layer, neural networks can be employed to integrate information from various domains. By employing this design, the model can generate an output layer that synthesizes information from multiple domains, thereby facilitating more comprehensive and accurate recommendations.

Cross-domain recommender systems aim to improve recommendation performance [113] by utilizing data from various domains, particularly in scenarios with data sparsity or cold start challenges. These systems take users' behaviors and preferences across different domains into account, harnessing comprehensive multidomain information to provide a more holistic user profile and, in turn, bolster recommendation accuracy. Cross-domain recommender systems exploit multidomain learning and knowledge transfer [114], diversifying their recommended content to satisfy the needs of varied scenarios and domains. However, multidomain data processing and the associated knowledge transfer step may also result in increased computational costs.

Traditional recommender systems grapple with two main challenges: the cold start problem and data sparsity [115]. CDR methods offer an effective remedy for these issues by analyzing user interactions from other domains to capture user preferences from specific perspectives. This can be employed to enhance data in the target domain or supplement information during the initialization process for new users. While traditional recommender systems build and analyze a recommendation model within the current domain, cross-domain recommendations necessitate deciding which information should be transferred between distinct domains and the means of performing such transfers. Cross-domain recommendation approaches can harness user interaction data from other domains for auxiliary analysis purposes, capturing certain user preferences. This enables enhanced recommendations to be produced in the target domain or even across multiple domains and provides additional information for new user initialization, thereby addressing the two challenges that are endemic to traditional recommender systems.

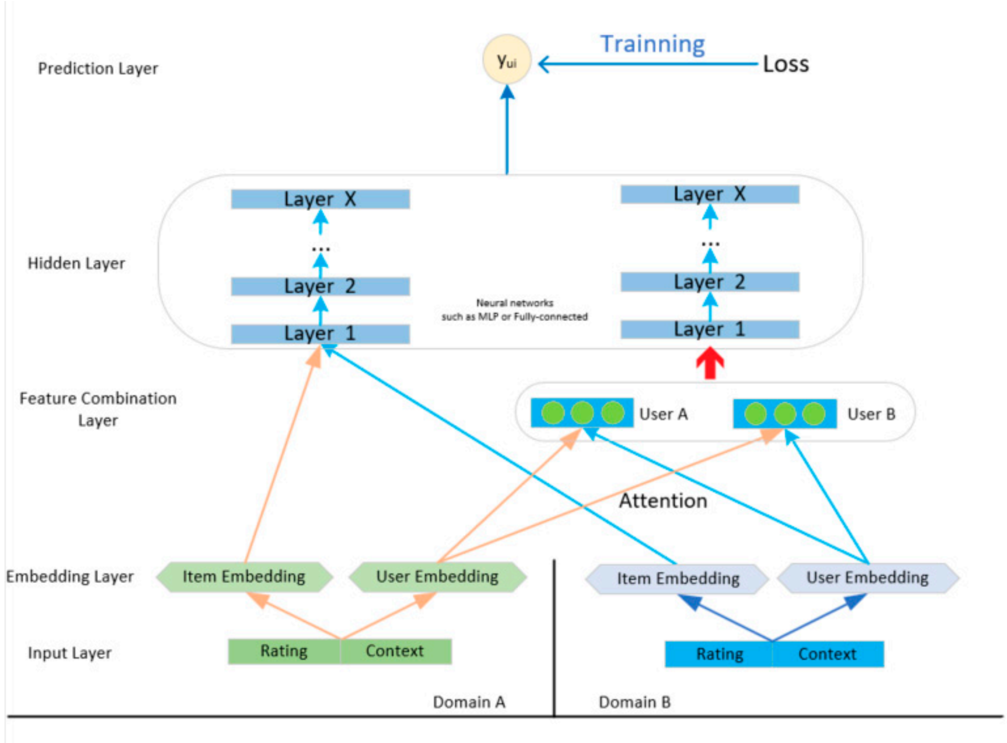

Figure 5. Cross-domain recommendation procedure.

In recent years, researchers have focused on addressing data sparsity and cold-start issues in the target domain for cross-domain recommendation. They have conducted extensive studies on deep learning-based cross-domain recommendation methods. These approaches consider deep auxiliary information, including neighboring domain descriptions and rating information [116–118]. For instance, a multilayer perceptron (MLP) utilizes the latent features of common users in the source domain as inputs and generates latent rating features for those users in the target domain as outputs during the integration step. Through this process, the network is trained, obtaining a nonlinear mapping function. Optimization is finally achieved via backpropagation, resulting in improved recommendation outcomes. Later, Natarajan, S et al. proposed CD-SemMF [38], which uses the semantic relevance of a project to obtain better results.

In the realm of CDR, multiple domains are encountered, each with an information volume denoted as *IN*. Within this framework, Domain A possesses dense data, and a newer Domain B is characterized by data sparsity and suboptimal recommendation performance. The inequality $INA \gg INB$ suggests that the information volume in Domain A significantly surpasses that of Domain B. This discrepancy prompts the consideration of whether information from Domain A can be transferred for utilization in Domain B, a process termed cross-domain transfer. Domain A is designated as the source or auxiliary domain, symbolized as D_S, whereas Domain B is identified as the target domain, denoted as D_T. The source domain can comprise multiple distinct domains, which are aimed at supplementing and enriching the information within the target domain D_T. The respective user sets in domains S and T are U_S and U_T, the item sets are I_S and I_T, the rating sets are RS and RT, and the comment sets are S and C_T, respectively. Thus, the objective is to transfer the knowledge embedded in the source domain to enhance the recommendation quality of the target domain. The distinction between cross-domain recommendation and other methods lies in its feature input, which consolidates multiple domains. Let V represent the user's behavioral features; user embedding for cross-domain recommendation is illustrated in Equation (14).

$$u_i = [w_1 \cdot f(u_i^s) + w_2 \cdot u_i^t, v_1^h, v_2^h, \ldots, v_n^h] \tag{14}$$

Cross-domain recommender systems comprise three pivotal modules: a user modelling module, an item modelling module, and a recommendation algorithm module, as illustrated in Figure 6.

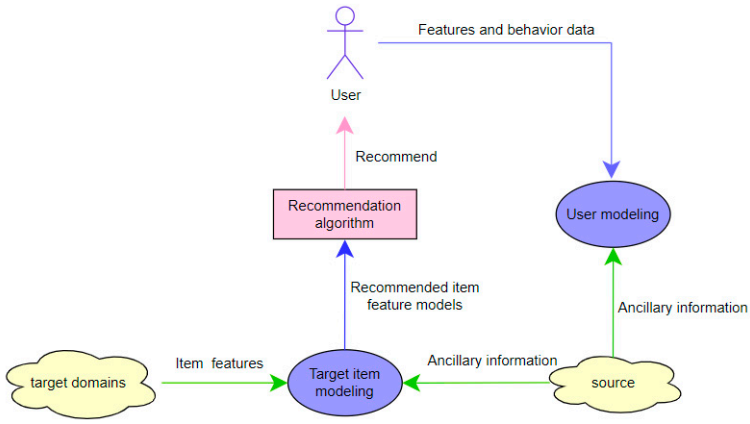

Figure 6. Process of the cross-domain recommender system.

CDR seeks to merge data from various domains by integrating supplementary information from other domains to improve recommendations within the target domain or even across multiple domains. However, it is not a matter of arbitrarily combining any group of domains for cross-domain recommendation purposes; typically, there should be some overlapping information among the different domains. Compared to conventional recommender systems, CDR necessitates meticulously considering which information should be transferred between domains and the methodology used for such transfers.

CDR tasks are closely linked to user-related aspects, with two fundamental factors driving this connection: the range of items for recommendations and the diversity of target users, giving rise to a variety of recommendation scenarios. CDR algorithms can be classified according to the target domain into three categories: single-domain CDR, cross-domain CDR, and multi-target CDR.

3.3.1. Single-Target CDR

Single-target CDR is the conventional approach in the CDR domain, and most CDR methods primarily focus on this scenario. It involves recommending items to users in the target domain by utilizing information acquired from the source domain. Single-objective CDR methods predominantly focus on transferring valuable knowledge.

There are also various recommendation methods, it initially establishes connections based on common content elements such as user/item attributes [119], 'like' data, and browsing or viewing histories [120]. Subsequently, they employ these connections for the cross-domain transfer of user/item data or knowledge. Embedding-based transfer methods often project instances from both the source and target domains into a novel feature space, where instances from both domains display similarities. Other methods often adopt distinct techniques or philosophies, including Bayesian latent factor models, interest drift, ternary relations, and reinforcement learning. Rating pattern-based transfer methods typically begin by learning the distinct rating patterns of users from the source domain. Subsequently, these patterns from the source domain are harnessed to enhance the model's recommendation performance in the target domain, facilitating knowledge transfer. Shapira et al. and Tiroshi et al. [121] both focused their efforts on leveraging social networks to enhance recommender systems. While the former harnessed Facebook's friend relationships to reinforce user models in the target domain, the latter delved deeper, employing random walk algorithms to extract latent user information within social networks. These approaches can be perceived as endeavours to uncover both explicit and implicit user relationships within the constructed network.

Subsequently, Jiang et al. [122] implemented a semisupervised transfer learning methodology, exploring the user interest similarities between the source and target domains, thereby indicating the transferability of user interests across different domains. Their research contradicts the traditional belief, suggesting that even with a small number of overlapping users, valuable information can be provided for the entire system. By extracting and utilizing information from overlapping user groups, semisupervised transfer learning approaches can be employed to deal with the issue.

However, these studies also highlighted challenges, emphasizing that the effectiveness of the recommendation process heavily relies on the extent of user overlap across different domains. This identified limitation provides a direction for subsequent research, specifically on how to deliver effective recommendations even when the user intersection is minimal. To exploit these latent and obscured relationships, studies have increasingly pivoted towards multidomain recommendation (MDR). Compared to single-target CDR, the focus of MDR primarily lies in handling recommendations across different domains. Although multidomain methods can be applied within the context of CDR, their scopes in practical applications often face certain constraints. To address data sparsity and the other challenges inherent in multidomain recommendation scenarios, Zhang et al. introduced MCF. Successive studies [123–125] integrated more advanced techniques such as feature fusion aiming to further enhance the accuracy of the output recommendations.

In summary, from harnessing both explicit and implicit information within social networks to the transfer of interests across domains and onwards to the exploration of multidomain recommendations, researchers have continuously striven to identify more effective and universal recommendation methods that can cater to user needs across various domains and scenarios.

3.3.2. Dual-Target CDR

Single-target approaches can only utilize auxiliary information from a richer domain to assist a sparser domain. However, the richness of various types of information varies across different domains. If these types of information can be effectively harnessed, it is possible to enhance the recommendation performance in all domains simultaneously, rather than limiting improvements to a single target domain. To address this, dual-target CDR and multi-target CDR have been recently proposed to enhance recommendation performance across dual/multiple domains. To achieve cross-domain recommendations for dual-target domains, researchers have begun to develop various models. These models often incorporate advanced techniques such as graph models and attention mechanisms to facilitate knowledge transfer between different domains.

Research on recommender systems that operate across multiple target domains has been increasingly explored in depth. Zhu et al. [39] were pioneers in that they introduced a dual-target-domain CDR framework called DTCDR. This framework capitalizes on multisource information, ensuring that more detailed embeddings are produced for both users and items. To further integrate this information, the authors employed multitask learning techniques to merge the embeddings of overlapping users.

Building upon this foundation, Liu et al. [126] delved deeper into the embeddings of overlapping users. They created two distinct heterogeneous graphs utilizing ratings and content information from both domains to produce more representative user and item embeddings. Taking hyperparameters and data sparsity into consideration, they achieved more accurate embeddings for these users. Subsequently, the DDTCDR model presented in [40], another dual-target domain CDR framework, took a different approach, emphasizing the bidirectional implicit relationships between users and items. It utilized implicit orthogonal mapping to learn user preferences, allowing for the bidirectional transfer of user embeddings between two domains.

Furthermore, in [127], the authors incorporated graph embedding techniques. They not only utilized connectivity of the user-item graph within a single domain via a novel feature propagation layer but also facilitated bidirectional knowledge transfer between the two domains by employing common users as bridges. Moreover, distinct from prior cross-domain collaborative filtering methods, they integrated both shared user features and domain-specific attributes during the transfer process.

In summary, these studies were dedicated to optimizing the performance of dual-target-domain recommender systems. Their collective efforts have contributed valuably to enhancing the accuracy and robustness of cross-domain recommender systems.

3.3.3. Multi-Target CDR

Multitarget domain CDR is a recommendation to users in either domain for items in both domains. However, its objectives are more intricate, aiming to provide a comprehensive solution to the data sparsity problem. The fundamental concept behind multitarget CDR is to harness additional auxiliary information from multiple domains to enhance recommendation performance. Theoretically, if one can identify and effectively utilize a sufficient number of relevant domains, the longstanding data sparsity issue can be substantially mitigated. Nonetheless, beyond the challenges that are inherent in single- and dual-target-domain CDR scenarios, negative transfer becomes inevitable in practical multitarget-domain CDR contexts. Specifically, with the addition of domains, the recommendation performance of the constructed model might deteriorate, especially in scenarios where the domains are sparse.

The multitarget domain task presents a particularly challenging recommendation scenario, and as of now, no perfect solution tailored to this objective is available. In [128], the authors employed a shared heterogeneous graph to generate richer user and item feature embeddings across domains. The MDCDR [41] utilizes auxiliary data from source domains to improve recommendation quality across multiple domains. However, none of these approaches effectively addresses the negative transfer issue.

As the complexity of multiplatform and multidevice environments intensifies, the demand and significance of cross-domain recommendations are anticipated to escalate further. Advanced models, such as deep learning and graph neural networks, may offer more robust support for cross-domain recommendation tasks. These developments have introduced new challenges such as feature mapping and negative transfer. These emerging research trends inspire us to delve into the challenges within CDR and provide an overview of the ongoing research advancements.

3.4. Social Recommendation

Users' decision-making processes are influenced by their social circles, and numerous researchers have integrated social relations into recommender systems. The main goal is to integrate users' social information as auxiliary data into conventional recommendation frameworks, thereby improving the accuracy of the resulting recommendations. Distinct from other recommendation strategies, social recommendation focuses on leveraging users' social network information to generate more personalized and accurate recommendations. The framework for social recommendation is illustrated in Figure 7.

The primary concept involves aggregating features from both a node itself and its neighboring nodes to generate a consolidated representation. The inputs of this approach are users, items, and the social relationship network among the users. After performing embedding, neural networks are generally employed to handle the relationships between users. The final output consists of item recommendations.

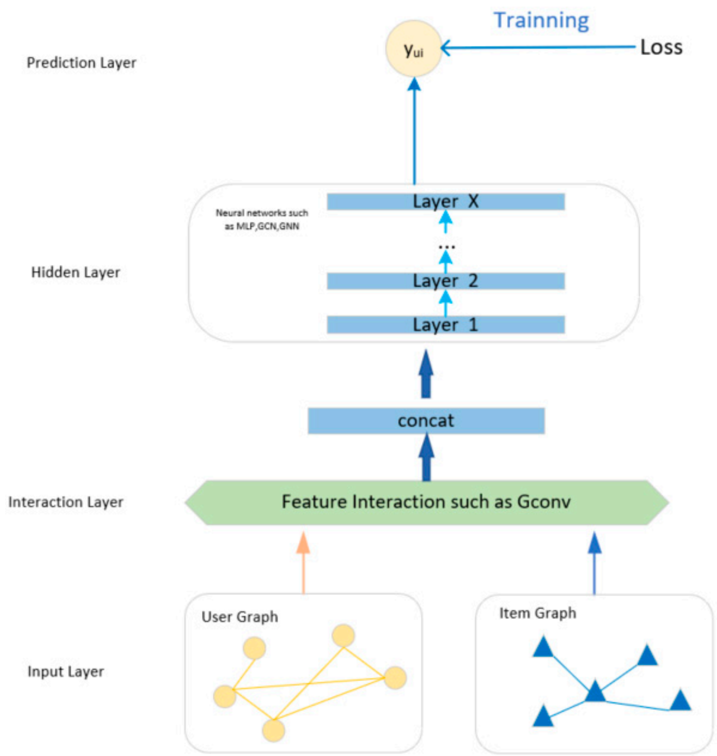

Figure 7. Social recommendation process.

Numerous studies have integrated social relational information into collaborative filtering-based recommendations with the goal of modelling the interrelationships between users. The evolution of social recommendation has spanned three phases. Initially, researchers began to recognize the potential value of trust relationships within a user's social network for recommendation purposes. Leveraging random walks, a trust-based recommendation method was proposed in [129]. The trust relationships among users were employed as weights to adjust the recommendation outcomes. Subsequently, with the application of matrix factorization techniques, models integrating matrix factorization with social networks gained popularity. Building upon trust relationships, these models endeavored to incorporate both strong and weak social ties into a unified framework, allowing recommendations to benefit from both types of relations [130]. With the help of deep learning, the field has progressed into the current era of graph-based social recommendations. These models usually take data from a user's social network as input and subsequently utilize neural networks to forecast users' interests [131].

A graph-based social recommendation has emerged as a focal area of academic interest. The advantages of GNNs in terms of mining graph-structured data have facilitated research on social network recommender systems. GNNs possess the capability to efficiently learn and extract features from graph structures, demonstrating commendable results in user-item representation learning tasks. Consequently, numerous scholars have integrated social networks into recommender system research and employed GNNs to construct recommender system models grounded in graph representation learning. For instance, the GC-MC model [132] and the NGCF model [27] graph convolutions to identify user-item interactions in the original space, yielding enhanced recommendation outcomes in practical applications. Moreover, some researchers have found that adding nonlinear activation functions to GNNs does not significantly improve the accuracy of the resulting recommendations. This observation led to the proposal of a streamlined GNN recommendation model, LightGCN [42], which outperforms NGCF in terms of recommendation performance. LR-GCCF [43] eliminates nonlinear transformations from graph neural networks and substitutes them with linear embedding propagation, thereby reducing unnecessary operations, simplifying the complexity of the model, and still ensuring that satisfactory performance metrics are produced.

Social recommendation algorithms can be broadly classified into three main categories: traditional collaborative filtering methods, deep social recommendation methods relying on graph embeddings, and social recommendation methods based on graph neural networks.

3.4.1. Traditional Collaborative Filtering-Based Social Recommender Systems

In traditional social network-based recommendations, two separate graphs need to be constructed: one representing user-item interactions, and another representing user-user social connections. Therefore, social recommendation algorithms utilizing graph neural networks require knowledge extraction from both graphs simultaneously for accurate inference. They can leverage coarse-grained and sparse user trust relationships to enhance conventional collaborative filtering methods. Essentially, this approach involves the use of similarity metrics to identify users similar to the target user. Subsequently, different weights are assigned to ratings for the target item based on this similarity, facilitating the computation of the user's rating for that item. The method for determining user-to-user similarity is crucial to this concept. Consequently, existing social recommendation algorithms have explored various designs for graph neural network encoders and adopted different architectures based on their objectives. Upon determining the similarity between two entities, the set of users most similar to the target user can be identified based on their ratings. This, in turn, allows for the recommendation of items that the user may find interesting and ranks them accordingly.

3.4.2. Deep Social Recommendation Based on Graph Embedding

Traditional social recommender systems only integrate explicit trust relationships between users and ignore implicit trust relationships. Graph embedding models represent all entities as nodes and their relationships as edges in a graph. These models transform nodes into vector representations, allowing them to retain their approximate relationships from the graph structure in a Euclidean space. Afterwards, vector distances are computed to predict the scores of candidate items for the target user.

Using the application of graph embedding techniques, it becomes possible to acquire low-dimensional feature representations of users within a social network. These low-dimensional representations enable the inference of detailed trust relationships between users. Consequently, by incorporating the weighted ratings from trusted and similar users for a specific item, predictions regarding the user's rating for that item can be generated. Figure 8 illustrates the framework of a recommendation algorithm employing graph embedding models.

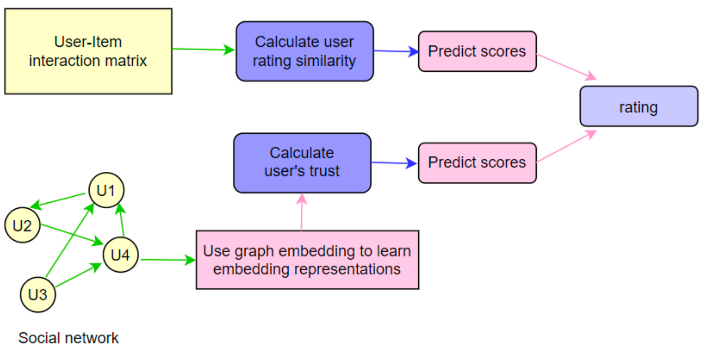

Figure 8. Recommendation framework based on graph embedding.

Prominent graph embedding techniques, such as Deep Walk [133], graph factorization [44], and LINE [134], possess distinct characteristics and limitations. For instance, graph factorization [45] is suited for performing embedding learning in large-scale networks but fails to preserve the network's global structure and is solely applicable to undirected information networks. In contrast, while Deep Walk reveals certain attributes of an information network using a random walk algorithm, it lacks a comprehensive depiction of the preserved network properties and overlooks the connections between feature vectors. In contrast, LINE effectively maintains both local and global structural aspects of information networks and has found widespread application in recommender systems. Additionally, the prowess of graph embedding techniques in terms of extracting the semantic relationships within structures is evident in applications such as the work performed by Xie et al. [135] on POI recommendations and the research conducted by Yin et al. [136] on low-dimensional vector representations for users and items. Graph embedding representation learning, with its powerful representational capabilities, low memory consumption, and efficient computations, has been widely researched and applied for semantic relationship extraction in network structures [137].

Overall, given the advantages of graph embedding techniques in modelling and optimization tasks, they present broad prospects for various applications and merit further exploration.

3.4.3. Social Recommendation Based on GNN

Over time, neural networks' applications in the recommendation field have significantly broadened. Graph neural networks (GNNs) represent innovative extensions of neural networks, drawing inspiration from convolutional neural networks (CNNs) and the concept of graph embeddings. They are adept at extracting and representing features in graph-based data. Characterized by their efficiency and scalability, GNNs have demonstrated profound capabilities when handling graph data. Users and items are considered as nodes in the graph, and interactions are represented as edges. This process leads to the transformation of user-item interactions into embedded representations within the graph. By integrating these social relationships into recommender systems and subsequently feeding them into neural networks, more comprehensive and personalized recommendations can be provided.

The goal of a GNN is to acquire a state embedding that captures the neighborhood information of each node. The embedding is a z-dimensional real-valued vector associated with node u that can be used to generate a node label output, representing the vector output of node u at a given network layer. Local transfer functions are shared across all nodes and update node state based on input neighborhood. Finally, the output is produced. Then, based on the characteristics of nodes, the characteristics of adjacent edges, and the state embeddings and features of adjacent fixed points, the social relationship model is further improved, for example, by considering the impact of various social relationships on user behaviors. Attention mechanisms, among other techniques, have been introduced [138]. Broadly speaking, the architecture of a social recommendation model comprises three key components: an encoder, a decoder, and a loss function. The encoder [139] utilizes different graph neural network encoders to represent users and items as low-dimensional vectors (i.e., embeddings). The decoder then predicts each user's preference for each item by performing different operations on the obtained user and item embeddings. Optimization is finally achieved using various loss functions [140]. The performance of social recommendation methods based on GNN largely depends on their encoders.

Beyond user behaviors, social recommender systems also take various types of information into account, including social interactions and shared interests. They operate on the premise that users are likely to share similar interests with friends or contacts within their social networks, utilizing these social relationships to enhance the accuracy and relevance of recommendations. Additionally, the strength and depth levels of social ties are considered to enhance the produced recommendation outcomes. For new users, the cold-start problem can be mitigated via their social networks. Furthermore, recommendations stemming from friends are more likely to be trusted and accepted by users. However, social recommendations exhibit some limitations. Utilizing users' social network data may raise privacy and security concerns [141]. Moreover, not all social interactions correlate with genuine preferences, necessitating the filtering of irrelevant or misleading information.

Graph neural networks offer robust mechanisms for recommender systems to capture the intricate interactions and relationships between users and items [142], with several models demonstrating significant influences and innovations. Initially, GraphSAGE [28], a pioneering spatial GNN model, marked an essential milestone. It moved away from relying on the entirety of graph information and instead strategically sampled the neighbors of target nodes and combined their embeddings. This approach achieved efficient target embedding updates and offered a viable tool for providing social recommendations in large-scale networks. Following GraphSAGE, the GAT [143] further refined the spatial GNN concept, addressing the limitations of the previous spectral methods regarding key issues such as model generalization. Notably, the GAT introduced an attention mechanism, enabling the model to allocate different weights based on the importance of neighboring nodes, leading to a more selective neighborhood feature aggregation process. This strategy enhanced the model's discriminatory capacity, achieving commendable results in social recommendation tasks. For dealing with heterogeneous graph structures, HetGNN [144] emerged as a representative technique. Distinct from traditional GNN models, HetGNN devises specific aggregation strategies for nodes and edges that are present in heterogeneous graphs. Initially, it segregates neighbors based on their types and then employs LSTM and MEAN operations to individually process these subsets, effectively capturing the abundant structural and attribute information inherent in heterogeneous graphs. Furthermore, it is noteworthy that HetGNN implements spectral GNN methods on hypergraph structures, propelling recommender system research in a novel direction. Regarding the exploration of social network structures, DiffNet and DiffNet++ stand as pivotal milestones. Both methods emphasize delving into higher-order social structures to achieve enhanced recommendation accuracy and efficacy. DiffNet accentuates capturing higher-order neighbor information within the network to boost its recommendation results [145], while DiffNet++ further refines and extends this base [46]. Following DiffNet and DiffNet++, GraphRec emerged as an innovative approach, emphasizing not only capturing interactions between users and items but also jointly capturing opinions about both users and items [146]. This dual capture strategy is designed to ensure recommendation accuracy while maintaining system interpretability.

With the continuous evolution of social media and platforms, the significance of social recommender systems is set to expand further [147]. Advanced technologies such as deep learning and graph neural networks have notably enhanced the accuracy and efficiency of social recommendation [148]. Moreover, striking a balance between user privacy and the quality of recommendations will emerge as a pivotal research direction [149]. In the future, we may also witness an increase in cross-platform social recommendation solutions, ensuring that users receive consistent and high-quality recommendation experiences across diverse platforms.

4. Challenges and Developments

Recommender systems are designed to help users discover items that align with their preferences from a wide range of potential recommendations. Deep learning can aid researchers in improving the effectiveness of these recommendations. This paper analyzes four distinct types of recommender systems, including content-based recommendation, sequential recommendation, cross-domain recommendation, and social recommender systems. Although the integration of these recommendation techniques with deep learning has achieved satisfactory results, challenges remain [150]. Future research efforts can be pursued in the following areas.

(1) Security enhancements are needed

With the growth of networking sites, accurately recommending items of interest to users has become one of the key strategies employed by various websites to attract users. Only by digging deep into multidimensional user data can recommendations be identified that truly align with user preferences [151]. In reality, while users expect recommender systems to suggest items of interest, they do not want their other private details to be disclosed [152]. Current research primarily involves distorting and obfuscating user data to ensure privacy [153]. Although this data perturbation strategy does protect the user's personal information, it can lead to the inaccurate extraction of user data, significantly compromising the resulting recommendation accuracy. Therefore, future research could focus on a methodology that not only safeguards user privacy but also enhances recommendation precision.

(2) Methods for extracting user preference features are lacking

At present, recommendation subjects rely heavily on users' ratings or feedback on recommended items. Current research lacks adequate modeling methods that can multi-dimensionally extract user features and recommended items [154], as well as their linear and non-linear relationships [155]. Although neural networks can address this issue to some extent, there remains significant room for improvement. Consequently, forthcoming research should incorporate a more diverse range of methods to extract the features of both users and recommended entities.

(3) Evaluation metrics are singular

When evaluating the performance of recommender systems, the existing studies primarily emphasize the accuracy of results and associated precision rate, considering accuracy as the key metric for determining the effectiveness of recommender systems. However, when users interact with these applications in real-world scenarios, they not only expect the system to provide accurate recommendations for items of interest but also seek a broader and more innovative range of suggestions [156]. Hence, the novelty and diversity of the output recommendations should be considered in the future.

In summary, in addition to the aforementioned aspects, there are several directions for future research. There may be an increased focus on the interpretability of deep learning models. This is because, in practical applications, there is a growing demand from users and regulatory bodies for greater transparency and comprehensibility in recommender system decisions. With the proliferation of multimodal data (text, images, audio, etc.), future recommender systems may integrate these data sources more extensively to provide a richer and more diverse recommendation experience. The application of reinforcement learning in recommender systems may become more widespread, optimizing recommendation strategies via user interactions for higher long-term returns.

Future research may explore how to transfer deep learning models from one domain to another to enhance the generalizability of recommender systems. Federated learning, while protecting user privacy, allows different institutions and platforms to collaboratively train recommendation models and may become an important direction in future recommender system research.

Despite improvements in handling sparse data with deep learning methods, data sparsity and cold-start problems remain challenges. Specifically, providing accurate recommendations for new users and items remains a challenge. Privacy and fairness have long been key challenges in the field of recommender systems. Balancing personalized recommendations with user privacy and ensuring that recommendation system decisions are fair will be a focus of future research. Deep learning models often require substantial computational resources. Building scalable and efficient deep learning recommender systems in practical applications remains a challenge. Deep learning methods typically require a large amount of training data, so encouraging user participation in feedback collection to improve the performance of recommender systems remains a question.

5. Conclusions

As technologies like deep learning, data mining, and predictive algorithms mature, future research will focus on enhancing the accuracy, security, and privacy of recommender systems. This article explores both traditional recommendation methodologies and those incorporating various deep learning models. We compare the differences between conventional recommendation models and deep learning-based approaches, summarizing the prevalent challenges in recommender systems. We also offer insights into future research directions for recommender systems, benefiting researchers who are interested in the fields of recommender systems or deep learning.

Author Contributions: Conceptualization, H.Z.; methodology, H.Z.; validation, H.Z.; formal analysis, H.Z., F.X. and H.C.; investigation, H.Z.; resources, H.Z.; data curation, H.Z.; writing—original draft preparation, H.Z.; writing—review and editing, H.Z., F.X. and H.C.; visualization, H.Z., F.X. and H.C.; supervision, F.X.; project administration, F.X. and H.C.; funding acquisition, H.Z., F.X. and H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities (No. 2022JBMC005), the National Natural Science Foundation of China under Grant 61872033 and Grant 72004009, the National Key R&D Program of China under Grant 2018YFC0832304, and the Beijing Nova Program from Beijing Municipal Science & Technology Commission under Grant Z201100006820015.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare that they have no conflict of interest or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Zhu, Y.; Lin, Q.; Lu, H. Recommending learning objects through attentive heterogeneous graph convolution and operation-aware neural network. *IEEE Trans. Knowl. Data Eng.* **2021**, *35*, 4178–4189. [CrossRef]
2. Leiva, M.; Budán, M.C.D.; Simari, G.I. Guidelines for the analysis and design of argumentation-based recommendation systems. *IEEE Intell. Syst.* **2020**, *35*, 28–37. [CrossRef]
3. Goldberg, K.; Roeder, T.; Gupta, D.; Perkins, C. Eigentaste: A constant time collaborative filtering algorithm. *Inf. Retr.* **2001**, *4*, 133–151. [CrossRef]
4. Abiodun, O.I.; Jantan, A.; Omolara, A.E.; Dada, K.V.; Umar, A.M.; Linus, O.U.; Arshad, H.; Kazaure, A.A.; Gana, U.; Kiru, M.U. Comprehensive review of artificial neural network applications to pattern recognition. *IEEE Access* **2019**, *7*, 158820–158846. [CrossRef]
5. Gheisari, M.; Wang, G.; Bhuiyan, M.Z.A. A survey on deep learning in big data. In Proceedings of the 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, China, 21–24 July 2017; IEEE: Piscataway, NJ, USA, 2017; Volume 2, pp. 173–180.
6. Covington, P.; Adams, J.; Sargin, E. Deep neural networks for youtube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 19 September 2016; pp. 191–198.
7. Sun, K.; Wang, L.; Xu, B.; Zhao, W.; Teng, S.W.; Xia, F. Network representation learning: From traditional feature learning to deep learning. *IEEE Access* **2020**, *8*, 205600–205617. [CrossRef]
8. Han, S.; Qiao, Y.; Zhang, Y.; Lin, W.; Yang, J. Analyze users' online shopping behavior using interconnected online interest-product network. In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–6.
9. Zhang, Y.; Qian, Y.; Gan, M.; Tang, X.; Lin, Z. Service Recommendation Based on User Dynamic Preference Extraction and Prediction. In Proceedings of the 2019 IEEE World Congress on Services (SERVICES), Milan, Italy, 8–13 July 2019; IEEE: Piscataway, NJ, USA, 2019; Volume 2642, pp. 121–126.
10. Skarding, J.; Gabrys, B.; Musial, K. Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey. *IEEE Access* **2021**, *9*, 79143–79168. [CrossRef]
11. Ouyang, Y.; Guo, B.; Wang, Q.; Yu, Z. Cross-domain recommendation with cross-graph knowledge transfer network. In Proceedings of the ICC 2021-IEEE International Conference on Communications, Montreal, QC, Canada, 14–23 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6.
12. Jiang, M.; Cui, P.; Wang, F.; Zhu, W.; Yang, S. Scalable recommendation with social contextual information. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 2789–2802. [CrossRef]
13. Guo, G. Resolving data sparsity and cold start in recommender systems. In Proceedings of the User Modeling, Adaptation, and Personalization: 20th International Conference, UMAP 2012, Montreal, QC, Canada, 16–20 July 2012; Proceedings 20. Springer: Berlin/Heidelberg, Germany, 2012; pp. 361–364.
14. Tang, J.; Hu, X.; Liu, H. Social recommendation: A review. *Soc. Netw. Anal. Min.* **2013**, *3*, 1113–1133. [CrossRef]

15. Wang, H.; Kou, G.; Peng, Y. An iterative algorithm to derive priority from large-scale sparse pairwise comparison matrix. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *52*, 3038–3051. [CrossRef]
16. Zhang, S.; Yao, L.; Sun, A.; Tay, Y. Deep learning based recommender system: A survey and new perspectives. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 1–38. [CrossRef]
17. Yang, Z.; Ding, M.; Zou, X.; Tang, J.; Xu, B.; Zhou, C.; Yang, H. Region or global a principle for negative sampling in graph-based recommendation. *IEEE Trans. Knowl. Data Eng.* **2022**, *35*, 6264–6277. [CrossRef]
18. Li, H.; Zhang, H.; Wang, S.; Hassan, A.E. Studying the Practices of Logging Exception Stack Traces in Open-Source Software Projects. *IEEE Trans. Softw. Eng.* **2021**, *48*, 4907–4924. [CrossRef]
19. Gantner, Z.; Drumond, L.; Freudenthaler, C.; Rendle, S.; Schmidt-Thieme, L. Learning attribute-to-feature mappings for cold-start recommendations. In Proceedings of the 2010 IEEE International Conference on Data Mining, Sydney, NSW, Australia, 13–17 December 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 176–185.
20. Ma, H.; Yang, H.; Lyu, M.R.; King, I. Sorec: Social recommendation using probabilistic matrix factorization. In Proceedings of the 17th ACM Conference on Information and Knowledge Man-Agement, Napa Valley, CA, USA, 26–30 October 2008; pp. 931–940.
21. Mnih, A.; Salakhutdinov, R.R. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*; 2007; Volume 20, Available online: https://proceedings.neurips.cc/paper/2007 (accessed on 18 September 2023).
22. Jamali, M.; Ester, M. A matrix factorization technique with trust propagation for recommendation in social networks. In Proceedings of the Fourth ACM Conference on Recommender Systems, Barcelona, Spain, 26–30 September 2010; pp. 135–142.
23. Ma, H.; Zhou, D.; Liu, C.; Lyu, M.R.; King, I. Recommender systems with social regularization. In Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, Hong Kong, China, 9–12 February 2011; pp. 287–296.
24. Sedhain, S.; Menon, A.K.; Sanner, S.; Xie, L. Autorec: Autoencoders meet collaborative filtering. In Proceedings of the 24th Intermnational Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 111–112.
25. Devooght, R.; Bersini, H. Long and short-term recommendations with recurrent neural networks. In Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, Bratislava, Slovakia, 9–12 July 2017; pp. 13–21.
26. Wu, S.; Sun, F.; Zhang, W.; Xie, X.; Cui, B. Graph neural networks in recommender systems: A survey. *arXiv* **2020**, arXiv:2011.02260. [CrossRef]
27. Wang, X.; He, X.; Wang, M.; Feng, F.; Chua, T.S. Neural graph collaborative filtering. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 165–174.
28. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. *arXiv* **2017**, arXiv:1706.02216.
29. Ying, R.; He, R.; Chen, K.; Eksombatchai, P.; Hamilton, W.L.; Leskovec, J. Graph convolutional neural networks for web-scale recommender systems. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 974–983.
30. He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; Chua, T.S. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 173–182.
31. Guo, H.; Tang, R.; Ye, Y.; Li, Z.; He, X. DeepFM: A factorization-machine based neural network for CTR prediction. *arXiv* **2017**, arXiv:1703.04247.
32. Wu, Y.; Su, L.; Wu, L.; Xiong, W. FedDeepFM: A Factorization Machine-Based Neural Network for Recommendation in Federated Learning. *IEEE Access* **2023**, *11*, 74182–74190. [CrossRef]
33. Rendle, S.; Freudenthaler, C.; Schmidt-Thieme, L. Factorizing personalized markov chains for next-basket recommendation. In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC, USA, 26–30 April 2010; pp. 811–820.
34. Kang, W.C.; McAuley, J. Self-attentive sequential recommendation. In Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM), Singapore, 17–20 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 197–206.
35. Wang, K.; Wang, X.; Lu, X. POI recommendation method using LSTM-attention in LBSN considering privacy protection. *Complex Intell. Syst.* **2023**, *9*, 2801–2812. [CrossRef]
36. Lv, F.; Jin, T.; Yu, C.; Sun, F.; Lin, Q.; Yang, K.; Ng, W. SDM: Sequential Deep Matching Model for Online Large-scale Recommender System. In Proceedings of the CIKM '19: The 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019.
37. Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; Tikk, D. Session-based recommendations with recurrent neural networks. *arXiv* **2015**, arXiv:1511.06939.
38. Natarajan, S.; Vairavasundaram, S.; Kotecha, K.; Indragandhi, V.; Palani, S.; Saini, J.R.; Ravi, L. CD-SemMF: Cross-Domain Semantic Relatedness Based Matrix Factorization Model Enabled With Linked Open Data for User Cold Start Issue. *IEEE Access* **2022**, *10*, 52955–52970. [CrossRef]
39. Zhu, F.; Chen, C.; Wang, Y.; Liu, G.; Zheng, X. Dtcdr: A framework for dual-target cross-domain recommendation. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 1533–1542.
40. Li, P.; Tuzhilin, A. Ddtcdr: Deep dual transfer cross domain recommendation. In Proceedings of the 13th International Conference on Web Search and Data Mining, Houston, TX, USA, 3–7 February 2020; pp. 331–339.
41. Krishnan, A.; Das, M.; Bendre, M.; Yang, H.; Sundaram, H. Transfer learning via contextual invariants for one-to-many cross-domain recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, China, 25–30 July 2020; pp. 1081–1090.

42. He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; Wang, M. Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, China, 25–30 July 2020; pp. 639–648.
43. Chen, L.; Wu, L.; Hong, R.; Zhang, K.; Wang, M. Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 27–34. [CrossRef]
44. Ahmed, A.; Shervashidze, N.; Narayanamurthy, S. Distributed large-scale natural graph factorization. In Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 13–17 May 2013; pp. 37–48.
45. Li, Z.; Wu, S.; Cui, Z.; Zhang, X. GraphFM: Graph factorization machines for feature interaction modeling. *arXiv* **2021**, arXiv:2105.11866.
46. Wu, L.; Li, J.; Sun, P.; Hong, R.; Ge, Y.; Wang, M. Diffnet++: A neural influence and interest diffusion network for social recommendation. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 4753–4766. [CrossRef]
47. Goldberg, D.; Nichols, D.; Oki, B.M.; Terry, D. Using collaborative filtering to weave an information tapestry. *Commun. ACM* **1992**, *35*, 61–70. [CrossRef]
48. Wu, C.Y.; Ahmed, A.; Beutel, A.; Smola, A.J.; Jing, H. Recurrent recommender networks. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, Cambridge, UK, 6–10 February 2017; pp. 495–503.
49. Kamishima, T.; Akaho, S.; Asoh, H.; Sakuma, J. Efficiency Improvement of Neutrality-Enhanced Recommendation. In Proceedings of the Decisions@ RecSys, Hong Kong, China, 12–16 October 2013; pp. 1–8.
50. Rendle, S. Factorization machines. In Proceedings of the 2010 IEEE International Conference on Data Mining, Sydney, NSW, Australia, 13–17 December 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 995–1000.
51. Zhou, G.; Zhu, X.; Song, C.; Fan, Y.; Zhu, H.; Ma, X.; Yan, Y.; Jin, J.; Li, H.; Gai, K. Deep interest network for click-through rate prediction. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 1059–1068.
52. Zhou, G.; Mou, N.; Fan, Y.; Pi, Q.; Bian, W.; Zhou, C.; Zhu, X.; Gai, K. Deep interest evolution network for click-through rate prediction. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 5941–5948. [CrossRef]
53. Li, C.; Liu, Z.; Wu, M.; Xu, Y.; Zhao, H.; Huang, P.; Kang, G.; Chen, Q.; Li, W.; Lee, D.L. Multi-interest network with dynamic routing for recommendation at Tmall. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 2615–2623.
54. Marcuzzo, M.; Zangari, A.; Albarelli, A.; Gasparetto, A. Recommendation systems: An insight into current development and future research challenges. *IEEE Access* **2022**, *10*, 86578–86623. [CrossRef]
55. De Gemmis, M.; Lops, P.; Semeraro, G.; Basile, P. Integrating tags in a semantic content-based recommender. In Proceedings of the 2008 ACM Conference on Recommender Systems, Lausanne, Switzerland, 23–25 October 2008; pp. 163–170.
56. Mooney, R.J.; Roy, L. Content-based book recommending using learning for text categorization. In Proceedings of the Fifth ACM Conference on Digital Libraries, San Antonio, TX, USA, 2–7 June 2000; pp. 195–204.
57. Waila, P.; Singh, V.; Singh, M. A Scientometric Analysis of Research in Recommender Systems. *J. Scientometr. Res.* **2016**, *4*, 71–84. [CrossRef]
58. Wang, H.; Czerminski, R.; Jamieson, A.C. Neural networks and deep learning. In *The Machine Age of Customer Insight*; Emerald Publishing Limited: Bingley, UK, 2021; pp. 91–101.
59. Lu, Y.T.; Yu, S.I.; Chang, T.C.; Hsu, J.Y.J. A content-based method to enhance tag recommendation. In Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence, Pasadena, CA, USA, 11–17 July 2009.
60. Peng, J.; Zeng, D.D.; Zhao, H.; Wang, F.Y. Collaborative filtering in social tagging systems based on joint item-tag recommendations. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, Toronto, ON, Canada, 26–30 October 2010; pp. 809–818.
61. Vasile, F.; Smirnova, E.; Conneau, A. Meta-prod2vec: Product embeddings using side-information for recommendation. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; pp. 225–232.
62. Wu, L.; He, X.; Wang, X.; Zhang, K.; Wang, M. A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation. *IEEE Trans. Knowl. Data Eng.* **2022**, *35*, 4425–4445. [CrossRef]
63. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
64. Xu, M.; Liu, F.; Xu, W. A survey on sequential recommendation. In Proceedings of the 2019 6th International Conference on Information Science and Control Engineering (ICISCE), Shanghai, China, 20–22 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 106–111.
65. Yang, Y.; Xiang, Y.; Xiong, L. Collaborative filtering and recommendation algorithm based on matrix factorization and user nearest neighbor model. *J. Comput. Appl.* **2012**, *32*, 395. [CrossRef]
66. Wang, J.F.; Liu, R.D.; Liu, Y. Non-negative matrix factorization algorithm with bias in recommender system. *J. Chin. Comput. Syst.* **2018**, *39*, 69–73.
67. Yoon, J.H.; Jang, B. Evolution of Deep Learning-Based Sequential Recommender Systems: From Current Trends to New Perspectives. *IEEE Access* **2023**, *11*, 54265–54279. [CrossRef]
68. Park, K.; Lee, J.; Choi, J. Deep neural networks for news recommendations. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 2255–2258.

69. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian personalized ranking from implicit feedback. *arXiv* **2012**, arXiv:1205.2618.
70. Kurgan, L.A.; Musilek, P. A survey of knowledge discovery and data mining process models. *Knowl. Eng. Rev.* **2006**, *21*, 1–24. [CrossRef]
71. Villatel, K.; Smirnova, E.; Mary, J.; Preux, P. Recurrent neural networks for long and short-term sequential recommendation. *arXiv* **2018**, arXiv:1807.09142.
72. Thaipisutikul, T.; Shih, T.K.; Enkhbat, A.; Aditya, W. Exploiting long-and short-term preferences for deep context-aware recommendations. *IEEE Trans. Comput. Soc. Syst.* **2021**, *9*, 1237–1248. [CrossRef]
73. Fattah, S.M.M.; Bouguettaya, A.; Mistry, S. Long-term IaaS selection using performance discovery. *IEEE Trans. Serv. Comput.* **2020**, *15*, 2129–2143. [CrossRef]
74. Zheng, C.; Tao, D.; Wang, J.; Cui, L.; Ruan, W.; Yu, S. Memory augmented hierarchical attention network for next point-of-interest recommendation. *IEEE Trans. Comput. Soc. Syst.* **2020**, *8*, 489–499. [CrossRef]
75. Ying, H.; Zhuang, F.; Zhang, F.; Liu, Y.; Xu, G.; Xie, X.; Xiong, H.; Wu, J. Sequential recommender system based on hierarchical attention network. In Proceedings of the IJCAI International Joint Conference on Artificial Intelligence, Vienna, Austria, 23–29 July 2018.
76. Gers, F.A.; Schmidhuber, J.; Cummins, F. Learning to forget: Continual prediction with LSTM. *Neural Comput.* **2000**, *12*, 2451–2471. [CrossRef]
77. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.
78. Li, J.; Ren, P.; Chen, Z.; Ren, Z.; Lian, T.; Ma, J. Neural attentive session-based recommendation. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 1419–1428.
79. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; 2017; Volume 30, Available online: https://proceedings.neurips.cc/paper/2017 (accessed on 18 September 2023).
80. Zhang, S.; Tay, Y.; Yao, L.; Sun, A.; An, J. Next item recommendation with self-attentive metric learning. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 9.
81. Cho, K.; Van Merrienboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
82. Tsai, Y.H.H.; Bai, S.; Liang, P.P.; Kolter, J.Z.; Morency, L.P.; Salakhutdinov, R. Multimodal transformer for unaligned multimodal language sequences. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; Volume 2019, p. 6558.
83. Azhir, E.; Hosseinzadeh, M.; Khan, F.; Mosavi, A. Performance Evaluation of Query Plan Recommendation with Apache Hadoop and Apache Spark. *Mathematics* **2022**, *10*, 3517. [CrossRef]
84. Singer, U.; Roitman, H.; Eshel, Y.; Nus, A.; Guy, I.; Levi, O.; Hasson, I.; Kiperwasser, E. Sequential modeling with multiple attributes for watchlist recommendation in e-commerce. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, Tempe, AZ, USA, 21–25 February 2022; pp. 937–946.
85. Tuan, T.X.; Phuong, T.M. 3D convolutional networks for session-based recommendation with content features. In Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, 27–31 August 2017; pp. 138–146.
86. Yan, A.; Cheng, S.; Kang, W.C.; Wan, M.; McAuley, J. CosRec: 2D convolutional neural networks for sequential recommendation. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; ACM: New York, NY, USA, 2019; pp. 2173–2176.
87. Tang, J.X.; Wang, K. Personalized top-n sequential recommendation via convolutional sequence embedding. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. Association for Computing Machinery, Marina Del Rey, CA, USA, 5–9 February 2018; pp. 565–573.
88. Tan, Y.K.; Xu, X.; Liu, Y. Improved recurrent neural networks for session-based recommendations. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, Boston, MA, USA, 15 September 2016; pp. 17–22.
89. Bogina, V.; Kuflik, T. Incorporating Dwell Time in Session-Based Recommendations with Recurrent Neural Networks. In Proceedings of the RecTemp@ RecSys, Como, Italy, 27–31 August 2017; pp. 57–59.
90. Hidasi, B.; Karatzoglou, A. Recurrent neural networks with top-k gains for session-based recommendations. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 843–852.
91. Hidasi, B.; Quadrana, M.; Karatzoglou, A.; Tikk, D. Parallel recurrent neural network architectures for feature-rich session-based recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; pp. 241–248.
92. Jannach, D.; Ludewig, M. When recurrent neural networks meet the neighborhood for session-based recommendation. In Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, 27–31 August1 2017; pp. 306–310.
93. Chen, X.; Xu, H.; Zhang, Y.; Tang, J.; Cao, Y.; Qin, Z.; Zha, H. Sequential recommendation with user memory networks. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, Marina Del Rey, CA, USA, 5–9 February 2018; pp. 108–116.

94. Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; Jiang, P. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 1441–1450.
95. Zhou, W.; Liu, Y.; Li, M.; Wang, Y.; Shen, Z.; Feng, L.; Zhu, Z. Dynamic Multi-Objective Optimization Framework with Interactive Evolution for Sequential Recommendation. *IEEE Trans. Emerg. Top. Comput. Intell.* **2023**, *7*, 1228–1241. [CrossRef]
96. Cen, Y.; Zhang, J.; Zou, X.; Zhou, C.; Yang, H.; Tang, J. Controllable multi-interest framework for recommendation. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 2942–2951.
97. Meng, W.; Yang, D.; Xiao, Y. Incorporating user micro-behaviors and item knowledge into multi-task learning for session-based recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, China, 25–30 July 2020; pp. 1091–1100.
98. Yin, D.; Feng, S. Enhanced Attention Framework for Multi-Interest Sequential Recommendation. *IEEE Access* **2022**, *10*, 67703–67712. [CrossRef]
99. Huang, X.; Qian, S.; Fang, Q.; Sang, J.; Xu, C. Csan: Contextual self-attention network for user sequential recommendation. In Proceedings of the 26th ACM International Conference on Multimedia, Seoul, Republic of Korea, 22–26 October 2018; pp. 447–455.
100. Hoppe, A.; Nicolle, C.; Roxin, A. Automatic ontology-based user profile learning from heterogeneous web resources in a big data context. *Proc. VLDB Endow.* **2013**, *6*, 1428–1433. [CrossRef]
101. Gao, C.; He, X.; Gan, D.; Chen, X.; Feng, F.; Li, Y.; Chua, T.-S.; Yao, L.; Song, Y.; Jin, D. Learning to recommend with multiple cascading behaviors. *IEEE Trans. Knowl. Data Eng.* **2019**, *33*, 2588–2601. [CrossRef]
102. Le, D.T.; Lauw, H.W.; Fang, Y. Modeling contemporaneous basket sequences with twin networks for next-item recommendation. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18), Stockholm, Sweden, 13–19 July 2018.
103. Li, Z.; Zhao, H.; Liu, Q.; Huang, Z.; Mei, T.; Chen, E. Learning from history and present: Next-item recommendation via discriminatively exploiting user behaviors. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 1734–1743.
104. Zhou, C.; Bai, J.; Song, J.; Liu, X.; Zhao, Z.; Chen, X.; Gao, J. Atrank: An attention-based user behavior modeling framework for recommendation. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
105. Fan, Z.; Liu, Z.; Zhang, J.; Xiong, Y.; Zheng, L.; Yu, P.S. Continuous-time sequential recommendation with temporal graph collaborative transformer. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Gold Coast, QLD, Australia, 1–5 November 2021; pp. 433–442.
106. Liu, Z.; Chen, Y.; Li, J.; Yu, P.S.; McAuley, J.; Xiong, C. Contrastive self-supervised sequential recommendation with robust augmentation. *arXiv* **2021**, arXiv:2108.06479.
107. Memmel, C. What drives the short-term fluctuations of banks' exposure to interest rate risk? *Rev. Financ. Econ.* **2020**, *38*, 674–686. [CrossRef]
108. Luo, H.; Yang, N.; Philip, S.Y. Hybrid deep embedding for recommendations with dynamic aspect-level explanations. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 870–879.
109. Choe, B.; Kang, T.; Jung, K. Recommendation system with hierarchical recurrent neural network for long-term time series. *IEEE Access* **2021**, *9*, 72033–72039. [CrossRef]
110. Wang, S.; Cao, L.; Wang, Y.; Sheng, Q.Z.; Orgun, M.A.; Lian, D. A survey on session-based recommender systems. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–38. [CrossRef]
111. Elkahky, A.M.; Song, Y.; He, X. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 278–288.
112. Man, T.; Shen, H.; Jin, X.; Cheng, X. Cross-domain recommendation: An embedding and mapping approach. *Int. Jt. Conf. Artif. Intell.* **2017**, *17*, 2464–2470.
113. Wang, C.-D.; Chen, Y.-H.; Xi, W.-D.; Huang, L.; Xie, G. Cross-Domain Explicit–Implicit-Mixed Collaborative Filtering Neural Network. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *52*, 6983–6997. [CrossRef]
114. Li, B. Cross-Domain Collaborative Filtering: A Brief Survey. In Proceedings of the 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, Boca Raton, FL, USA, 7–9 November 2111; IEEE: Piscataway, NJ, USA, 2011; pp. 1085–1086.
115. Gupta, A.; Budania, H.; Singh, P.; Singh, P.K. Facebook based choice filtering. In Proceedings of the 2017 IEEE 7th International Advance Computing Conference (IACC), Hyderabad, India, 5–7 January 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 875–879.
116. Chen, H.; Wang, X.; Xie, R.; Zhou, Y.; Zhu, W. Cross-domain Recommendation with Behavioral Importance Perception. *Proc. ACM Web Conf.* **2023**, *2023*, 1294–1304.
117. Zhu, Y.; Tang, Z.; Liu, Y.; Zhuang, F.; Xie, R.; Zhang, X.; Lin, L.; He, Q. Personalized transfer of user preferences for cross-domain recommendation. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, Tempe, AZ, USA, 21–25 February 2022; pp. 1507–1515.

118. Vodungbo, B.; Gautier, J.; Lambert, G.; Sardinha, A.B.; Lozano, M.; Sebban, S.; Ducousso, M.; Boutu, W.; Li, K.; Tudu, B.; et al. Laser-induced ultrafast demagnetization in the presence of a nanoscale magnetic domain network. *Nat. Commun.* **2012**, *3*, 999. [CrossRef]
119. Shapira, B.; Rokach, L.; Freilikhman, S. Facebook single and cross domain data for recommendation systems. *User Model. User-Adapt. Interact.* **2013**, *23*, 211–247. [CrossRef]
120. Zhou, K.; Wang, H.; Zhao, W.X.; Zhu, Y.; Wang, S.; Zhang, F.; Wang, Z.; Wen, J.R. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual Event, Ireland, 19–23 October 2020; pp. 1893–1902.
121. Tiroshi, A.; Kuflik, T. Domain ranking for cross domain collaborative filtering. In Proceedings of the User Modeling, Adaptation, and Personalization: 20th International Conference, UMAP 2012, Montreal, QC, Canada, 16–20 July 2012; Proceedings 20. Springer: Berlin/Heidelberg, Germany, 2012; pp. 328–333.
122. Jiang, M.; Cui, P.; Yuan, N.J.; Xie, X.; Yang, S. Little is much: Bridging cross-platform behaviors through overlapped crowds. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30.
123. Moreno, O.; Shapira, B.; Rokach, L.; Shani, G. Talmud: Transfer learning for multiple domains. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, Maui, HI, USA, 29 October–2 November 2012; pp. 425–434.
124. Pan, W.; Yang, Q. Transfer learning in heterogeneous collaborative filtering domains. *Artif. Intell.* **2013**, *197*, 39–55. [CrossRef]
125. Zhang, Z.; Jin, X.; Li, L.; Ding, G.; Yang, Q. Multi-domain active learning for recommendation. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30.
126. Zhu, F.; Wang, Y.; Chen, C.; Liu, G.; Zheng, X. A graphical and attentional framework for dual-target cross-domain recommendation. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20), Yokohama, Japan, 7–15 January 2020; pp. 3001–3008.
127. Liu, M.; Li, J.; Li, G.; Pan, P. Cross domain recommendation via bi-directional transfer graph collaborative filtering networks. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual Event, Ireland, 19–23 October 2020; pp. 885–894.
128. Cui, Q.; Wei, T.; Zhang, Y.; Zhang, Q. HeroGRAPH: A Heterogeneous Graph Framework for Multi-Target Cross-Domain Recommendation. In Proceedings of the ORSUM@ RecSys, Online, 25–26 September 2020.
129. Jamali, M.; Ester, M. Trustwalker: A random walk model for combining trust-based and item-based recommendation. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June 2009–1 July 2009; pp. 397–406.
130. Wang, X.; Lu, W.; Ester, M.; Wang, C.; Chen, C. Social recommendation with strong and weak ties. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, Indianapolis, IN, USA, 24–28 October 2016; pp. 5–14.
131. Yang, X.; Steck, H.; Liu, Y. Circle-based recommendation in online social networks. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing China, 12–16 August 2012; pp. 1267–1275.
132. Berg, R.; Kipf, T.N.; Welling, M. Graph convolutional matrix completion. *arXiv* **2017**, arXiv:1706.02263.
133. Perozzi, B.; Al-Rfou, R.; Skiena, S. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 701–710.
134. Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; Mei, Q. Line: Large-scale information network embedding. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 1067–1077.
135. Xie, M.; Yin, H.; Wang, H.; Xu, F.; Chen, W.; Wang, S. Learning graph-based poi embedding for location-based recommendation. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, Indianapolis, IN, USA, 24–28 October 2016; pp. 15–24.
136. Yin, H.; Zou, L.; Nguyen, Q.V.H.; Huang, Z.; Zhou, X. Joint event-partner recommendation in event-based social networks. In Proceedings of the 2018 IEEE 34th International Conference on Data Engineering (ICDE), Paris, France, 16–19 April 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 929–940.
137. Tang, J.; Qu, M.; Mei, Q. Pte: Predictive text embedding through large-scale heterogeneous text networks. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, 10–13 August 2015; pp. 1165–1174.
138. Ma, R.; Qiu, X.; Zhang, Q. Co-attention memory network for multimodal microblog's hashtag recommendation. *IEEE Trans. Knowl. Data Eng.* **2019**, *33*, 388–400. [CrossRef]
139. Wang, Q.; Oàeilly-morgan, D.; Tragos, E.Z.; Hurley, N.; Smyth, B.; Lawlor, A.; Dong, R. Learning Domain-Independent Representations via Shared Weight Auto-Encoder for Transfer Learning in Recommender Systems. *IEEE Access* **2022**, *10*, 71961–71972. [CrossRef]
140. Ye, L.; Xie, H.; Lin, Y.; Lui, J.C. Rewarding Social Recommendation in OSNs: Empirical Evidences, Modeling and Optimization. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 4410–4424. [CrossRef]
141. Ye, Q.; Cao, Y.; Chen, Y. Deep Learning-Based User Privacy Settings Recommendation in Online Social Networks. In Proceedings of the 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 18–23 July 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–9.

142. Farmaki, A.; Olya, H.; Taheri, B. Unpacking the complex interactions among customers in online fan pages. *J. Bus. Res.* **2021**, *125*, 164–176. [CrossRef]
143. Velickovic, P.; Cucurull, G.; Casanova, A. Graph attention networks. *Stat* **2017**, *1050*, 10–48550.
144. Zhang, C.; Song, D.; Huang, C.; Swami, A.; Chawla, N.V. Heterogeneous graph neural network. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 793–803.
145. Wu, L.; Sun, P.; Fu, Y.; Hong, R.; Wang, X.; Wang, M. A neural influence diffusion model for social recommendation. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 235–244.
146. Fan, W.; Ma, Y.; Li, Q.; He, Y.; Zhao, E.; Tang, J.; Yin, D. Graph neural networks for social recommendation. In Proceedings of the The World Wide Web Conference, Lyon, France, 23–27 April 2019; pp. 417–426.
147. Abdollahpouri, H.; Adomavicius, G.; Burke, R.; Guy, I.; Jannach, D.; Kamishima, T.; Krasnodebski, J.; Pizzato, L. Multistakeholder recommendation: Survey and research directions. *User Model. User-Adapt. Interact.* **2020**, *30*, 127–158. [CrossRef]
148. Guo, Z.; Wang, H. A deep graph neural network-based mechanism for social recommendations. *IEEE Trans. Ind. Inform.* **2020**, *17*, 2776–2783. [CrossRef]
149. Lina, L.F.; Setiyanto, A. Privacy concerns in personalized advertising effectiveness on social media. *Sriwij. Int. J. Dyn. Econ. Bus.* **2021**, *5*, 147–156. [CrossRef]
150. Koohang, A.; Sargent, C.S.; Nord, J.H.; Paliszkiewicz, J. Internet of Things (IoT): From awareness to continued use. *Int. J. Inf. Manag.* **2022**, *62*, 102442. [CrossRef]
151. Zhong, W.; Yin, X.; Zhang, X.; Li, S.; Dou, W.; Wang, R.; Qi, L. Multi-dimensional quality-driven service recommendation with privacy-preservation in mobile edge environment. *Comput. Commun.* **2020**, *157*, 116–123. [CrossRef]
152. Hasan, M.K.; Alkhalifah, A.; Islam, S.; Babiker, N.B.; Habib, A.A.; Aman, A.H.M.; Hossain, M.A. Blockchain technology on smart grid, energy trading, and big data: Security issues, challenges, and recommendations. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 9065768. [CrossRef]
153. Zhang, J.; Askari, H.; Psounis, K.; Shafiq, Z. A Utility-Preserving Obfuscation Approach for YouTube Recommendations. *Proc. Priv. Enhancing Technol.* **2023**, *4*, 522–539. [CrossRef]
154. Seaton, D.B.; Caspi, A.; Casini, R.; Downs, C.; Gibson, S.E.; Gilbert, H.; Glesener, L.; Guidoni, S.E.; Hughes, J.M.; McKenzie, D.; et al. Improving Multi-Dimensional Data Formats, Access, and Assimilation Tools for the Twenty-First Century. *arXiv* **2023**, arXiv:2305.16535.
155. Hallifax, D.; Houston, J.B. Binding of drugs to hepatic microsomes: Comment and assessment of current prediction methodology with recommendation for improvement. *Drug Metab. Dispos.* **2006**, *34*, 724–726. [CrossRef]
156. Castells, P.; Hurley, N.; Vargas, S. Novelty and diversity in recommender systems. In *Recommender Systems Handbook*; Springer: New York, NY, USA, 2021; pp. 603–646.