A Survey on Accuracy-Oriented Neural Recommendation: From Collaborative Filtering to Information-Rich Recommendation

Le Wu, Member, IEEE, Xiangnan He, Member, IEEE, Xiang Wang, Member, IEEE, Kun Zhang, Member, IEEE, and Meng Wang, Fellow, IEEE

Abstract—Influenced by the great success of deep learning in computer vision and language understanding, research in recommendation has shifted to inventing new recommender models based on neural networks. In recent years, we have witnessed significant progress in developing neural recommender models, which generalize and surpass traditional recommender models owing to the strong representation power of neural networks. In this survey paper, we conduct a systematic review on neural recommender models from the perspective of recommendation modeling with the accuracy goal, aiming to summarize this field to facilitate researchers and practitioners working on recommender systems. Specifically, based on the data usage during recommendation modeling, we divide the work into collaborative filtering and information-rich recommendation: 1) collaborative filtering, which leverages the key source of user-item interaction data; 2) content enriched recommendation, which additionally utilizes the side information associated with users and items, like user profile and item knowledge graph; and 3) temporal/sequential recommendation, which accounts for the contextual information associated with an interaction, such as time, location, and the past interactions. After reviewing representative work for each type, we finally discuss some promising directions in this field.

Index Terms—Recommendation survey, deep learning, neural networks, neural recommendation models

1 INTRODUCTION

Information overload is an increasing problem in people’s everyday life due to the proliferation of the Internet. Recommender system serves as an effective solution to alleviate the information overload issue, to facilitate users seeking desired information, and to increase the traffic and revenue of service providers. It has been used in a wide range of applications, such as e-commerce, social media sites, news portals, app stores, digital libraries, and so on. It is one of the most ubiquitous user-centered artificial intelligence applications in modern information systems.

The research in recommendation can be dated back to 1990s [1], in the age the early work has developed many heuristics for content-based and Collaborative Filtering (CF) [2]. Popularized by the Netflix challenge, Matrix Factorization (MF) [3] later becomes the mainstream recommender model for a long time (from 2008 until 2016) [4], [5]. However, the linear nature of factorization models makes them less effective when dealing with large and complex data, e.g., the complex user-item interactions, and the items may contain complex semantics (e.g., texts and images) that require a thorough understanding. Around the same time in the mid-2010s, the rise of deep neural networks in machine learning (a.k.a., Deep Learning) has revolutionized several areas including speech recognition, computer vision, and natural language processing [6]. The great success of deep learning stems from the considerable expressiveness of neural networks, which are particularly advantageous for learning from large data with complicated patterns. This naturally brings new opportunities to advance the recommendation technologies. And not surprisingly, there emerges a lot of work on developing neural network approaches to recommender systems in the past several years. In this work, we aim to provide a systematic review on the recommender models that use neural networks — referred to as “neural recommender models”. This is the most thriving topic in current recommendation research, not only has many exciting progresses in recent years, but also shows the potential to be the technical foundations of the next-generation recommendation systems.

1.1 Differences With Existing Surveys

Given the significance and popularity of recommendation research, there are some recently published surveys also reviewed this area [2], [7], [8], [9], [10]. Here we shortly
discuss the main differences with these work to highlight the necessity and significance of this survey.

Existing surveys consist of two main parts. The first part focuses on the specific topics or directions, such as side information utilization in collaborative filtering [7], cross-domain recommendation [8], explainable recommendation [11], knowledge graph-enhanced recommendation [12], sequential recommendation [13], [14], and session-based recommendation [15]. The other part follows the taxonomy of Deep Learning (DL) to summarize the recommendation methods. For example, Zhang et al.[16] organized the discussions on recommendation methods into MLP based, autoencoder based, RNN based, attention based, etc. Similar surveys can also be found [17], [18]. These surveys mainly compare the technical difference of using various deep learning methods for recommendation.

Different from existing surveys, our survey is organized from the perspective of recommendation modeling with the accuracy goal, and covers the most typical recommendation scenarios, such as CF, content-enriched methods, and temporal/sequential methods. This will not only help researchers understanding why and when a deep learning technique would work but also facilitate practitioners designing better solutions for a specific recommendation scenario.

1.2 How Do We Collect the Papers?

Since our survey focuses on reviewing recommender system from the perspective of recommendation modeling with the accuracy goal, we retrieved most of the related top conferences such as WWW, SIGIR, KDD, ICLR, AAAI, IJCAI, WSDM, and RecSys, as well as the top journals such as TKDE, TKDD, and so on. Meanwhile, we also leveraged Google Scholar to search the recent related work. According to the categories that we made in this survey, we used key words such as collaborative filtering, content+RS, recommender systems, context+RS, side information, graph neural network, neural recommendation, etc, to search the relevant work. Then, based on the retrieved papers, we carefully design the topical structure to cover all papers as completely as possible. Besides, in order to avoid missing some important work, we also double-checked those classic and influential papers in recommendation.

1.3 Scope and Organization of This Survey

This survey is organized into two major parts: Sections 2, 3, and 4 review existing methods, and Section 5 discusses future directions and open issues. Before elaborating each section, we first give the problem formulation.

Regardless of the recommendation domain and scenario, we can abstract the “learning to recommend” problem as:

\[ \hat{y}_{u,i,c} = f(D_u, D_i, D_c), \]  

that is, learning the prediction function \( f \) to estimate the likelihood that a user \( u \) will favor an item \( i \) under the context \( c \), given the data \( D_u \), \( D_i \), and \( D_c \) to describe the user \( u \), item \( i \), and context \( c \), respectively. In doing so, we allow a unified framework to summarize neural recommendation models:

- Section 2 reviews collaborative filtering models, which forms the basis of personalized recommendation and is the most researched topic in recommendation. They can be seen as ignoring the context data \( D_c \) and using only the ID or interaction history in \( D_u \) and \( D_i \).
- Section 3 reviews the models that integrate the side information of users and items into recommendation, such as user profiles and social network, item attributes and knowledge graph. We term them as content-enriched models, which naturally extend collaborative filtering (CF) by integrating the side information into \( D_u \) and \( D_i \), whereas the context data \( D_c \) is also ignored.
- Section 4 reviews the models that use contextual information. The contextual data are associated with each user-item interaction, but do not belong to either user content or item content, like time, location, and the past interaction sequence [2]. The context-aware models make predictions based on the context data \( D_c \) in addition to the user-related data \( D_u \) and item-related data \( D_i \). Due to page limit, we focus on temporal context, which is one of the most common contextual data.

Fig. 1 illustrates the typical data used for recommendation modeling and three model types. It is worth noting that different models are designed for different recommendation scenarios. Nevertheless, in many cases we can make simple adjustments on a model’s component to make it suitable (at least technically viable) for another scenario. For example, many CF models are designed to first obtain user and item representations, and then the prediction function is learned given the user and item representations. To make them be content-enriched, we simply need to enhance the representation learning component with content modeling. Another example is that we can treat the contextual information as part of user data, i.e., constructing \( D_{u,c} \) to replace \( D_u \) to tweak content-enriched models to also be context-aware. Although these adjusted models may not be officially proposed or published, they can be obtained without much effort and worth exploring in real applications. Such design flexibility can be attributed to the layer-wise architecture of neural recommendation models, where different layers are designed for different aims. For convenience, we also summarize related neural recommendation models into the taxonomy of recommendation modeling.\(^1\) We hope this survey would provide a clear road-map to facilitate practitioners understanding and better designing models for their purpose.

2 COLLABORATIVE FILTERING MODELS

The concept of CF stems from the idea that leveraging collaborative behaviors of all users for predicting the behavior

\[^1\] https://github.com/lmcRS/AWS-recommendation-papers
of a target user. Early approaches directly calculate the behavior similarity of users (user-based CF) or items (item-based CF) with memory based models. Later on, matrix factorization based models become prevalent by collectively finding the latent spaces that encode user-item interaction matrix [3], [19]. Given the expressive complex modeling power of neural networks, the current solutions for neural CF can be summarized into two categories: representation modeling of users and items, and user-item interaction modeling given the representations.

### 2.1 Representation Learning

Let $\mathcal{U}$ and $\mathcal{V}$ denote users and items in CF, with $\mathbf{R} \in \mathbb{R}^{M \times N}$ is a user-item interaction behavior matrix. The general objective is to learn a user embedding matrix $\mathbf{P}$ and an item embedding matrix $\mathbf{Q}$, with $\mathbf{p}_u$ and $\mathbf{q}_i$ denote the representation parameters for user $u$ and item $i$, respectively.

In fact, as each user has limited behavior compared to the large item set, a key challenge that lies in CF is the sparsity of the user-item interaction behavior for accurate user and item embedding learning. Different kinds of representation learning models vary in input data, as well as the representation modeling techniques given the input data. We divide this section into three categories: history behavior aggregation enhanced models, autoencoder based models, and graph learning approaches. For ease of explanation, we list the typical representation learning models in Table 1.

#### 2.1.1 History Behavior Attention Aggregation Models

By taking the one-hot User ID (UID), and one-hot Item ID (IID) as input, classical latent factor models associate each UID $u$ and IID $i$ with a free embedding vector of $\mathbf{p}_u$ and $\mathbf{q}_i$ [3], [19]. Instead of modeling users with free embeddings, researchers further proposed borrowing users’ historical behavior for better user representation modeling. E.g., Factored Item Similarity Model (FISM) pools the interacted item embeddings as a user representation vector [20], and SVD++ [24] adds UID embedding $\mathbf{p}_u$ with the interaction history embedding (i.e., the FISM user representation) as the final user representation. These models relied on simple linear matrix factorization, and used heuristics or equal weights for the interaction history aggregation.

However, different historical items should contribute differently to model a user’s preference. Thus, some researchers integrate neural attention mechanism into history representation learning [25], [26], [42]. One representative work is Attentive Collaborative Filtering (ACF) [26], which assigns each interacted item with a user-aware attentive weight to indicate its importance to user representation:

$$
\hat{r}_{ui} = \left( \mathbf{p}_u + \sum_{j \in \mathcal{R}_u} \alpha(u,j) \mathbf{q}_j \right)^T \mathbf{q}_i, 
$$

where $\mathbf{p}_u$ is the ID embedding of user $u$, $\mathcal{R}_u$ denotes the items that $u$ has interacted with, $\alpha(u,j)$ is the attentive weight defined as:

$$
\alpha(u,j) = \frac{\exp(F(\mathbf{p}_u, \mathbf{q}_j))}{\sum_{j' \in \mathcal{R}_u} \exp(F(\mathbf{p}_u, \mathbf{q}_{j'}))},
$$

where $F(\cdot, \cdot)$ is a function that can be implemented as a MLP or simply inner product.

In practice, the influence of a historical item can be dependent on the target item, e.g., the purchase of a phone case is more related to the previous purchase of phone, while the purchase of a pant could be more related to the previous purchase of a shirt. As such, it may be beneficial to have dynamic user representation when considering the prediction on different target items. To this end, the Neural Attentive Item Similarity model (NAIS) model [25] revises the attention mechanism to be target item-aware:

$$
\hat{r}_{ui} = \left( \sum_{j \in \mathcal{R}_u} \alpha(i,j) \mathbf{q}_j \right)^T \mathbf{q}_i, 
$$

$$
\alpha(i,j) = \frac{\exp(F(\mathbf{q}_i, \mathbf{q}_j))}{\sum_{i' \in \mathcal{R}_u} \exp(F(\mathbf{q}_{i'}, \mathbf{q}_j))},
$$

where $\alpha(i,j)$ denotes the contribution of historical item $j$ to user representation when predicting a user’s preference on target item $i$. $\beta$ is a hyper-parameter between 0 and 1 (e.g., 0.5), for smoothing the interaction histories of different lengths. Similar attention mechanisms have been adopted for representation learning from interaction history, e.g., the Deep Item-based CF model (DeepICF) [42] and Deep Interest.
Agg denotes from representation models, this component aims at is an activation function.

\[
\mathbf{p}_u^{(l+1)} = \rho(\mathbf{W}_l^{(l)}[\mathbf{p}_u^{(l)}, \mathbf{a}_u^{(l+1)}]),
\]

where \( \mathbf{q}_j \) is item \( j \)'s representation at \( l \)th layer, \( R_u \) denotes items that connect to user \( u \) in the user-item bipartite graph. \( \mathbf{a}_u^{(l+1)} \) is the aggregation of connected items' representations in the \( l \)th layer, \( \mathbf{W}_l \) is an embedding transformation matrix that needs to be learned, and \( \rho(\cdot) \) is an activation function. After that, each user's (item's) final embedding can be seen as combining each entity's embedding at each layer.

The above steps can be seen as embedding propagation in the user-item bipartite graph. With a predefined layer \( L \), the up to \( L \)th order sub graph structure is directly encoded in the user and item embedding representation step. For example, SpectralCF utilized the spectral graph convolutions for CF [35], GC-MC [33] and NGCF [34] modeled the graph convolutions of user-item interactions in the original space, and are more effective and efficient in practice. Very recently, researchers argued that these neural graph based CF models differ from the classical GNNs as CF models do not contain any user or item features. Directly borrowing complex steps such as embedding transformation, and non-linear activations in GNNs may not be a good choice. Simplified neural graph CF models, including LR-GCCF [39], and LightGCN [40] have been proposed, which eliminate unnecessary deep learning operations. These simplified neural graph based models show superior performance in practice without the need of carefully chosen activation functions.

### 2.1 Interaction Modeling

#### 2.1.3 Graph Based Representation Learning

The CF effects are reflected in interaction histories of multiple users. As such, using collective interaction histories has the potential to improve the representation quality. From the perspective of user-item interaction graph, the individual interaction history is equivalent to the first-order connectivity of the user. Thus, a natural extension is to mine the higher-order connectivity from the user-item graph structure. For example, the second-order connectivity of a user consists of similar users who have co- interacted with the same items. Fortunately, with the success of Graph Neural Networks (GNNs) for modeling graph structure data in the community [44], many prior studies have been proposed to model the user-item bipartite graph structure for neural graph based representation learning. Given the user-item bipartite graph, let \( \mathbf{P}_0 \) and \( \mathbf{Q}_0 \) denote the free user latent matrix and item latent matrix as many classical latent factor based models, i.e., the 0th-order user and item embedding. These neural graph based models iteratively update the \((l + 1)\)th-order user (item) embedding as an aggregation of the \( l \)th-order item (user) embedding. For instance, each user \( u \)'s updated embedding \( \mathbf{p}_u^{(l+1)} \) is calculated as:

\[
\mathbf{a}_u^{(l+1)} = \text{Agg}([\mathbf{q}_j | j \in R_u]),
\]

where \( R_u \) is an embedding transformation matrix

#### 2.2 Interaction Modeling

#### 2.2.1 Distance Based Metrics

To solve the first issue, a line of research [45], [46], [47] borrows ideas from translation principles and uses distance metric as the interaction function. The inherent triangle inequality assumption plays an important role in helping capture underlying relationships among users and items. For instance, if user \( u \) tends to purchase items \( i \) and \( j \), the representations of \( i \) and \( j \) should be close in the latent space.
Towards this end, CML [45] minimizes the distance $d_{ui}$ between each user-item interaction $< u, i >$ in euclidean space as: $d_{ui} = \| p_u - q_i \|_2$. Instead of minimizing the distance between each observed user-item pair, TransRec exploits the translation principle to model the sequential behaviors of users [46]. In particular, the representation of user $u$ is treated as the translation vector between the representations of the items $i$ and the item $j$ to visit next, namely, $q_j + p_u \approx q_i$.

Distinct from CML that uses simple metric learning that assumes each user’s embedding is equally close to every item embedding she likes, LRML introduces the relation vectors $r$ to capture the relationships between user and item pairs [47]. More formally, the score function is defined as:

$$s_{ui} = \| p_u + e - q_i \|_2^2$$

where the relation vector $e \in \mathbb{R}^d$ is constructed using a neural attention mechanism over a memory matrix $M$. $M \in \mathbb{R}^{m \times d}$ is the trainable memory module, hence $E$ is the attentive sum of $m$ memory slots. As a result, the relation vectors not only ensure the triangle inequality, but also achieve better representation ability.

### 2.2.2 Neural Network Based Metrics

Distinct from the foregoing that employs linear metrics, recent studies adopt a diverse array of neural architectures, spanning from MLP, Convolutional Neural Network (CNN), and AE as the main building block to mine complex and nonlinear patterns of user-item interactions.

Researchers made attempts to replace similarity modeling between users and items with MLFs, as MLFs are general function approximators to model any complex continuous function. NCF is proposed to model the interaction function between each user-item pair with MLFs as: $\hat{r}_{ui} = f_{\text{MLF}}(p_u, q_i)$. Besides, NCF also incorporates a generic MF component into the interaction modeling, thereby making use of both linearity of MF and non-linearity of MLP to enhance recommendation quality.

Researchers also proposed to leverage CNN based architecture for interaction modeling. This kind of models first generate interaction maps via outer product of user and item embeddings, explicitly capturing the pairwise correlations between embedding dimensions [50], [51]. These CNN based CF model focuses on higher-order correlations among representation dimensions. However, such improvements on performance come at the cost of increasing model complexity and time cost.

Besides, a line of research exploits AEs to fulfill the blanks of user-item interaction matrix directly in the decoder part [27], [28], [29], [30], [52], [53], [54]. As the encoder and decoder can be implemented via neural networks, such stacks of nonlinear transformations give the recommenders more capacity to model the user representation from complex combinations of all historically interacted items.

**Summary:** Many recent studies have shown the superiority of GNNs in the representation learning of users and items. We ascribe the success to (1) the essential data structure, where the user-item interactions can be naturally represented as a bipartite graph between user and item nodes; and (2) GNNs can explicitly encode the crucial collaborative filtering signal of user-item interactions through information propagation process. As for interaction modeling, compared with the complex functions and metrics, simple inner product is much more efficient especially in the online and large-scale recommendation.

### 3 CONTENT-ENRICHED RECOMMENDATION

In collaborative filtering, item representations encode the collaborative signal — behavioral patterns of users — solely, but ignore the semantic relatedness. To enhance the representation learning, many researchers go beyond the user-item interactions and exploit auxiliary data. The auxiliary data could be classified into two categories: content based information and context-aware data. Specifically, the first category of content information is associated with users and items, including general user and item features, textual content (a.k.a, item tags, item textual descriptions and users’ reviews for items), multimedia descriptions (a.k.a, images, videos, and audio information), user social networks, and knowledge graphs. In contrast, contextual information shows the environment when users make item decisions, which usually denotes descriptions that beyond users and items [2]. Contextual information includes time, location, and specific data that are collected from sensors (such as speed, and weather), and so on. Due to page limits, we discuss the most typical contextual data: temporal data. In the following of the two sections, we would give a detailed summary of the content-enriched recommendation and context-aware recommendation. For the content-enriched recommendation, we classify the related work into five categories based on the available content information: the general features of users and items, the textual content information, the multimedia information, social networks and knowledge graphs.
### 3.1 Modeling General Feature Interactions

Factorization Machine (FM) provides an intuitive idea of feature interaction modeling [55]. As features are usually sparse, FM first embeds each feature $i$ into a latent embedding $v_i$, and models second-order interaction of any two feature instances with $x_i$ and $x_j$ as: $v_i^T \times v_j x_i x_j$. Naturally, FM models the second-order interactions, and reduces the parameter size of computing similarity of any two features with embedding based models. FM has been extended to field-aware FM by expanding each feature with several latent embeddings based on the field aware property [56], or higher-order FMs by directly expanding 2-order interactions with all feature interactions [66]. Despite the ability to model higher-order interactions, these models suffer from noisy feature interactions in the modeling process.

Researchers have explored the possibility of adopting neural models to automatically discover complex higher-order feature interactions for CTR prediction and recommendation. As shown in Table 3, besides FM based approaches, current related work on this topic can be classified into three categories: implicit MLP structures and explicit up to K-th order modeling, and tree enhanced models.

**MLP Based High Order Modeling.** As the feature interactions are hidden, researchers proposed to first embed each feature with an embedding layer, and then exploit MLPs to discover high order correlations. This category can be seen as modeling feature interactions in an implicit way as MLPs are black-box approaches, and we do not know what kind of feature interactions from the output of the MLP structure models. Since MLPs suffer from training difficulties, some researchers proposed pretraining techniques [58]. Others injected specific structures in MLPs for better capturing feature interactions. DeepCrossing designed residual structures to add back the original input after every two layers of MLPs [60]. The NFM architecture has a proposed bi-interaction operation before MLP layers [57]. PNN modeled both the bit-wise interactions of feature embedding interactions and vector-wise feature interactions [59]. Besides the complex high order interactions, another effective approach is to combine the MLP based high order modeling with the classical linear models [62], [63].

**Cross Network for K-th Order Modeling.** The cross network differs from the MLP based approaches with a carefully designed cross network operation, such that a K-th layer cross network models the up to K-th order feature interactions. The $q$th hidden layer output $x_q$ is calculated by the cross operation as: $x_q = x_{q-1} w_q + b_q + x_{q-1}$ [64]. Instead of operating cross operations at a bitwise level, xDeepFM applies cross interactions at the vector-wise level explicitly [65]. These kinds of models are able to learn bounded-degree feature interactions.

### 3.2 Modeling Textual Content

Neural network technique has revolutionized Natural Language Processing (NLP) [67], [68]. These neural NLP models enable multi-level automatic representation learning of textual content, and can be combined in the recommendation framework for better user and item semantic embedding learning. Given the above neural based NLP models, we discuss some typical textual enhanced recommendation models based on the above techniques. Textual content input for recommendation could be classified into two categories: the first category is the content descriptions associated with either items or users, such as the abstract of an article, or the content descriptions of a user. The second category links a user-item pair, such as users annotating tags to items, or writing reviews for products. For the second category, most models summarize the associated content with each user, and each item [69], [70]. Under such a situation, the second category of content information degenerates to the first category. In the following, we do not distinguish the input content data types, and summarize the related work for modeling contextual content into the following categories: autoencoder based models, word embeddings, attention models, and text explanations for recommendation.

**Autoencoder Based Models.** By treating item content as raw features, such as bag-of-words and item tag representations, these models use autoencoders and their variants to learn the bottleneck hidden content representations of items [30], [54], [71], [72], [73], [74], [75], [76], [77], [78]. For example, Collaborative Deep Learning (CDL) [71] is proposed to simultaneously learn each item’s embedding $q_i$ as a combination of two parts: a hidden representation from the item content $x_i$ with a stacked denoising autoencoder and an auxiliary embedding $\theta$ that is not encoded in the item content as:

$$q_i = f_i(x_i) + \theta_i, \quad \theta_i \sim \mathcal{N}(0, \sigma^2),$$

where $f_i(x)$ transforms raw content input into a bottleneck hidden vector with an autoencoder. $\theta_i$ is a free item latent vector that is not captured in the item content, which is

| Category                        | Modeling Summarization                          | Models                                      |
|---------------------------------|-------------------------------------------------|---------------------------------------------|
| Second order                    | Model second order correlations with embedding based similarity | FM [55], FFM [56]                          |
| MLP based higher order          | Design better initialization techniques to facilitate MLP modeling | NFM [57], FNN [58], PNN [59], DeepCrossing [60], [61] |
| Up to K-th order modeling       | Combine deep and shallow features               | Wide&Deep [62], DeepFM [63]                |
| Tree structure                  | Deep cross network structure for defined order depth | DCN [64], xDeepFM [65]                     |
|                                 | Tree enhanced embedding for attentive cross feature aggregation | TEM [11]                                    |
similar to many classical latent factor based CF models. In the model optimization process, the objective function is to simultaneously optimize the rating based loss from users’ historical behavior and the content-reconstruction loss from the autoencoder:

\[ 
\mathcal{L} = \mathcal{L}_R(\mathbf{R}, \bar{\mathbf{R}}) + \lambda \mathcal{L}_X(\mathbf{x}, f_d(f_c(\mathbf{X}))), 
\]

where \( \lambda \) is a parameter that measures the relative weight between the two loss terms. In the above optimization function, \( \mathbf{R} \) is the user-item rating matrix and \( \bar{\mathbf{R}} \) denotes the predicted rating. Similarly, \( \mathbf{X} \) is the item content input and \( f_d(f_c(\mathbf{X})) \) is an reconstructed content from an autoencoder that encodes the item content into a bottleneck representation \( f_c \), and then reconstructs it with a decoder \( f_d \).

Following this basic autoencoder based recommendation model, some studies proposed improvements to consider the uniqueness of the content information. For example, instead of learning a deterministic vector representation of the item content, a Collaborative Variational AutoEncoder (CVAE) is proposed to simultaneously recover the rating matrix and the side content information with a variational autoencoder [74]. Researchers also proposed to leverage the item neighbor information from item content to better represent the bottleneck representation of the item [79]. For some recommendation scenarios, items are also associated with category information. A denoising autoencoder with weak supervision is proposed to learn the distributed representation vector of each item [80]. Besides, as both users and items could be associated with content information, dual autoencoder based recommendation models have been proposed [30], [54], [73], [81].

Leveraging Word Embeddings for Recommendation. Autoencoders provide general neural solutions for unsupervised feature learning, which do not take the uniqueness of text input into consideration. Recently, researchers proposed to leverage word embedding techniques for better content recommendation [69], [82], [83], [84], [85], [86], [87], [88]. With the success of TextCNN [89], a Convolutional Matrix Factorization (ConvMF) is proposed to integrate CNN into probabilistic matrix factorization [82]. Let \( \mathbf{x}_i \) denote the text input of item \( i \). The item latent embedding matrix \( \mathbf{Q} \) is then represented as a Gaussian distribution that centers around its embedding representation as:

\[
 p(\mathbf{Q}|\mathbf{W}, \mathbf{X}, \sigma^2) = \prod_{i=1}^{V} \mathcal{N}(\mathbf{q}_i|\text{TextCNN} (\mathbf{W}, \mathbf{x}_i), \sigma^2),
\]

where \( \mathbf{W} \) is the parameters in TextCNN. Besides CNN based models, researchers also employed various state-of-the-art content embedding techniques, such as RNNs for item content representation [90].

Reviews widely appear in recommendation applications and are natural forms for users to express feelings about items. Given user’s rating records and associated reviews, most review based recommendation algorithms aggregate historical review text of users (items) as user content input \( D(u) \) (item content input \( D(i) \)). DeepCoNN [69] is a deep model for review based recommendation. As shown in Fig. 2, DeepCoNN consists of two parallel TextCNNs for content modeling: one focuses on learning user behaviors by exploiting review content \( D(u) \) written by user \( u \) and the other one learns item embedding from reviews \( D(i) \) written for item \( i \). After that, a factorization machine is proposed to learn the interaction between user and item latent vectors. Specifically, DeepCoNN can be formulated as:

\[
 \hat{r}_{ui} = \text{FM}(\text{TextCNN}(D_u), \text{TextCNN}(D_i)).
\]

Many studies have empirically found that the most predictive power of review text comes from the particular review of the target user to the target item. As the associated reviews of a user-item pair are not available in the test stage, TransNet is proposed to tackle the situation when the target review information is not available [83]. TransNet has a source network of DeppCoNN that does not include the joint review \( \text{rev}_{ui} \), and a target network that models the joint review of the current user-item pair \( (u, i) \). Therefore, the target network could approximate the predicted review \( \hat{\text{rev}}_{ui} \) for the test user-item pair even when users do not give reviews to items.

Attention Models. Attention mechanism has also been widely used in content enriched recommender systems. Given textual descriptions of an item, attention based models have been proposed to assign attentive weights to different pieces of content, such that informative elements are automatically selected for item content representation [91], [92], [93], [94], [95], [96], [97], [98]. For example, given a tweet, the attention based CNN learns the trigger words in the tweet for better hashtag recommendation [91]. With the historical rated items of a user, an attention model is proposed to selectively aggregate content representations of each historical item for user content preference embedding modeling [99], [100], [101]. Given user (item) collaborative embeddings, and content based embeddings, attention networks have also been designed to capture the correlation.
and alignment between these two kinds of data sources [98], [102]. Researchers have also proposed a co-evolutionary topical attention regularized matrix factorization model, with the user attentive features learned from an attention network that combines the user reviews, and the item attentive features learned from an attention network that combines the item reviews [102]. For review based recommendation, researchers argued that most content based user and item representation models neglected the interaction behavior between user-item pairs, and a dual attention model named DAML is proposed to learn the mutual enhanced user and item representations [103]. As item content sometimes is presented in multi-view forms (e.g., title, body, keywords and so on), multi-view attention networks are applied to learn unified item representations by aggregating multiple representations from different views [104], [105], [106]. With both the textual descriptions and the image visual information, co-attention is utilized to learn the correlation between the two modalities for better item representation learning [107], [108].

Text Explanations for Recommendation. Instead of improving recommendation accuracy with content input, there is a growing interest of providing text explanations for recommendation. Current solutions for explainable recommendations with text input can be classified into two categories: extraction based models and generation based models.

Extraction based models focus on selecting important text pieces for recommendation explanation. Attention techniques are widely used for extraction based explainable recommendation, with the learned attentive weights empirically showing the importance of different elements for model output [104], [109]. After that, the text pieces with larger attentive weights are extracted as recommendation explanations. Despite extracting text pieces from reviews, there exist other methods to extract useful text information for explanation, such as review-level explanations [109], [110].

With the huge success of language generation techniques [111], generation based models draw more and more attention [70], [112], [113], [114], [115], [116]. Given both users’ rating records and reviews, the key idea of these models is to design an encoder-decoder structure, with the encoder part encodes related embeddings of users and items, and the decoder generates reviews that are similar to the ground truth of the corresponding user-item review text. NRT is a state-of-the-art model that simultaneously predicts ratings and generates reviews [112]. By taking the one-hot user representation and item representation, the encoder part outputs the user latent embedding and item latent embedding, the review is generated with an RNN based decoder structure, and the rating is predicted with an MLP structure. Since we have both the ground truth rating records and the corresponding records of users, the two tasks of rating prediction and review generation can be trained in a multi-task framework. Meanwhile, additional information and more advanced encoder-decoder structures are also applied to explanation generation. For example, user and item attributes [113], [117] are multimodal item data [118], which are considered in the encoder. Then, an advanced attention selector [114] is designed in the decoder.

3.3 Modeling Multimedia Content

With the popularity of multimedia based platforms, visual content based multimedia contents, e.g., images and videos, are the most eye-catching for users. In the following, we introduce related work on modeling multimedia content in recommender systems. For ease of explanation, we summarize the related work on multimedia based recommendation with different kinds of input data in Table 4.

| Category | Model Summarization | Models |
|----------|---------------------|--------|
| Image    | CNN content based features | ACF [26], VBPR [119], OutfitNet [120] |
|          | Aesthetic based features pretrained from a deep aesthetic network | BDN [121] |
|          | CNN content based features and the style features from feature maps of CNNs | DMF [122] |
|          | Fine grained image attributes | SAERS [123], SNMO [124], AIC [125] |
|          | Co-attention networks for learning enhanced user and item representation | UVCAN [126] |
|          | CNNs to model visual relationships | PinSage [127], HGCN [128], TransGec [129] |
| Image+Behavior Time | CNN based temporal content evolution | BDN [121], [130] |
| Image+Text | Deep fusion networks to learn unified item representation | DMF [122], GraphCAR [131], CKE [132], Transfmc [133] |
|          | Co-attention networks for learning unified item representation | CoA-CAMN [107], Co-Attention [108] |
|          | Multi-task learning model with detailed image attributes | [134] |
|          | Text generation models by encoding user, item text and image content | VECF [117], KFR [135], MRG [118], [136] |
| Audio    | Learning deep audio feature | HLDN [137], [138], [139] |
| Video    | Attention networks to learn video representations from multiple image representations | ACF [26], JIFR [140], AGCN [141] |
|          | GNNS to learn video representation | AGCN [141] |
| Video+Audio | Deep fusion networks to learn unified item representation | CDML [142], [143] |

3.3.1 Modeling Image Information

The current solutions for image recommendation can be categorized into two categories: content based models and hybrid recommendation models. Content based models exploit visual signals for constructing item visual representations, and the user preference is represented in the visual space [108], [122], [123], [124], [131], [144], [145], [146], [147]. In contrast, the hybrid recommendation models alleviate the data sparsity issue in CF with item visual modeling [26], [119], [121], [130], [134], [148], [149].
Image Content Based Models. Image content based models are suitable for recommendation scenarios that rely heavily on visual influence (e.g., fashion recommendation) or new items with little user feedback. As visual images are often associated with text descriptions (e.g., tags, titles), researchers designed some unpersonalized recommender systems that suggest tags to images [108], [145]. These models apply CNNs to extract image visual information, and content embedding models to get textual embedding. Then, in order to model the correlation between visual and textual information, these models either project text and images into a same space [122], concatenate representations from different modalities [133] or design co-attention mechanism to better describe items [107], [108].

For personalized image recommendation, a typical solution is to project both users and items in the same visual space, with the item visual space derived from CNNs, and the user’s visual preference either modeled by the items they like [144] or a deep neural network that takes the user related profiles as input [131], [146]. Researchers have also argued that CNNs focus on the global item visual representation without fine-grained modeling. Therefore, some sophisticated image semantic understanding models have been proposed to enhance image recommendation performance [122], [123], [124], [125]. For instance, in order to suggest makeups for people, makeup related facial traits are first classified into structured coding. The facial attributes are then fed into a deep learning based recommendation system for personalized makeup synthesis [124]. In some visual based recommendation domains, such as the fashion domain, each product is associated with multiple semantic attributes [123], [125]. To exploit users’ semantic preferences for detailed fashion attributes, a semantic attributed explainable recommender system is proposed by projecting both users and items in a fine-grained interpretable semantic attribute space [123].

Hybrid Recommendation Models. Hybrid models utilize both the collaborative signals and the visual content for recommendation, which could alleviate the data sparsity issue in CF and improve recommendation performance. Some researchers proposed to first extract item visual information as features, and the item visual features are fed into factorization machines for recommendation. Instead of the inferior recommendation performance induced by the two step learning process, recent studies proposed end-to-end learning frameworks for hybrid visual recommendation [26], [119], [130], [134], [140], [148]. Visual Bayesian Personalized Ranking (VBPR) is one of the first few attempts that leverage the visual content for unified hybrid recommendation [119]. In VBPR, each user (item) is projected into two latent spaces: a visual space that is projected from the CNN based visual features, each user (item) is projected into two latent spaces: a visual and textual space. In VBPR, the visual content of the item has been leveraged as a regularization term in matrix factorization based models, ensuring that the learned item latent vector of each item is similar to the visual image representation learned from CNNs [130]. Besides learning the CNN content representations for item visual representation, many models have been proposed to consider additional information from the imagery for item visual representation, such as the pretrained aesthetics learned from a deep aesthetic network [121]. As users show time-synchronized comments on video frames, researchers proposed a multi-modal framework to simultaneously predict users’ preferences to key frames and generate personalized comments [135]. Compared to review generation models [112], the visual embedding is injected into both the user preference prediction part, as well as each hidden state of the LSTM architecture for better text generation.

Recently, GNNs have shown powerful performance in modeling graph data with heuristic graph convolution [150], [151]. PinSage is one of the first few attempts to apply GNNs for web-scale recommender systems [127]. Given an item-item correlation graph, PinSage takes node attributes as input, and iteratively generates node embeddings to learn the graph structure with iterative graph convolutions. Researchers also proposed to formulate a heterogeneous graph of users, outfits and items, and performed hierarchical GNNs for personalized outfit recommendation [128].

3.3.2 Video Recommendation

Researchers proposed content-based video recommender systems with rich visual and audio information [142], [143]. Specifically, these proposed models first extracted video features and audio features, and then adopted a neural network to fuse these two kinds of features with early fusion or late fusion techniques. As these content based video recommendation models do not rely on user-video interaction behavior, they can be applied to new video recommendation without any historical behavior data [142], [143]. In contrast to the content-based recommendation models, with user-video interaction records, researchers proposed an Attentive Collaborative Filtering (ACF) model for multimedia recommendation [26]. ACF leverages the attention mechanism with visual inputs to learn the attentive weights to summarize users’ preferences for historical items and the components of the item. The key idea of ACF is to leverage users’ multimedia behavior and explicitly project users into two spaces: a collaborative space and a visual space, such that users’ key frame preference could be approximated in visual space. The authors designed a model to discern both the collaborative and visual dimensions of users, and model how users make decisive item preferences from these two aspects [140].

3.4 Modeling Social Network

With the emergence of social networks, users like to perform item preferences on these social platforms and share
their interests with social connections. Social recommendation has emerged in these platforms, with the goal to model the social influence and social correlation among users to boost recommendation performance. The underlying reason for social recommendation is the existence of social influence among social neighbors, leading to the correlation of users’ interests in a social network [152], [153], [154], [155], [156], [157], [158]. We summarize social recommendation models into following two categories: the social correlation enhancement and regularization models, and GNN based models.

**Social Correlation Enhancement and Regularization.** By treating users’ social behavior as the social domain and item preference behavior as the item domain, the social correlation enhancement and regularization models tried to fuse users’ two kinds of behaviors from two domains in a unified representation. For each user, her latent embedding $p_u$ is composed of two parts: a free embedding $e_u$ from the item domain, and a social embedding $h_u$ that is similar with social connections in the social domain [152], [155], [157], [159], [160]. In other words, we have:

$$h_u = g(u, S)$$  \hspace{1cm} (13)$$

$$p_u = f(e_u, h_u).$$  \hspace{1cm} (14)

where $g$ models the social embedding part with the social network structure as input, and $f$ fuses the two kinds of embeddings, such as concatenation, addition or neural networks. Different models vary in the detailed implementation of the social domain representation $h_u$. For example, it can be directly learned from the social network embedding models [152], aggregated from the social neighbors’ embedding [155], [159], or transferred from the social domain to item domain with attention based transfer learning models [157]. Besides, the social network is also utilized as a regularization term in the model optimization process, with the assumption that connected users are more similar in the learned embedding space [152].

In the real-world, users’ interests are dynamic over time due to users’ personal interests change and the varying social influence strengths. Researchers extended the social correlation based model with RNN to model the evolution of users’ preferences under dynamic social influences [153], [154]. Specifically, for each user $u$, her latent preferences $h_u^t$ at time $t$ could be modeled as the transition from her previous latent preference $h_u^{t-1}$, as well as the social influence from social neighbors at $t - 1$ as:

$$h_u^t = f_{RNN}\left(R_u^t, h_u^{t-1}, \sum_{a \in S_u} t_{au} h_u^{t-1}\right)$$  \hspace{1cm} (15)

where $R_u^t$ is the temporal behaviors of user $u$ at this time, $\sum_{a \in S_u} t_{au} h_u^{t-1}$ denotes the influences from her social neighbors. In particular, the social influence strength $t_{au}$ could be simply set as equally for each social neighbor, or with attention modeling for influence strength inference.

**GNN Based Approaches.** Most of the above social recommendation models utilized the local first-order social neighbors for social recommendation. In the real world, the social diffusion process presents a dynamic recursive effect to influence a user’s decision. In other words, each user is influenced recursively by the global social network graph structure. To this end, researchers argued that it is better to leverage the GNN based models to better model the global social diffusion process for recommendation. DiffNet is designed to simulate how users are influenced by the recursive social diffusion process for social recommendation with the social GNN modeling. Specifically, DiffNet recursively diffuses the social influence from step 0 to the stable diffusion depth $K$. Let $h^0$ denote the user embedding at the $k$th diffusion process, which is modeled as:

$$h_u^0 = f_{NN}(x_u, e_u)$$  \hspace{1cm} (16)$$

$$h_u^{(k-1)} = \text{Pool}(h_u^{(k-1)}|a \in S_u)$$  \hspace{1cm} (17)$$

$$h_u^k = s(W^k[h_u^{k-1}, h_u^{(k-1)}])$$  \hspace{1cm} (18)$$

where $x_u$ is the user feature $x_u$ and user free latent vector $e_u$ with a neural network $f_{NN}$ for initial influence diffusion. At each diffusion step $k$, Eq. (17) models the influence diffusion from $u$’s social neighbors, and Eq. (18) depicts the user embedding at the recursive step $k$ by fusing her previous embedding $h_u^{k-1}$ and influences from her social neighbors as $h_u^{k-1}$. As $k$ diffuses from step 1 to depth $K$, the recursive social diffusion process is captured.

Instead of performing GNNs on the user-user social graph, researchers have also considered jointly modeling the social diffusion process in the social network and the interest diffusion process in the user-item graph with heterogeneous GNN based models [158], [161], [162], [166], [164], [165], [166]. For instance, DiffNet++ is proposed to jointly model the interest diffusion from user-item bipartite graph and the influence diffusion from the user-user social graph for user modeling in social recommendation, and have achieved state-of-the-art performance [164].

### 3.5 Modeling Knowledge Graph

Researchers have also considered leveraging Knowledge Graphs (KG) for recommendation, which provide rich side information for items (i.e., item attributes and external knowledge). Typically, KG organizes such subject-property-object facts in the form of directed graph $G = \{(h,r,t)|h \in \mathcal{H}, r \in \mathcal{R}, t \in \mathcal{T}\}$, where each triple presents that there is a relationship $r$ from head entity $h$ to tail entity $t$. Exploring such interlinks, as well as user-item interactions, being a promising solution to enrich item profile and enhance the relationships between users and items. Furthermore, such graph structure endows recommender systems the ability of reasoning and explainability [167], [168], [169], [170], [171], [172], [173]. Recent efforts for KG enhanced recommendation can be roughly categorized into three categories: path-based models [168], [174], [175], [176], [177], regularization-based models [132], [169], [178], [179], and GNN-based approaches [97], [128], [141], [170], [180], [181], [182], [183], [184], [185], [186], [187].

**Path Based Methods.** Many efforts introduce meta-paths [174], [176], [177], [188], [189], [190], [191], [192] and paths [168], [175], [193], [194], [195] that present high-order connectivity between users and items, and then feed them into predictive models to directly infer user preferences. In particular, a path from user $u$ to item $i$ can be defined as a
sequence of entities and relations: \( p = [e_1, e_2, e_3, \ldots, e_{L-1}, e_L] \), where \( e_1 = u \) and \( e_L = i \), and \( (e_l, r_l, e_{l+1}) \) is the \( l \)-th triplet in \( p \), and \( L - 1 \) denotes the number of triplets in the path. As such, the set of paths connecting \( u \) and \( i \) can be defined as \( \mathcal{P}(u, i) = \{p\} \).

FMG [176], MRec [177], and KPRN [168] convert the path set into an embedding vector to represent the user-item connectivity. Such paradigm can be summarized as follows:

\[
c = f_{\text{pooling}}(\{f_{\text{Embed}}(p) | p \in \mathcal{P}(u, i)\}),
\]

(19)

where \( f_{\text{Embed}}(\cdot) \) embeds path \( p \) as a trainable vector. \( f_{\text{pooling}}(\cdot) \) is the pooling operation to synthesize all path information into the connectivity representation, such as the attention networks adopted in MRec and KPRN. RippleNet [196] constructs ripple set (i.e., high-order neighboring items derived from \( \mathcal{P} \)) for each user to enrich her representations.

While explicitly modeling high-order connectivity, it is highly challenging in real-world recommendation scenarios because most of these methods require extensive domain knowledge to define meta-paths or labor-intensive feature engineering to obtain qualified paths [167], [170]. Moreover, the scale of paths can easily reach millions or even larger when a large number of KG entities are involved, making it prohibitive to efficiently transfer knowledge.

**Regularization Based Methods.** This research line devises a joint learning framework, where direct user-item interactions are used to optimize the recommender loss, and KG triples are utilized as additional loss terms to regularize the recommender model learning. In particular, the anchors between two modeling components are the embeddings of the overlapped items. CKE [132] makes use of Knowledge Graph Embedding (KGE) techniques, especially TransR [197], to generate additional representations of items, and then integrates them with item embeddings of the recommender MF, which is defined as:

\[
q_i = f_{\text{Embed}}(i) + f_{\text{KGE}}(i|\mathcal{G}),
\]

(20)

where \( f_{\text{Embed}}(\cdot) \) is the embedding function which takes the item ID as the input, while \( f_{\text{KGE}}(\cdot) \) is the output of KGE method which considers the KG structure. Similarly, DKN [198] generates item embeddings from NCF and TransE. These approaches focus on enriching item representations by the joint learning framework.

**GNN Based Methods.** The regularization-based methods only take direct connectivity between entities into consideration, while encoding the high-order connectivity in a rather implicit manner. Due to the lack of explicit modeling, neither the long-range connectivities are guaranteed to be captured, nor the results of high-order modeling are interpretable [170]. More recent studies, such as KGAT [170], CKAN [199], MKM-SR [200], and KGCF [180], get inspired by the advances of GNNs and explore the message-passing mechanism over graphs to exploit high-order connectivity in an end-to-end fashion.

KGAT [170] encodes user-item interactions and KG as a unified relational graph by representing each user behavior as a triplet, \((u, \text{Interact}, i)\). Based on the item-entity alignment set, the user-item bipartite graph can be seamlessly integrated with KG as a so-called collaborative knowledge graph \( \mathcal{G} = \{(h, t, r) | (h, t) \in \mathcal{E}', r \in \mathcal{R}'\} \), where \( \mathcal{E}' = \mathcal{E} \cup \mathcal{U} \) and \( \mathcal{R}' = \mathcal{R} \cup \{\text{Interact}\} \). Over such graph, KGAT recursively propagates the embeddings from a node’s neighbors (which can be users, items, or other entities) to refine the node’s embedding, and employs an attention mechanism to discriminate the importance of the neighbors as:

\[
P_u = f_{\text{GNN}}(u, \mathcal{G}),
\]

(21)

where \( f_{\text{GNN}}(\cdot) \) is the GNN component.

**Summary.** Auxiliary data, such as text, multimedia, and social network, is capable of enhancing the user and item representation learning and boosting the recommendation performance. The keys are the selection of auxiliary data and the integration methods. For example, text information can help models to generate corresponding recommendation explanation. Social network information is very useful to provide social influence and social correlation among users for better recommendation. Meanwhile, attention mechanism is a general method to select the most relevant information from auxiliary data to enhance the recommendation learning. GNN-based methods are good at obtaining structure information and high-order correlation for the utilization of auxiliary data. As a conclusion, based on the recommendation target (recommendation accuracy, explanation, cold-start problem, etc), selecting proper auxiliary data and integration method can help recommendation models to achieve a good performance.

## 4 Temporal/Sequential Models

Users’ preferences are not static but evolve over time. Instead of modeling users’ static preferences with the aforementioned models, temporal/sequential based recommendation focuses on modeling users’ dynamic preferences or sequential patterns over time. Given a userset \( U = [u_1, u_2, \ldots, u_M] \) and an itemset \( V = [i_1, i_2, \ldots, i_N] \), current temporal/sequential recommendation could be generally classified into three categories:

- **Temporal based recommendation:** For a user \( u \in U \) and an item \( i \in V \), the associated user-item interaction behavior is denoted as a quadri-tuple as \([u, i, r_{ui}, t_{ui}]\). In this representation, \( r_{ui} \) denotes the detailed rating and \( t_{ui} \) is the timestamp of this behavior. Temporal recommendation focuses on modeling the temporal dynamics of users’ behavior over time.

- **Session based recommendation:** In a certain session \( f = [i_1, i_2, \ldots, i_s] \subseteq V \), a user interacts with a collection of items (e.g., consumption with a shopping basket, browsing the internet in a limited time period). In many session based applications, users do not log in and user IDs are not available [201], [202], [203]. Therefore, the popular direction of session based recommendation is to mine the sequential item-item interaction patterns from the session data for better recommendation.

- **Temporal and session based recommendation:** This approach combines the definition of temporal recommendation and session recommendation, in which each transaction is described as \([u, s, t]\), with \( s \subseteq V \) is a collection of items that are consumed at a particular
time $t$. Under this scenario, both the temporal evolution and the sequential patterns of items need to be captured.

We summarize the main techniques for modeling temporal and sequential effects in recommender systems in Table 5 and illustrate some representative work in Fig. 3.

### 4.1 Temporal Based Recommendation

Temporal recommendation models focus on capturing the temporal evolution of users’ preferences over time. Due to the superior of RNNs in modeling temporal patterns, many temporal based approaches take RNNs into consideration. Recurrent Recommender Networks (RRN) is one of the representative studies for temporal recommendation by endowing both users and items with an LSTM autoregressive architecture [204]. In RRN, the predicted rating $r_{tu}^t$ of user $u$ to item $i$ at time $t$ is modeled as:

$$r_{tu}^t = f(p_{tu}^t, q_{ti}^t) \quad \text{where} \quad p_{tu}^t = RNN(p_{tu}^{t-1}, Wx_{tu}) \quad \text{and} \quad q_{ti}^t = RNN(q_{ti}^{t-1}, Wx_i)$$

where $p_{tu}^t$ and $q_{ti}^t$ are the dynamic embeddings of user $u$ and item $i$ at time $t$, respectively. Specifically, $f$ in Eq. (22) is a temporal rating prediction function. Eq. (23) models the evolution of users and items’ dynamic embeddings with RNN architecture. As the user side and item side share similar LSTM structure, we take the user side as an example.

$x_u^t \in \mathbb{R}^{V_l}$ is a rating vector for $u$ between $t-1$ and current time $t$, with each element $x_{ul}^t$ denotes the rating of user $u$ to each item $l$ at that time. $W$ is a transformation matrix that needs to be learned. Therefore, RRN learns the evolution of user and item latent vectors over time with two RNNs. Based on RRNs, rich context factors were considered, such as the social influence [153], item metadata [208], [250] and multimedia data fusion [251]. Take the RNN in the user side as an example, and we can generalize the user latent embedding evolution as:

$$p_{tu}^t = RNN(p_{tu}^{t-1}, Wx_{tu}, ContextualEmbedding),$$

where additional contextual embeddings are also injected to model temporal evolution of users’ temporal embedding.

Recently, an emerging trend is to model the temporal evolution with Neural Turning Machines [252] and Memory Networks [253]. Compared to RNNs, memory networks introduce a memory matrix to store the states in memory slots, and update memories over time with read and write operations. As the memory storage is limited, the key component in applying memory networks in recommendation is how to update memories over time with users’ temporal behavior. Researchers proposed a general memory augmented neural network with user memory networks to store and update users’ historical records, and the user memory network is implemented from the item and feature level [211]. Researchers further proposed to use attention
mechanism in the memory reading and writing process with soft-addressing, in order to better capture users’ long-term stable and short-term temporal interests [210].

4.2 Session Based Recommendation

Many real-world recommender systems often encounter the short session data from anonymous users, i.e., the user ID information is not available. Session based recommendation is popular under this situation, which models the sequential item transition patterns given many session records. Hidasi et al. [201] made one of the first few attempts to design GRU4REC for session based recommendation under the RNN based framework. Specifically, GRU4REC resembles an RNN structure, which recursively takes the current item in the session as input, updates the hidden states, and outputs the predicted next item based on the hidden state. Given anonymous sessions, the key component of GRU4REC is how to construct mini-batches to suit the data forms of RNNs. Since the goal is to capture how a session evolves over time with item dependencies, the authors designed a session parallel mini-batch. The first events of the first several sessions are extracted to form the first mini-batch, with the desired output is the second event of the corresponding session. Under such a formulation, the complex correlations of items in a session are captured for session based recommendation.

GRU4REC has been further investigated with item feature consideration [202], local intent [214], user information consideration [254], data augmentation techniques [215]. By treating item ID, name, and category with an embedding matrix, a sequence of clicks could be represented as frames. Therefore, the architecture of 3D CNNs could be transferred to session-based recommendation [220]. Furthermore, a self-attention based sequential model of SASRec is proposed. SASRec models the entire user sequence without any recurrent and convolutional operations, and adaptively considers consumed items for recommendation [221].

Researchers also proposed a translation based model to capture the personalized sequential third order interactions between a user \( u \), the previous item \( j \), and the current item \( i \). Given the item embedding matrix \( Q \), each user’s embedding \( p_u \) can be approximated as: \( q_i + p_u \approx q_j \) [46]. Therefore, the translation based models capture the correlation of two constructive items.

While above models built relationships between consecutive items in a session, how to globally model the transitions in a session among distant items remain under explored. Researchers adopted GNNs for session based recommendation [203], [224], [225], [226], [227], [228], [255], [256]. SRGNN is one of the first few attempts. As shown in Fig. 3, the graph is constructed by taking all items as the graph node set, and there is an edge between two nodes if these two nodes appear in consecutive orders in a session. Then, the GNN is adopted to learn item embeddings, such that the higher-order relationships of items from session behavior data can be modeled [203]. Different GNN based models vary in graph construction, and graph aggregation process [224], [225], [226].

4.3 Temporal and Session Based Recommendation

Given the session data of each user over time, models in this category leverage both the temporal evolution modeling of users, as well as the sequential item patterns hidden in the sessions for recommendation. Currently, the solutions could be classified into two categories: the first category learns both users’ long term preference and the short term dynamic preferences, and the second category adopts advanced neural models for learning a unified user representation.

In the first category, each user’s long term preference is modeled from her historical behaviors, and the short term dynamics is modeled from the previous session or the current session [229], [230], [232]. For example, researchers proposed hierarchical attention networks for temporal and session based recommendation, with the first attention layer learns the user long term preference based on historical records, and the second one attentively aggregates user representation from the current session as:

\[
p_i' = Att_2(p_u, Att_1(q_i, l \in T^{(t-1)}_u)),
\]

where \( Att_1 \) denotes the bottom layer attention network that depicts the user’s short term preference from recent user behavior \( T^{(t-1)}_u \), and \( Att_2 \) is a top layer attention network that balances the short term user preference and long term preference embedding vector \( p_u \). Instead of using hierarchical attentions, researchers proposed to adopt attention techniques to learn item correlations, and designed recurrent states at top layers for sequential recommendation [240].

Hierarchical RNNs are also proposed for personalized session-based recommendation over time, with a session level GRU unit to model the user activity within sessions, and a user level GRU models the evolution of the user preference over time [200], [236]. Besides, researchers exploited hierarchical attention networks to learn better short term user preference with feature-level attention and item level attention [231]. For the long term user interest modeling, researchers proposed to leverage nearby sessions [235], designed attention modeling or memory addressing techniques to find related sessions [237], [244], [245], [257].

Another kind of models utilize the 3D convolutional networks for recommendation, which defines the recommendation problem as [51], [246]: \( \{S_{L}: L=1, \ldots, L\} \rightarrow S_V \), where \( S_V \subseteq V \) is the \( t \)-th time sequential behavior of user \( u \) at time \( t \), and \( L \) denotes the maximum sequence length. Convolutional Sequence Embedding Recommendation (Caser) is a representative work that incorporates CNNs to learn the sequential patterns. It captures both user’s general preferences and sequential patterns, at both the union level and point level with convolution operations, and captures the skip behavior [51], [246].

Besides, researchers proposed to leverage the advances of GNN based models for recommendation [247], [248], [249]. The graph structure is constructed from all sessions to form a global item correlation graph or graphs at each time period. For example, researchers constructed time-aware hypergraphs to model item correlations over time. After that, the self-attention modules are used to model users’ dynamic interests based on the learned dynamic item embeddings over time [247].

Summary: Temporal/sequential based models focus on the dynamic preferences of users over times. Therefore, most existing work concentrates on the sequential information of users and items, and leverages sequential models
users and items. Incorporating self-supervised learning into recommendation could offer promising solutions to the long-standing issues of data sparsity and long-tail distribution.

**Evaluation: Multi-Objective Goals for Social Good Recommendation.** Recommender systems have penetrated every aspect in our daily life, and have greatly shaped the decision process of providers and users. Most previous recommender systems concentrated on the single goal of recommendation accuracy based user experience. These systems limit the ability to incorporate user satisfaction from multiple goals, e.g., recommendation diversity and explanations to persuade users [9]. Besides, the user-centric approach neglects system objectives from multistakeholders and the society. The data-driven approaches with accuracy as goals may lead to biases in the algorithmic process decision process [270], [271], [272], [273]. For recommender systems, researchers have realized that long-tailed items have fewer chances to be recommended, and benefiting users may obscure concerns that might come from other stakeholders in this system. How to provide multi-objective goals for social good recommendation, such as explainability, balance of multistakeholders, and fairness for the society is an important research topic that needs to be paid attention to.

**Discussion: Reproducibility.** While the neural recommender models have dominated in the recommendation field and claimed substantial improvements over previous models, recent efforts raise questions about their reproducibility and published claims [258], [259], [260], [261], [274]. This can be attributed to two aspects. First, neural recommendation models are based on neural networks, which are hard to tune in practice. Thus, we should carefully choose the initialization, tune hyperparameters, avoid model collapse, and so on. Besides, due to the various application scenarios of recommendation, different models vary in the selection of datasets and setting of experiments. Specifically, it is well known that recommender models are sensitive to the dataset size, the dataset sparsity, the data preprocessing and splitting techniques, the strategy of negative sampling, the choice of loss function and optimization manner, and the evaluation metrics of performance. Thus, it is very challenging to conduct a fair performance comparison. In order to advance the recommendation community, some researchers make efforts on the data level, such as industry-relevant recommendation benchmark [275], Microsoft News Dataset (MIND) [276], and Yelp dataset. Others concentrate on the unified evaluation framework [277], [278]. For example, researchers argue that previously default choice of evaluating recommender models with sampled metrics (e.g., rather than using the full set, only sampling a small set of negative items during testing) would be inconsistent to the true trend [279]. Towards fair and reproducible comparisons, it is of crucial importance to make the experimental settings transparent (e.g., release the codes, datasets, and experimental settings, and set up a leaderboard if possible). Furthermore, beyond network architecture engineering and hunting for the “best” performance, research studies on theoretical considerations and reproducibility analysis should be encouraged.

2. https://www.yelp.com/dataset
6 Conclusion

In this survey, we provide a systematic review on neural recommender models from the perspective of recommendation modeling with accuracy goal. Based on the data usage, we organize existing work into three categories: collaborative filtering model, content enriched model, and temporal/sequential model. In each part, we summarize a bunch of influential research work and conclude corresponding main contributions as well as our opinions. Moreover, we also elaborate possible promising directions from the basics, modeling, and evaluation perspectives, and reproducibility problem in recommender systems. Still, a large number of novel methods and techniques are proposed each year. We hope this survey is able to help reader to quickly understand the development and key aspects of recommendation modeling, and inspires some future studies.

Acknowledgments

Le Wu greatly thanks the support of Young Elite Scientists Sponsorship by CAST and ISZS.

References

[1] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, “Using collaborative filtering to weave an information tapestry,” Commun. ACM, vol. 35, no. 12, pp. 61–70, 1992.
[2] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,” IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734–749, Jun. 2005.
[3] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[4] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver, “Multiverse recommendation: N-dimensional tensor factorization for context-aware collaborative filtering,” in Proc. 4th ACM Conf. Recommender Syst., 2010, pp. 79–86.
[5] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, “BPR: Bayesian personalized ranking from implicit feedback,” in Proc. 25th Conf. Uncertainty Artif. Intell., 2009, pp. 456–461.
[6] S. Kabbur, X. Ning, and K. G. Karypis, “FISM: Factorized item similarity models for top-n recommender systems,” in Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2013, pp. 659–667.
[7] C. Ma, L. Ma, Y. Zhang, R. Tang, X. Liu, and M. Coates, “Probabilistic metric learning with adaptive margin for top-k recommendation,” in Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2020, pp. 1036–1045.
[8] L. Chen, X. Xie, and X. Xiong, “Product quantized collaborative filtering,” IEEE Trans. Knowl. Data Eng., vol. 33, pp. 3284–3296, Sep. 2021.
[9] J. Chen, C. Wang, S. Zhou, Q. Shi, J. Chen, Y. Feng, and C. Chen, “Fast adaptively weighted matrix factorization for recommendation with implicit feedback,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 3470–3477.
[10] S. Rendle et al., “Item interaction attention graph neural network,” in Proc. 36th Conf. Neural Inf. Process. Syst., 2023, pp. 9702–9716.
[11] S. Zhang, L. Yao, A. Sun, and Y. Tay, “Deep learning based recommender system: A survey and new perspectives,” ACM Comput. Surv., vol. 52, no. 1, pp. 51–538, 2019.
[12] Z. Batmaz, A. Yurekli, A. Bilge, and C. Kaleli, “A review on deep learning for recommender systems: Challenges and remedies,” Artif. Intell. Rev., vol. 52, no. 1, pp. 1–37, 2019.
[13] A. Da’u and N. Salim, “Recommendation system based on deep learning methods: A systematic review and new directions,” Artif. Intell. Rev., vol. 53, no. 4, pp. 2709–2748, 2020.
[14] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, “BPR: Bayesian personalized ranking from implicit feedback,” in Proc. 25th Conf. Uncertainty Artif. Intell., 2009, pp. 456–461.
T. Bansal, D. Belanger, and A. McCallum, “Ask the GRU: Multi-task learning for deep text recommendations,” in Proc. ACM Conf. Recommender Syst., 2016, pp. 107–114.

Y. Gong and Q. Zhang, “Hashtag recommendation using attention-based convolutional neural network,” in Proc. Int. Joint Conf. Artif. Intell., 2017, pp. 2792–2798.

S. Seo, J. Huang, H. Yang, and Y. Liu, “Interpretable convolutional neural networks with dual local and global attention for review rating prediction,” in Proc. 4th ACM Conf. Recommender Syst., 2017, pp. 297–305.

Y. Li, T. Liu, J. Jiang, and L. Zhang, “Hashtag recommendation with topical attention-based LSTM,” in Proc. Int. Conf. Comput. Linguistics, 2016, pp. 3019–3029.

C. Qin et al., “A personalized question recommender system for intelligent job interview,” in Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 2165–2173.

H. Wang, F. Wu, Z. Liu, and X. Xie, “Fine-grained interest matching for neural news recommendation,” in Proc. Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 836–845.

T. Qi, F. Wu, C. Wu, Y. Huang, and X. Xie, “Privacy-preserving multi-view learning between ratings and reviews,” in Proc. Conf. Empir. Methods Natural Lang. Process., 2020, pp. 1423–1432.

D. Lee, B. Oh, S. Seo, and K.-H. Lee, “News recommendation with topic-enriched knowledge graphs,” in Proc. 29th ACM Int. Conf. Inf. Knowl. Manage., 2020, pp. 695–704.

Z. Zheng, Y. Ding, X. He, L. Zhu, X. Song, and M. S. Kankanhalli, “A3+SCAN: An adaptive aspect attention model for rating prediction,” in Proc. Int. Joint Conf. Artif. Intell., 2019, pp. 3748–3754.

Q. Zhu, X. Zhou, Z. Song, J. Tan, and L. Guo, “DAN: Deep attention neural network for news recommendation,” in Proc. AAAI Conf. Artif. Intell., 2019, pp. 5973–5980.

X. Chen, Y. Zhang, and Z. Qin, “Dynamic explainable recommendation based on neural attentive models,” in Proc. AAAI Conf. Artif. Intell., 2019, pp. 53–60.

C. Wu, F. Wu, M. An, J. Huang, Y. Huang, and X. Xie, “NPA: Neural news recommendation with personalized attention,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 2576–2584.

Y. Lu, R. Dong, and B. Smyth, “Co-evolutionary recommendation model: Mutual learning between ratings and reviews,” in Proc. Int. Conf. World Wide Web, 2018, pp. 773–782.

D. Liu, J. Li, B. Du, J. Chang, and R. Gao, “DAML: Dual attention recommendation for multimodal microblog with cross-attention memory network,” in Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 344–352.

L. Hu, S. Jian, L. Cao, and Q. Chen, “Interpretable recommendation via attraction modeling: Learning multilevel attractiveness over multimodal movie contents,” in Proc. Int. Joint Conf. Artif. Intell., 2019, pp. 3485–3496.

C. Wu, F. Wu, M. An, J. Huang, Y. Huang, and X. Xie, “Neural news recommendation with attentive multi-view learning,” in Proc. Int. Joint Conf. Artif. Intell., 2019, pp. 3863–3869.

J. Gao et al., “Set-sequence-graph: A multi-view approach towards exploiting reviews for recommendation,” in Proc. 29th ACM Int. Conf. Inf. Knowl. Manage., 2020, pp. 395–404.

R. Ma, Q. Zhang, J. Wang, L. Cui, and X. Huang, “Mention recommendation for multimodal microblog with cross-attention memory network,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2018, pp. 195–204.

Q. Zhang, J. Wang, H. Huang, X. Huang, and Y. Gong, “Hashtag recommendation for multimodal microblog using co-attention network,” in Proc. Int. Joint Conf. Artif. Intell., 2017, pp. 3420–3426.

C. Chen, M. Zhang, Y. Liu, and S. Ma, “Neural attentional rating regression with review-level explanation,” in Proc. 19th Int. Conf. World Wide Web, 2018, pp. 1583–1592.

X. Wang, Y. Chen, J. Yang, L. Wu, Z. Wu, and X. Xie, “A reinforcement learning framework for explainable recommendation,” in Proc. IEEE Int. Conf. Data Mining, 2018, pp. 587–596.

D. Badhannu, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” in Proc. Int. Conf. Learn. Represent., 2015.

P. Li, Z. Wang, Z. Ren, L. Bing, and W. Lam, “Neural rating regression with abstractive tips generation for recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2017, pp. 345–354.

P. Li, Z. Wang, L. Bing, and W. Lam, “Persona-aware tips generation,” in Proc. 19th Int. Conf. World Wide Web, 2019, pp. 1006–1016.

Z. Chen et al., “Co-attentive multi-task learning for explainable recommendation,” in Proc. Int. Joint Conf. Artif. Intell., 2019, pp. 2137–2143.

L. Li, Y. Zhang, and L. Chen, “Generate neural template explanations for recommendation,” in Proc. ACM Int. Conf. Knowl. Discov. Data Mining, 2020, pp. 755–764.

Z. Chen et al., “Towards explainable conversational recommendation,” in Proc. Int. Joint Conf. Artif. Intell., 2020, pp. 2994–3000.

X. Chen et al., “Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2019, pp. 765–774.

Q.-T. Truong and H. Lauw, “Multimodal review generation for DMO recommender systems,” in Proc. Int. Conf. World Wide Web, 2019, pp. 1864–1874.

R. He and J. McAuley, “VBP: Visual bayesian personalized ranking from implicit feedback,” in Proc. AAAI Conf. Artif. Intell., 2016, pp. 144–150.

Y. Lin, M. Moosaei, and H. Yang, “OutfitMet: Fashion outfit recommendation with attention-based multiple instance learning,” in Proc. Int. Conf. World Wide Web, 2018, pp. 649–658.

W. Yu, H. Zhang, X. He, X. Chen, L. Xiong, and Z. Qin, “Aesthetic-based clothing recommendation,” in Proc. Int. Conf. World Wide Web, 2018, pp. 653–662.

D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation,” in Proc. Conf. World Wide Web, 2016, pp. 107–114.

S. Liu, Z. Chen, H. Liu, and X. Hu, “User-video co-attention network for personalized micro-video recommendation,” in Proc. Int. Conf. World Wide Web, 2019, pp. 3020–3026.

R. Ying, R. He, K. Chen, P. Eksoimbatchai, W. L. Hamilton, and J. Leskovec, “Graph convolutional neural networks for web-scale recommender systems,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2018, pp. 974–983.

X. Li, X. Wang, X. He, L. Chen, X. Xiao, and T.-S. Chua, “Hierarchical fashion graph network for personalized outfit recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2020, pp. 3943–3946.

L. Wu, Y. Yuan, L. Chen, D. Lian, R. Hong, and M. Wang, “Learning to transfer graph embeddings for inductive graph based recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2020, pp. 1211–1220.

R. He and J. McAuley, “UPS and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering,” in Proc. 19th Int. Conf. World Wide Web, 2016, pp. 507–517.

Q. Xu, F. Shen, L. Liu, and H. T. Shen, “Graphpcr: Content-aware multimedia recommendation with graph autoencoder,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2018, pp. 981–984.

F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, “Collaborative knowledge base embedding for recommender systems,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 353–362.

X. Yang, Y. Ma, L. Xiao, and T.-S. Chua, “Transnfcm: Translation-based neural fashion compatibility modeling,” in Proc. AAAI Conf. Artif. Intell., 2019, pp. 403–410.

A. Cardoso, F. Daolio, and S. Vargas, “Product characterisation towards personalisation: Learning attributes from unstructured data to recommend fashion products,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2017, pp. 80–89.

X. Chen, Y. Zhang, Q. Ai, H. Xu, J. Yan, and Z. Qin, “Personalized key frame recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2017, pp. 315–324.

T. Yu, Y. Shen, and H. Jin, “Towards hands-free visual dialog interactive recommendation,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 1137–1144.
Knowledge-aware graph neural networks with...

L. Wu, P. Sun, Y. Fu, Y. Ge, and M. Wang, "Collaborative deep metric learning for video understanding," in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2018, pp. 481–490.

J. Lee and S. Abu-El-Haija, "Large-scale content-only video retrieval," in Proc. IEEE Int. Conf. Comput. Vis. Workshops, 2017, pp. 987–995.

J. McAuley, C. Targert, Q. Shi, and A. Van Den Hengel, "Image-based recommendations on styles and substitutes," in Proc. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2015, pp. 43–52.

Y. S. Rawat and M. S. Kankanhalli, "Contagnet: Exploiting user context for image tag recommendation," in Proc. 24th ACM Int. Conf. on Information & Knowledge Management, 2015, pp. 1092–1100.

C. Lei, D. Liu, W. Li, Z.-J. Zha, and H. Li, "Comparative deep learning of hybrid representations for image recommendations," in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., 2016, pp. 2545–2553.

Q. Liu, S. Wu, and L. Wang, "Deepstyle: Learning user preferences for visual recommendation," in Proc. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2017, pp. 841–844.

W. Niu, J. Caverlee, and H. Lu, "Neural personalized ranking for image recommendation," in Proc. Proc. ACM Int. Conf. Web Search Data Mining, 2018, pp. 423–431.

S. Wang, Y. Wang, J. Tang, K. Shu, S. Ranganath, and H. Liu, "What your images reveal: Exploiting visual contents for point-of-interest recommendation," in Proc. Int. Conf. World Wide Web, 2017, pp. 391–400.

T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in Proc. Int. Conf. Learn. Representations, 2017.

Z. Zhang, P. Cui, and W. Zhu, "Deep learning on graphs: A survey," IEEE Trans. Knowl. Data Eng., vol. 34, pp. 249–270, Jan. 2022.

L. Wu, P. Sun, R. Hong, Y. Ge, and M. Wang, "Collaborative neural social recommendation," IEEE Trans. Syst., Man, Cybern. Syst., vol. 51, no. 9, pp. 1106–1108.

L. Wu, L. Chen, R. Hong, Y. Fu, X. Xie, and M. Wang, "A hierarchical attention model for social contextual image recommendation," IEEE Trans. Knowl. Data Eng., vol. 32, no. 10, pp. 1854–1867, Oct. 2020.

L. Wu, P. Sun, Y. Fu, R. Hong, X. Wang, and M. Wang, "A neural influence diffusion model for social recommendation," in Proc. Proc. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2019, pp. 235–244.

C. Chen et al., "An efficient adaptive transfer neural network for social-aware recommendation," in Proc. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2019, pp. 225–234.

Y. Liu, C. Liang, X. He, J. Peng, Z. Zheng, and J. Tang, "Modelling high-order social relations for item recommendation," in Proc. IEEE Trans. Knowl. Data Eng., vol. 32, no. 10, pp. 1102–1106, Aug. 2020.

L. Hu, S. Jian, L. Cao, Z. Gu, Q. Chen, and A. Amirbekyan, "HERS: Modeling influential contexts with heterogeneous relations for sparse and cold-start recommendation," in Proc. AAAI Conf. Artif. Intell., 2019, pp. 3830–3837.

W. Xiao, H. Zhao, H. Pan, Y. Song, V. W. Zheng, and Q. Yang, "Beyond personalization: Social content recommendation for creator equality and consumer satisfaction," in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 235–245.

F. Fan et al., "Graph neural networks for social recommendation," in Proc. Internet Computing, 2019, pp. 417–426.

W. Song, Z. Xiao, Y. Wang, L. Charlin, M. Zhang, and J. Tang, "Session-based social recommendation via dynamic graph attention networks," in Proc. Int. ACM Conf. Web Search Data Mining, 2019, pp. 555–563.

Q. Wu et al., "Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems," in Proc. Int. Conf. World Wide Web, 2019, pp. 2091–2102.

W. Niu, J. Caverlee, and H. Lu, "Neural influence and interest diffusion network for social recommendation," IEEE Trans. Knowl. Data Eng., early access, Dec. 31, 2020, doi: 10.1109/TKDE.2020.3048414.

V. Satuluri et al., "SimClusters: Community-based representations for heterogeneous recommendations at twitter," in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2020, pp. 31–40.

B. Jin, K. Cheng, L. Zhang, Y. Fu, M. Yin, and L. Jiang, "Partial relationship aware influence diffusion via a multi-channel encoding scheme for social recommendation," in Proc. 29th ACM Int. Conf. Inf. Knowl. Manage., 2020, pp. 585–594.

Y. Yan, Z. Fu, S. Muthukrishnan, G. de Melo, and Y. Zhang, "Reinforcement knowledge graph reasoning for explainable recommendation," in Proc. Int. Conf. World Wide Web, 2021, pp. 1106–1110.

Y. S. Rawat and M. S. Kankanhalli, "Contagnet: Exploiting user context for image tag recommendation," in Proc. 24th ACM Int. Conf. on Information & Knowledge Management, 2015, pp. 1092–1100.

Z. Zhang, P. Cui, and W. Zhu, "Deep learning on graphs: A survey," IEEE Trans. Knowl. Data Eng., vol. 34, pp. 249–270, Jan. 2022.

B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based recommendations for heterogeneous information networks," in Proc. Proc. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2020, pp. 239–248.

W. X. Zhang, L. Zhang, L. Zhao, Z. Liu, C. Xing, and X. Xie, "Leveraging demonstrations for reinforcement recommendation reasons over knowledge graphs," in Proc. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2020, pp. 211–216.

W. Liu, Y. Fu, and Y. Ge, "DiffNet++: A graph convolutional network approach," in Proc. Proc. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2019, pp. 2091–2102.

L. Wu, J. Li, P. Sun, R. Hong, Y. Ge, and M. Wang, "DiffNet++: A graph convolutional network approach," in Proc. ACM Int. Conf. Web Search Data Mining, 2019, pp. 2091–2102.

X. Wang, Y. Cao, and T.-S. Chua, "Explainable reasoning over knowledge graphs for recommendation," in Proc. AAAI Conf. Artif. Intell., 2019, pp. 5329–5336.

Y. Cao, X. Wang, X. He, Z. Hu, and T. Chua, "Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences," in Proc. 53rd Int. Conf. World Wide Web, 2019, pp. 151–161.

X. Wang, Y. Cao, M. Liu, and T. Chua, "KGAT: Knowledge graph attention network for recommendation," in Proc. Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 950–958.

K. Zhao, X. Wang, Y. Zhang, L. Zhao, Z. Liu, C. Xing, and X. Xie, "Leveraging demonstrations for reinforcement recommendation reasons over knowledge graphs," in Proc. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2020, pp. 239–248.

X. Wang, X. He, and T.-S. Chua, "Learning and reasoning on graph for recommendation," in Proc. Int. ACM Conf. Web Search Data Mining, 2020, pp. 890–893.

Q. Zhu, X. Zhou, J. Wu, J. Tan, and L. Guo, "A knowledge-aware attentional reasoning network for recommendation," in Proc. AAAI Conf. Artif. Intell., 2020, pp. 6999–7006.

X. Yu et al., "Personalized entity recommendation: a heterogeneous information network approach," in Proc. Int. ACM Conf. Web Search Data Mining, 2014, pp. 283–292.

Z. Sun, J. Yang, J. Zhang, A. Bozzi, L. Huang, and C. Xu, "Recurrent knowledge graph embedding for effective recommendation," in Proc. Proc. ACM Recomender Syst., 2018, pp. 297–305.

H. Zhao, Q. Yao, J. Li, Y. Song, and D. L. Lee, "Meta-graph based recommendation fusion over heterogeneous information networks," in Proc. Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2017, pp. 635–644.

B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based context for top-N recommendation with A neural collaborative model," in Proc. Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2018, pp. 1531–1540.

J. Huang, W. X. Zhao, H. Dou, J.-R. Wen, and E. Y. Chang, "Improving sequential recommendation with knowledge-enhanced memory networks," in Proc. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, 2018, pp. 505–514.

W. Wang, Z. Wang, W. Li, X. Xie, and M. Guo, "Multi-task learning for knowledge graph enhanced recommendation," in Proc. Int. Conf. World Wide Web, 2019, pp. 2000–2010.

W. Wang et al., "Knowledge-aware graph neural networks with label smoothness regularization for recommender systems," in Proc. Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 968–977.
[181] C.-Y. Tai, M.-R. Wu, Y.-W. Chu, S.-Y. Chu, and L.-W. Ku, “MVIN: Learning multiview items for recommendation,” in Proc. ACM SIGIR Conf. Res. Devolop. Inf. Retrieval, 2020, pp. 99–108.

[182] C. Chen, M. Zhang, W. Ma, Y. Liu, and S. Ma, “Jointly non-sampling learning for knowledge graph enhanced recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Devolop. Inf. Retrieval, 2020, pp. 189–198.

[183] R. Sun et al., “Multi-modal knowledge graphs for recommender systems,” in Proc. 29th ACM Int. Conf. Inf. Knowl. Manage., 2020, pp. 1405–1414.

[184] K. Zhou, W. X. Zhao, S. Bian, Y. Zhou, J.-R. Wen, and J. Yu, “Improving conversational recommender systems via knowledge graph based multi-strategy fusion,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2020, pp. 1006–1014.

[185] B. Jin, C. Gao, X. He, D. Jin, and Y. Li, “Multi-behavior recommendation with graph convolutional networks,” in Proc. ACM SIGIR Conf. Res. Devolop. Inf. Retrieval, 2020, pp. 659–668.

[186] L. Hu et al., “Graph neural news recommendation with unsupervised graph embedding,” in Proc. 8th Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 4255–4264.

[187] X. Wang et al., “Learning intents behind interactions with knowledge graph for recommendation,” in Proc. Web Conf., 2021, pp. 878–887.

[188] L. Gao, H. Yang, J. Wu, C. Zhou, W. Lu, and Y. Hu, “Recommendation with multi-source heterogeneous information,” in Proc. Int. Joint Conf. Artif. Intel., 2020.

[189] J. Gong et al., “Attentional graph convolutional networks for knowledge graph concept recommendation in moocs in a heterogeneous view,” in Proc. Int. ACM SIGIR Conf. Res. Devolop. Inf. Retrieval, 2020, pp. 79–88.

[190] Y. Wang, S. Tang, Y. Lei, W. Song, S. Wang, and M. Zhang, “DisenHAN: Disentangled heterogeneous graph attention network for recommendation,” in Proc. Conf. Int. Knowl. Manage., 2020, pp. 1605–1614.

[191] Z. Han et al., “Genetic meta-structure search for recommendation on heterogeneous information network,” in Proc. 29th ACM Int. Conf. Int. Knowl. Manage., 2020, pp. 455–464.

[192] X. Wang, Y. Xu, X. He, Y. Cao, M. Wang, and T.-S. Chua, “Reinforced negative sampling over knowledge graph for recommendation,” in Proc. Int. Conf. World Wide Web, 2020, pp. 99–109.

[193] F. Wang, Y. Fan, L. Xia, W. X. Zhao, S. Liu, and J. Huang, “KERL: A knowledge-guided reinforcement learning model for sequential recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Devolop. Inf. Retrieval, 2020, pp. 209–218.

[194] P. Symeonidis et al., “Recommending the video to watch next: an offline and online evaluation at youtv. de,” in Proc. ACM Conf. Rec. Recommender Syst., 2020, pp. 299–308.

[195] W. Lei et al., “Interactive path reasoning on graph for conversational recommendation,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2020, pp. 2073–2083.

[196] H. Wang et al., “RippleNet: Propagating user preferences on the knowledge graph for recommender systems,” in Proc. 27th ACM Int. Conf. Int. Knowl. Manage., 2018, pp. 417–426.

[197] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, “Learning entity and relation embeddings for knowledge graph completion,” in Proc. AAAI Conf. Artif. Intell., 2015, pp. 2181–2187.

[198] H. Wang, F. Zhang, X. Xie, and M. Guo, “DKN: Deep knowledge-aware network for news recommendation,” in Proc. Int. Conf. World Wide Web, 2018, pp. 1835–1844.

[199] Z. Wang, G. Lin, H. Tan, Q. Chen, and X. Liu, “CKAN: Collaborative knowledge-aware attentive network for recommender systems,” in Proc. Int. ACM SIGIR Conf. Res. Devolop. Inf. Retrieval, 2020, pp. 219–228.

[200] W. Meng, D. Yang, and Y. Xiao, “Incorporating user micro-behaviors and item knowledge into multi-task learning for session-based recommendation,” in Proc. Int. ACM SIGIR Conf. Res. Devolop. Inf. Retrieval, 2020, pp. 1091–1100.

[201] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, “Session-based recommendations with recurrent neural networks,” in Proc. Int. Conf. Learn. Representations, 2016.

[202] B. Hidasi, M. Quadrana, A. Karatzoglou, and D. Tikk, “Parallel recurrent neural network architectures for feature-rich session-based recommendations,” in Proc. ACM Conf. Recommender Syst., 2016, pp. 241–248.
N. Zhu, J. Cao, Y. Liu, Y. Yang, H. Ying, and H. Xiong, “Make it a chorus: knowledge-and time-aware item modeling for nextbasket recommendation,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2019, pp. 1459–1468.

W. Ji, K. Wang, X. Wang, T. Chen, and A. Cristea, “Sequential recommender via time-aware attentive memory network,” in Proc. 29th ACM Int. Conf. Inf. Knowl. Manage., 2020, pp. 1459–1468.

W. Ye, S. Wang, X. Chen, X. Wang, Z. Qin, and D. Yin, “Time matters: Sequential recommendation with complex temporal information,” in Proc. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2020, pp. 109–118.

C. Wang, M. Zhang, W. Ma, Y. Liu, and S. Ma, “Make it a chorus: knowledge-and time-aware item modeling for sequential recommendation,” in Proc. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2020, pp. 109–118.

N. Zhu, J. Cao, Y. Liu, Y. Yang, H. Ying, and H. Xiong, “Sequential modeling of hierarchical user intention and preference for next-item recommendation,” in Proc. ACM Int. Conf. Web Search Data Mining, 2020, pp. 807–815.

M. Wang, P. Ren, L. Mei, Z. Chen, J. Ma, and M. de Rijke, “A collaborative session-based recommendation approach with parallel memory modules,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2019, pp. 345–354.

F. Yuan, A. Karatzoglou, I. Arapakis, J. M. Jose, and X. He, “A session-based generative model for next item recommendation,” in Proc. Int. ACM Conf. Web Search Data Mining, 2020, pp. 582–590.

J. Wang, K. Ding, L. Hong, H. Lui, and J. Caverlee, “Next-item recommendation with sequential hypergraphs,” in Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2020, pp. 1101–1110.
WU ET AL.: SURVEY ON ACCURACY-ORIENTED NEURAL RECOMMENDATION: FROM COLLABORATIVE FILTERING TO...

Le Wu (Member, IEEE) received the PhD degree from the University of Science and Technology of China. She is currently a professor and PhD supervisor with the Hefei University of Technology, China. She has authored or coauthored more than 200 book chapters, journal, and conference papers in his research areas. Her research interests include data mining, recommender systems, and social network analysis. She was the recipient of the Best of SDM 2015 Award, Distinguished Dissertation Award from China Association for Artificial Intelligence 2017, and the Youth Talent Promotion Project from China Association for Science and Technology.

Xiangnan He (Member, IEEE) received the PhD degree in computer science from the National University of Singapore. She is currently a professor with the University of Science and Technology of China. He has authored or coauthored more than 80 publications that appeared in several top conferences, such as SIGIR, WWW, and MM, and journals, including the IEEE Transactions on Knowledge and Data Engineering, ACM Transactions on Information Systems, and IEEE Transactions on Multimedia. His research interests include span information retrieval, data mining, and multi-media analytics. He was the recipient of the Best Paper Award Honorable Mention in WWW 2018 and ACM SIGIR 2016 for his work. He is on the editorial board of journals including Frontiers in Big Data, AI Open etc. Moreover, he was the PC chair of CCIS 2019 and SPC/PC member for several top conferences including SIGIR, WWW, KDD, MM, WSDM, ICML, and etc., and the regular reviewer for journals including the IEEE Transactions on Knowledge and Data Engineering, ACM Transactions on Information Systems, IEEE Transactions on Multimedia, etc.

Xiang Wang (Member, IEEE) received the PhD degree from the National University of Singapore, in 2019. He is currently a research fellow with the National University of Singapore. He has authored or coauthored some academic papers on international conferences, such as KDD, WWW, SIGIR, and AAAI. His research interests include recommender systems, graph learning, and explainable deep learning techniques. He is a program committee member for several top conferences, such as KDD, SIGIR, WWW, and IJCAI, and an invited reviewer for prestigious journals, such as the IEEE Transactions on Knowledge and Data Engineering, ACM Transactions on Information Systems, IEEE Transactions on Neural Networks and Learning Systems, and the IEEE Transactions on Multimedia.

Kun Zhang (Member, IEEE) received the PhD degree in computer science and technology from the University of Science and Technology of China, Hefei, China, in 2019. He is currently a faculty member with the Hefei University of Technology, China. He has authored or coauthored several papers in refereed journals and conferences, such as the IEEE Transactions on Systems, Man, and Cybernetics: Systems, ACM Transactions on Knowledge Discovery from Data, AAAI, KDD, ACL, SIGIR, WWW, and ICDM. His research interests include natural language understanding and recommendation system. He was the recipient of the KDD 2018 Best Student Paper Award.

Meng Wang (Fellow, IEEE) received the BE and PhD degrees from the University of Science and Technology of China, in 2003 and 2008, respectively. He is currently a professor with HFUT. He has authored more than 200 book chapters, journal, and conference papers in his research areas. His current research interests include multimedia content analysis, computer vision, and pattern recognition. He was the recipient of the ACM SIGMM Rising Star Award 2014. He is an associate editor for the IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Circuits and Systems for Video Technology, and IEEE Transactions on Neural Networks and Learning Systems. He is an IAPR fellow.

For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.