A Survey on Representation Learning for User Modeling

Sheng Li[1] and **Handong Zhao**[2]
[1]Department of Computer Science, University of Georgia, GA
[2]Adobe Research, San Jose, CA
sheng.li@uga.edu, hazhao@adobe.com

Abstract

Artificial intelligent systems are changing every aspect of our daily life. In the past decades, numerous approaches have been developed to characterize user behavior, in order to deliver personalized experience to users in scenarios like online shopping or movie recommendation. This paper presents a comprehensive survey of recent advances in user modeling from the perspective of representation learning. In particular, we formulate user modeling as a process of learning latent representations for users. We discuss both the static and sequential representation learning methods for the purpose of user modeling, and review representative approaches in each category, such as matrix factorization, deep collaborative filtering, and recurrent neural networks. Both shallow and deep learning methods are reviewed and discussed. Finally, we conclude this survey and discuss a number of open research problems that would inspire further research in this field.

1 Introduction

The concept of *user modeling* has been mainly discussed in the context of human–computer interaction (HCI), which describes the process of building up and modifying a conceptual understanding of the user. The major task of user modeling is customization and adaptation of systems to the user's specific needs. In recent years, user modeling has been extended to a wide range of domains, such as response prediction in display advertising, conversion prediction in online shopping, movie recommendation in video platforms, and student performance prediction in massive open online courses (MOOC).

User modeling is a very broad topic, and a large number of user modeling approaches have been proposed in the past decades. The existing approaches can be interpreted through various dimensions. In this survey, we analyze the user modeling problem from the perspective of representation learning, and categorize existing work into multiple groups accordingly. We first present the problem statement of user modeling as follows.

Problem Statement. Given a set of users, each user is associated with a number of attributes (either numerical or categorical values). The task of ***user modeling*** is to learn a latent representation for each user, with the help of items, item features and/or user-item response matrix, with applications to response prediction, recommendation, etc.

Representation learning could be used to extract meaningful latent features from either static data (e.g., tabular data) or sequential data (e.g., time-series data). In user modeling applications, user data are often organized in structured static datasets (e.g., user-movie rating matrix) or unstructured sequences (e.g., the purchase history of customers). Moreover, representation learning could be achieved by either shallow models or deep models. On one hand, traditional shallow learning models usually rely on statistical assumptions and learn low-dimensional linear or nonlinear subspaces from high-dimensional data space. On the other hand, deep learning methods employ neural networks with multiple layers to extract latent information from input static/sequential data. Accordingly, existing user modeling approaches can be identified in the intersections of two sets of dimensions, including static/sequential and shallow/deep.

In this survey, we focus on user modeling methods that explicitly consider learning latent representations for users. We will first introduce the static representation learning methods for user modeling, including shallow learning methods like matrix factorization and deep learning methods such as deep collaborative filtering. The advantages and disadvantages of these approaches will be analyzed. We then review the sequential representation learning methods, such as the tensor based methods and recurrent neural networks, in different applications. Unlike static methods, the sequential learning methods consider the evolving of user behavior over time, which have to deal with more challenging issues. Model comparisons and discussions will be provided. Due to the space limit, we cannot go over each category of methods in very detail, but mainly focus on the high-level connections among these categories. For each category, we will choose one or two representative methods and describe their motivations and technical approaches. Finally, we discuss a number of open problems and propose some potential future research directions that would inspire researchers and practitioners in the field.

Existing surveys on relevant topics usually focus on one particular application domain, such as recommender system, but they cannot reveal the interrelationships of various user modeling methods that are widely adopted in multiple domains. To our best knowledge, our paper is the first survey that focuses on the general user modeling task from the unique perspective of representation learning.

2 Static Representation Learning

The importance of accurate user modeling techniques motivated by wide-ranging applications has fueled a great amount of academic as well as industrial research in this area. In static user modeling, representation learning methods aim to extract latent user representations from static data, such as user-item interaction matrix or user-advertisement response matrix. Temporal information is not available (or not considered) in this setting. Static user modeling is a simplified but very commonly used problem setting in literature.

2.1 Factorization Methods

The most representative approach for static representation learning might be matrix factorization, which has been recognized as an effective solution to *collaborative filtering* and applied to many recommender systems. Matrix factorization (MF) is a type of *latent factor models*, which leads to the best performance at the Netflix competition in 2007 [Koren *et al.*, 2009]. There are mainly two types of factorization methods, which require different inputs for model training.

The **first category** of factorization methods only requires a user response matrix (i.e., interactions between users and items) as input, and learn latent features for users (and items). Let $X \in \mathbb{R}^{N \times M}$ denote a user response matrix, from N users to M items. The items shall be specified in different applications. For instance, the items correspond to movies in movie recommendation and correspond to ads in display advertising. Each row of X corresponds to a user, and each column corresponds to an item. The element X_{ij} denotes the response from the i-th user to the j-th item, such as movie rating. X could be very sparse, as we can only observe responses from users to a subset of items. Usually, $X_{ij} \in [0,\ C]$, where C is the maximum value in the response (e.g., highest movie rating). Matrix factorization based methods [Koren *et al.*, 2009] usually decompose the response matrix X into two latent factor matrices U and V, i.e., $X \approx UV$, $U \in \mathbb{R}^{N \times K}$, $V \in R^{K \times M}$, where K is the number of latent factors and it's usually much smaller than N and M.

The objective function of standard matrix factorization is:

$$\begin{aligned}\arg\min_{U,V} f(U,V) = \|I \odot (X - UV^{\mathrm{T}})\|_{\mathrm{F}}^2 \\ + \lambda(\|U\|_{\mathrm{F}}^2 + \|V\|_{\mathrm{F}}^2),\end{aligned} \tag{1}$$

where I is an indicator matrix with $I_{ij} = 1$ if X_{ij} is a valid value and 0 otherwise, $\odot$ denotes the Hadamard product (i.e., element-wise product), and λ is a trade-off parameter. The first term in (1) denotes the approximation errors, and the last two terms are regularizations used to prevent overfitting. The probabilistic explanation of the standard matrix factorization model is provided in [Mnih and Salakhutdinov, 2008].

Although promising, the conventional matrix factorization methods suffer from the problem of cold-start, i.e. what recommendations to make when a new user/item arrives in the system. Another problem often presented in many real world applications is data sparsity or reduced coverage.

The **second category** of factorization methods tries to address the aforementioned issues in conventional matrix factorization by incorporating side information to learn latent factors. Some methods have shown promising performance in collaborative filtering by following such a strategy. Porteous *et al.* proposed a Bayesian matrix factorization (BMF) approach with side information and Dirichlet process mixtures [Porteous *et al.*, 2010]. Hu *et al.* proposed a cross-domain triadic factorization (CDTF) method [Hu *et al.*, 2013], which leverages the information from other domains. Factorization machine (FM) is another representative model in this category, which has been widely applied in practice [Rendle, 2012]. FM can be considered as a general framework of matrix factorization, which has achieved remarkable performance in various regression and classification tasks and has been deployed in many industrial systems. It provides a general formulation to explicitly model the pairwise interactions of user features and item features. The predictive model of FM is defined as:

$$f(\mathbf{w}, \mathbf{G}|\mathbf{x}) = w_0 + \sum_{i=1}^{d} dw_i x_i + \sum_{i=1}^{d} \sum_{j=i+1}^{d} \mathbf{g}_i^{\top} \mathbf{g}_j x_i x_j, \tag{2}$$

where $\mathbf{w}$ and $\mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_d]$ are model parameters. $\mathbf{x}$ contains user features, item features, and other available side information. $\mathbf{g}_i \in \mathbb{R}^m$ denotes the m-dimensional embedding of the i-th feature in $\mathbf{x}$. Although FM does not directly generate embedding vectors for users, the feature embeddings $\mathbf{g}_i$ can be converted to user embedding and employed for user modeling purposes.

A comprehensive discussion of matrix factorization based user modeling approaches can be found in [Koren and Bell, 2015]. Beyond recommender systems, the matrix factorization approach has been applied to other areas such as educational assessment, with proper incorporation of domain knowledge. Recently, Jing *et al.* proposed a contextual collaborative filtering (CCF) approach for student response prediction in mixed-format tests [Jing and Li, 2018].

Remarks. Factorization methods are simple yet effective, which are easy to be implemented and deployed. Methods like standard matrix factorization and factorization machines have been widely employed in many commercial systems. However, these methods usually have strong statistical assumptions on data distribution, and have limited capability in modeling complex or dynamic user behavior data.

2.2 Deep Neural Networks

Developing customized deep neural networks for the task of user modeling has been an emerging topic in the past decade. Similar to factorization methods introduced before, most existing deep neural network approaches usually focus on the problem of collaborative filtering, in which the user modeling problem is defined as learning hidden representations for users. In the following, we present three categories of deep neural network methods for user modeling, including the restricted Boltzmann machine based methods, auto-encoder based methods, and feed-forward network based methods.

The **first category** is restricted Boltzmann machine (RBM) based methods. Salakhutdinov *et al.* [Salakhutdinov *et al.*, 2007] might be the first to apply RBM to the task of collaborative filtering. They modified the restricted Boltzmann machines as a two-layer undirected graphical model consisting of binary hidden units and softmax visible units for collaborative filtering. They further designed an efficient learning procedure called contrastive divergence to maximize an approximation to the true likelihood function. A conditional RBM model and inference procedures are also proposed. They demonstrated the model performance on the Netflix dataset for movie recommendation. Following this line, Truyen *et al.* [Truyen *et al.*, 2009] proposed ordinal Boltzmann machines for collaborative filtering. They studied the parameterizations for handling the ordinal nature of ratings, and presented the integration of multiple Boltzmann machines for user-based and item-based processes. Wang *et al.* [Wang and Wang, 2014] utilized deep belief nets (DBN) for music recommendation, which unifies feature extraction and recommendation of songs in a joint framework. They assumed that a user has a feature vector β_u drawn from a Gaussian prior and each songs has a feature vector x_v. They automatically learned the feature vectors of the songs using a deep belief network which is a generative probabilistic graphical model with hidden nodes and observation. The authors used stacked layers of Restricted Boltzmann Machines for pretraining in an unsupervised fashion, and then employed the Maximum Likelihood Estimation (MLE) for supervised learning.

The **second category** is auto-encoder based methods. These methods mainly aim to learn latent factors from content information such as raw features of audio or documents. Wang *et al.* [Wang *et al.*, 2015] proposed a hierarchical Bayesian model called collaborative deep learning (CDL) which tightly couples stacked denoising auto-encoders (SDA) and collaborative topic regression (CTR). Li *et al.* proposed a general deep collaborative filtering (DCF) framework, which unifies the deep learning models with matrix factorization based collaborative filtering [Li *et al.*, 2015a]. DCF is a *hybrid* model, which makes use of both user response matrix and side information and bridges together matrix factorization and deep auto-encoders. Given a user-item rating matrix R, the user side information X and the item side information Y, DCF jointly decomposes R and learns latent factor matrices (i.e., U, V) from ratings and side information (i.e., X and Y) through the following formulation:

$$\begin{aligned}\arg\min_{U,V} \quad & l(R,U,V) + \beta(\|U\|_{\mathrm{F}}^2 + \|V\|_{\mathrm{F}}^2) \\ & + \gamma\mathcal{L}(X,U) + \delta\mathcal{L}(Y,V),\end{aligned} \tag{3}$$

where β, γ and δ are trade-off parameters. There are two key components in the DCF framework: (1) the function $l(R,U,V)$ for decomposing the rating matrix R into the two latent matrices; (2) the functions $\mathcal{L}(X,U)$ and $\mathcal{L}(Y,V)$ that connect the user/item contextual features with the latent factors. The first component derived through matrix factorization extracts latent knowledge from the rating matrix. The second component devised using deep learning models establishes connections of the side information with the latent factors. In particular, Li *et al.* presented an instantiation of DCF by combining probabilistic matrix factorization (PMF) with marginalized denoising auto-encoders (mDA) [Li *et al.*, 2015a]. PMF is a widely applied CF approach with excellent performance, and mDA is a powerful tool in extracting high-level features from raw inputs. Empirical evaluations on movie recommendation and book recommendation datasets show that the combination of the two leverages their benefits for learning even richer models.

The **third category** is feed-forward neural network methods. Oord *et al.* [Oord *et al.*, 2013] addressed the music recommendation problem using the convolutional neural networks. They first conducted a weighted matrix factorization to handle implicit feedback and obtained latent factors for all songs. After that, they used deep neural networks to map audio content to those latent factors. In particular, they extracted local features from audio signals and aggregated them into a bag-of-words representation. Finally, the deep convolutional network was employed to map this feature representation to the latent factors. They tested their algorithm on the Million song dataset and showed that their model improved the recommendation performance by augmenting the audio signals. He *et al.* proposed a neural collaborative filtering (NCF) approach [He *et al.*, 2017], which replaces the inner product in collaborative filtering with a neural network architecture that can learn an arbitrary function from data. The predictive model of NCF is formulated as:

$$\hat{R}_{ij} = f(U^\top \mathbf{x}_i, V^\top \mathbf{y}_j | U, V, \Theta), \tag{4}$$

where $\hat{R}_{ij}$ denotes the response from the i-th user to the j-th item, $\mathbf{x}_i$ is the feature vector for the i-th user, $\mathbf{y}_j$ is the feature vector for the j-th item, and Θ denotes the model parameters of function f. U and V denote the embeddings of users and items, respectively. The function $f(\cdot)$ can be implemented as a multi-layer feed-forward neural network. The NCF framework also allows the combination of generalized matrix factorization and multi-layer perceptron.

In [He and Chua, 2017], a neural factorization machine (NFM) is proposed for sparse predictive analytics. NFM extends the idea of factorization machines (FM) to the setting of deep neural networks, which models both second-order and higher-order feature interactions. Zhang *et al.* presented a neural user-item coupling learning for collaborative filtering [Zhang *et al.*, 2018].

Deep learning based static user modeling, especially the deep collaborative filtering approach, has been rapidly developed in recent years. Readers could refer to an excellent survey on this topic for more information [Zhang *et al.*, 2019].

Remarks. Compared with factorization methods, the deep neural networks based methods have a better capability in terms of discovering complex feature interactions, but they still face some challenging issues. First, deep user modeling approaches have to deal with user data which are often discrete or categorical values. How to learn effective embeddings for categorical features is still an open problem. Second, it is usually difficult to explain the internal mechanisms of current deep user modeling methods. Designing explainable deep learning methods for user modeling would further promote the applications of user modeling techniques.

3 Sequential Representation Learning

The static user modeling discussed in Section 2 makes a simple assumption that user data and user preference are static observations. In reality, however, users have dynamic behaviors. A key observation is that *the behavior and preference of users may change over time*. For example, in the scenario of online shopping, if a user has already purchased an item in the previous week, it is unlikely that he/she will purchase the same item again in the next week. Sequential user modeling approaches have been developed to capture the dynamic behavior of users and learn robust user representations. In the following, we will discuss dynamic factorization methods and deep sequential learning methods.

3.1 Dynamic Factorization Methods

Factorization methods discussed in Section 2.1 can be extended to dynamic settings in two ways, including dynamic matrix factorization and tensor decomposition. The former one focuses on factorizing user-item interaction matrix at each time step, but incorporates temporal constraints. The latter one directly represents sequential response data as a high-order tensor, and factorizes tensors to obtain latent embeddings for users and items.

Dynamic Matrix Factorization

User modeling techniques considering temporal dynamics was first developed using collaborative filtering in [Koren, 2009]. Koenigstein *et al.* utilized the temporal information for music recommendation [Koenigstein *et al.*, 2011]. In [Chua *et al.*, 2013], a dynamic matrix factorization method was presented to model the temporal adoption effects in collaborative filtering. Devooght et al. introduced a novel dynamic matrix factorization framework for collaborative filtering, which allows to set an explicit prior to unknown values in rating matrix [Devooght *et al.*, 2015]. In addition to the consideration of temporal dynamics, it has been proven that modeling several related tasks jointly will further boost the model performance. In [Li *et al.*, 2015b; Li and Fu, 2017], a dynamic collective matrix factorization (DCMF) approach is proposed to deal with two user modeling tasks jointly, including click prediction and purchase prediction. It extends the collective matrix factorization formulation, and focuses on the application of user behavior prediction in digital marketing.

We briefly describe the idea of DCMF as an example of dynamic matrix factorization methods for user modeling. Given T pre-defined time slices $t \in \{1, 2, \cdots, T\}$, C^t and D^t are used to denote the click responses and purchase activities in the time slice t, respectively. Since the temporal dynamics are ignored in standard matrix factorization, it can only utilize all the previous data (i.e., $\{C^1, \cdots, C^t\}$ and $\{D^1, \cdots, D^t\}$) for training, and then predict D^{t+1}. As a result, the size of training data will increase significantly over time, which leads to a heavy computational burden. By exploiting the temporal relationships between click response and purchase events, the purchase events in time $t+1$ are mainly related to the click events in time t and hence the dynamic model needs to account for that. Furthermore, DCMF ensures that the latent features of users do not dramatically change in a short period of time, as in reality the user preferences would evolve smoothly. To address this concern, DCMF leverages the latent features of the users learned in time $t-1$ (i.e., U^{t-1}) at time t ($t > 1$). Specifically, a regularization term, $U^t \approx U^{t-1}M$ is designed to account for the drift in user preferences, where U^{t-1} is the latent features of users learned from the previous time slice $t-1$. M is a transition matrix of users' behavior, which tries to capture the mappings between users' behavior in two successive time slices. The intuition is that users' intention on purchasing items should be smoothly transited over time. Then, the objective function of DCMF is:

$$\begin{aligned}\min_{U^t, V^t, P^t, M} f(U^t, V^t, P^t, M) = \ & \alpha\|W^C \odot (C^t - U^t V^{t\mathrm{T}})\|_\mathrm{F}^2 \\ & + (1-\alpha)\|W^D \odot (D^t - U^t P^{t\mathrm{T}})\|_\mathrm{F}^2 \\ & + \lambda_1 \|U^t - U^{t-1}M\|_\mathrm{F}^2 \\ & + \lambda_2 (\|U^t\|_\mathrm{F}^2 + \|V^t\|_\mathrm{F}^2 + \|P^t\|_\mathrm{F}^2 + \|M\|_\mathrm{F}^2),\end{aligned} \tag{5}$$

where the latent features U^{t-1} is given, λ_1 and λ_2 are tradeoff parameters. The last regularization term $\|M\|_\mathrm{F}^2$ is used to control the complexity of model.

Tensor Factorization

Temporal user response data can be represented as a high-order tensor, by considering user, item, timestamps and other available factors as multiple dimensions. In this way, tensor factorization methods can be used to learn user embeddings. In [Karatzoglou *et al.*, 2010], a multiverse recommendation method is proposed, which considers different types of context in the tensor representation, and high-order singular value decomposition (SVD) is performed to get latent factors. Furthermore, Bhargava *et al.* proposed a multi-dimensional collaborative recommendation approach for Who (User), What(Activity), When (Time) and Where (Location), by using tensor factorization on sparse user-generated data [Bhargava *et al.*, 2015]. These approaches have compact and unified formulations, but also require high computational costs due to SVD operations.

Remarks. Dynamic factorization methods present practical solutions to modeling sequential user behavior data, and they demonstrate better flexibility than static user modeling approaches. However, they require pre-defined constraints to incorporate temporal smoothness constraints, and have limited capability in modeling long-term temporal dependencies in sequential data.

3.2 Deep Sequential Learning

In recent years, deep sequential learning methods based on recurrent neural networks have been successfully applied to a wide range of sequential data, such as speech and videos. For user modeling, a number of recurrent neural networks based approaches have been proposed to learn latent user embeddings from structured or unstructured user behavior data. Most of existing methods focus on modeling user behavior in a single domain, while some recent methods present a new setting of user modeling from multiple domains.

Single Domain

Recurrent neural networks (RNN) have been introduced to user modeling recently. For modeling the dynamic behavior of users, an RNN-based recommender system was firstly introduced for session-based recommendation [Hidasi *et al.*, 2015]. Based on this model, a number of RNN-based dynamic methods have been developed by additionally considering personalization and context-awareness. Donkers *et al.* [Donkers *et al.*, 2017] devised a user-based Gated Recurrent Units (GRU) method that attentively considers user embeddings along with sequential item information for personalized next item recommendations. Quadrana *et al.* [Quadrana *et al.*, 2017] hierarchically exploited a user-level RNN and a session-level RNN to reflect users' inter-session sequential dynamics into intra-session sequential dynamics for personalized session-based recommendations. Wu *et al.* [Wu *et al.*, 2017] exploited a user-level and an item-level RNNs in parallel based on the user-level and the item-level history, respectively, to consider temporal evaluation of users and items for the rating prediction.

To consider the context-aware information, Beutel et al. [Beutel *et al.*, 2018] reflected contextual information into input, output and RNN layers. Particularly, they parametrized hidden state transitions in RNNs with an element-wise multiplicative function of context embeddings for better next item recommendations and compared their approach with baselines without contexts. Zhu *et al.* [Zhu *et al.*, 2017] claimed the importance of taking into account time intervals in order to effectively capture the relations of user behaviors, and thus, they devised a new LSTM variant to equip LSTM [Hochreiter and Schmidhuber, 1997] with newly introduced time gates to model time intervals between two successive user behaviors for the next-basket recommendations.

The above methods perform user modeling by exploiting structured sequential data. However, many real-world applications involve unstructured sequential data, such as software log-trace data or web service log data. Tao *et al.* [Tao *et al.*, 2019] proposed a deep sequential framework (Log2Intent) to model the software user action preference using user session log-trace data. Specifically, a sequence-to-sequence-like model is proposed by treating each session as a sequential node. It assumes that the temporally encoded user action is capable of predicting the next actions in decoder. The gated recurrent unit (GRU) is used as the sequence encoder and decoder. In Log2Intent, the authors further explored an auxiliary natural language dataset to improve the interpretability of the learned user action embedding. To this end, the authors proposed a recurrent semantics memory unit (RSMU) to dynamically fetch the memory slot (i.e., sentence) from each action's memories as key-value dictionary. Once the most relevant sentences are retrieved, their embeddings are fed into the sequence encoder together with the log-trace information [Tao *et al.*, 2019].

Multiple Domains

Existing approaches on dynamic user modeling mainly fous on user behavior in a single domain. However, in reality, users may switch across multiple domains (e.g., different shopping websites, different mobile devices). Elkahky *et al.* proposed a single multi-view deep neural network model that jointly learns dense features of items from different domains such as News, Apps and Movie/TV via common users [Elkahky *et al.*, 2015].

In [Kim *et al.*, 2019], a domain switch-aware holistic recurrent neural network (DS-HRNN) is proposed for modeling multi-domain user behavior. DS-HRNN specifically addresses the *domain switch* challenge, by utilizing a domain switch-aware supplementary loss and domain switch-aware behavior regularizer. In DS-HRNN, the key ideas of handling multi-domain user behavior include: (1) aggregating user behaviors from multiple domains into one sequence in chronological order; and (2) exploiting a single RNN model that takes sequences of multi-domain user behaviors. To alleviate the disconnection of local dynamics, it is necessary to recover the disconnection as much as possible while preserving global dynamics of sequential behaviors simultaneously. To this end, a *domain switch-aware supplementary loss* is incorporated to DS-HRNN, which is an explicit way of recovering the lost connection at domain switches Similarly, alleviating the disconnection of local dynamics can be also achieved in an implicit way in terms of inputs. It leads to domain switch-aware behavior regularizer that minimizes the distance between correlated inputs from different domains.

Remarks. Deep sequential learning for user modeling has attracted increasing attention, owing to its remarkable performance in practice. Unlike other sequential data such as texts or videos, the temporal patterns in user behavior data usually have larger variations and higher heterogeneity. Designing customized deep sequential learning architectures for user modeling purpose would be a critical task in the future.

4 Conclusion and Future Directions

In this survey, we review the recent advances in user modeling from the perspective of static and sequential representation learning. Both shallow and deep representation learning methods are discussed. We consider the key problem in user modeling as the process of learning low-dimensional linear or nonlinear representations for users, but in the contexts of static or sequential, single-domain or multi-domain, etc. Due to the space limit, this survey cannot cover some other relevant topics, such as user modeling based on textual data [Zheng *et al.*, 2017; Chen *et al.*, 2018] and knowledge graph based user modeling [Wang *et al.*, 2019].

In the following, we present a few potential topics in user modeling, which are less explored so far. We hope this survey and the discussions on emerging topics could not only provide researchers and practitioners in related fields a comprehensive understanding of recent user modeling approaches, but also shed some light on designing novel and practical representation learning methods for user modeling.

- **Modeling Categorical User Features**. Existing computational methodologies, including both shallow and deep models, work well for feature vectors with numerical values, but could not well model the categorical features. One-hot encoding is often considered as a proper way to represent categorical features, but it cannot effectively handle categorical features with high cardinality. Designing novel neural network architectures for modeling categorical features is an open problem.

- **Knowledge Transfer for User Modeling**. To characterize the complex behavior of users, it is usually insufficient by leveraging only one dataset. Due to the commonality of human behavior in many domains, it might be possible to transfer knowledge from auxiliary data to boost the learning performance in target domains. Two problem settings might be considered. First, two user modeling tasks from different domains may help each other. Second, one user modeling task might benefit from external knowledge bases, such as knowledge graphs, text corpus, etc. Leveraging methodology and theory in domain adaptation [Li *et al.*, 2017; Ding *et al.*, 2018; Shao *et al.*, 2016] could be helpful.
- **Algorithmically Fair User Modeling**. Fairness-aware representation learning has garnered extensive attention in machine learning field and beyond. Despite a few successes in recommendation task [Yao and Huang, 2017], this line of research to prevent unfair predictions for users from minority groups is still underexplored. The open research problems include how to define new notions of fairness and how to design fairness-aware user embedding methods.
- **Explainable User Modeling**. Although latent representations of user modeling lead to remarkable results in terms of response prediction or recommendation, it is usually hard to explain why these models work so well. Adding explainability to user modeling would make such techniques more acceptable by practitioners. A potential direction is based on nonnegative matrix factorization (NMF) [Lee and Seung, 1999]. NMF and its variants show promising results of interpretable representation learning in a number of applications, including face clustering [Zhao *et al.*, 2017], topic modeling [Shi *et al.*, 2018], hand-written digit classification [Zhao *et al.*, 2015], etc. Research efforts on novel principles of explainable methodology are expected.
- **Multi-Level User Modeling**. Existing user modeling approaches consider users equally. However, hierarchical structures could be observed among a large set of users, considering the spatial or interest-driven grouping patterns of human beings. Discovering such multi-level patterns and leveraging them for user modeling would be an interesting topic.
- **Deep Integration of Intelligent Assistants**. By far, user modeling has been mainly discussed in the context of recommendation, advertising, etc. It will be meaningful if pushing user modelings techniques to other relevant domains that also involve dynamic modeling of user behaviors. One example is product question answering [Lai *et al.*, 2018], which aims to provide customers the relevant information regarding queries. User modeling might be useful in generating personalized responses or capturing dynamic changes of interests.

References

[Beutel *et al.*, 2018] Alex Beutel, Paul Covington, Sagar Jain, Can Xu, Jia Li, Vince Gatto, and Ed H. Chi. Latent cross: Making use of context in recurrent recommender systems. In *WSDM*, pages 46–54, 2018.

[Bhargava *et al.*, 2015] Preeti Bhargava, Thomas Phan, Jiayu Zhou, and Juhan Lee. Who, what, when, and where: Multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data. In *Proceedings of the 24th international conference on world wide web*, pages 130–140, 2015.

[Chen *et al.*, 2018] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 World Wide Web Conference*, pages 1583–1592, 2018.

[Chua *et al.*, 2013] Freddy Chong Tat Chua, Richard Jayadi Oentaryo, and Ee-Peng Lim. Modeling temporal adoptions using dynamic matrix factorization. In *Proceedings of the 13th IEEE International Conference on Data Mining*, pages 91–100, 2013.

[Devooght *et al.*, 2015] Robin Devooght, Nicolas Kourtellis, and Amin Mantrach. Dynamic matrix factorization with priors on unknown values. In *KDD*, pages 189–198. ACM, 2015.

[Ding *et al.*, 2018] Zhengming Ding, Sheng Li, Ming Shao, and Yun Fu. Graph adaptive knowledge transfer for unsupervised domain adaptation. In *ECCV*, pages 37–52, 2018.

[Donkers *et al.*, 2017] Tim Donkers, Benedikt Loepp, and Jürgen Ziegler. Sequential user-based recurrent neural network recommendations. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, RecSys, pages 152–160, 2017.

[Elkahky *et al.*, 2015] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *WWW*, pages 278–288, 2015.

[He and Chua, 2017] Xiangnan He and Tat-Seng Chua. Neural factorization machines for sparse predictive analytics. In *Proceedings of International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 355–364, 2017.

[He *et al.*, 2017] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 173–182. International World Wide Web Conferences Steering Committee, 2017.

[Hidasi *et al.*, 2015] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[Hu *et al.*, 2013] Liang Hu, Jian Cao, Guandong Xu, Longbing Cao, Zhiping Gu, and Can Zhu. Personalized recommendation via cross-domain triadic factorization. In *WWW*, pages 595–606, 2013.

[Jing and Li, 2018] Shumin Jing and Sheng Li. Contextual collaborative filtering for student response prediction in mixed-format tests. In *AAAI Conference on Artificial Intelligence*, 2018.

[Karatzoglou *et al.*, 2010] Alexandros Karatzoglou, Xavier Amatriain, Linas Baltrunas, and Nuria Oliver. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 79–86, 2010.

[Kim *et al.*, 2019] Donghyun Kim, Sungchul Kim, Handong Zhao, Sheng Li, Ryan A Rossi, and Eunyee Koh. Domain switch-aware holistic recurrent neural network for modeling multi-domain user behavior. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*, pages 663–671, 2019.

[Koenigstein *et al.*, 2011] Noam Koenigstein, Gideon Dror, and Yehuda Koren. Yahoo! music recommendations: modeling music ratings with temporal dynamics and item taxonomy. In *Proceedings of the 5th ACM conference on Recommender systems*, pages 165–172, 2011.

[Koren and Bell, 2015] Yehuda Koren and Robert Bell. Advances in collaborative filtering. In *Recommender Systems Handbook*, pages 77–118. Springer, 2015.

[Koren *et al.*, 2009] Yehuda Koren, Robert M. Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[Koren, 2009] Yehuda Koren. Collaborative filtering with temporal dynamics. In *KDD*, pages 447–456, 2009.

[Lai *et al.*, 2018] Tuan Manh Lai, Trung Bui, and Sheng Li. A review on deep learning techniques applied to answer selection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2132–2144, 2018.

[Lee and Seung, 1999] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by nonnegative matrix factorization. *Nature*, 401:788–791, 1999.

[Li and Fu, 2017] Sheng Li and Yun Fu. Robust representations for response prediction. In *Robust Representation for Data Analytics*, pages 147–174. Springer, 2017.

[Li *et al.*, 2015a] Sheng Li, Jaya Kawale, and Yun Fu. Deep collaborative filtering via marginalized denoising auto-encoder. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 811–820. ACM, 2015.

[Li *et al.*, 2015b] Sheng Li, Jaya Kawale, and Yun Fu. Predicting user behavior in display advertising via dynamic collective matrix factorization. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 875–878. ACM, 2015.

[Li *et al.*, 2017] Sheng Li, Kang Li, and Yun Fu. Self-taught low-rank coding for visual learning. *IEEE Transactions on Neural Networks and Learning Systems*, 29(3):645–656, 2017.

[Mnih and Salakhutdinov, 2008] Andriy Mnih and Ruslan R Salakhutdinov. Probabilistic matrix factorization. In *NIPS*, pages 1257–1264, 2008.

[Oord *et al.*, 2013] Aäron Van Den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. In *NIPS*, pages 2643–2651, 2013.

[Porteous *et al.*, 2010] Ian Porteous, Arthur U. Asuncion, and Max Welling. Bayesian matrix factorization with side information and dirichlet process mixtures. In *AAAI*, 2010.

[Quadrana *et al.*, 2017] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, RecSys '17, pages 130–137, 2017.

[Rendle, 2012] Steffen Rendle. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(3):57, 2012.

[Salakhutdinov *et al.*, 2007] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey E. Hinton. Restricted boltzmann machines for collaborative filtering. In *ICML*, pages 791–798, 2007.

[Shao *et al.*, 2016] Ming Shao, Zhengming Ding, Handong Zhao, and Yun Fu. Spectral bisection tree guided deep adaptive exemplar autoencoder for unsupervised domain adaptation. In *AAAI*, pages 2023–2029, 2016.

[Shi *et al.*, 2018] Tian Shi, Kyeongpil Kang, Jaegul Choo, and Chandan K. Reddy. Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *WWW*, pages 1105–1114, 2018.

[Tao *et al.*, 2019] Zhiqiang Tao, Sheng Li, Zhaowen Wang, Chen Fang, Longqi Yang, Handong Zhao, and Yun Fu. Log2intent: Towards interpretable user modeling via recurrent semantics memory unit. In *KDD*, pages 1055–1063, 2019.

[Truyen *et al.*, 2009] Tran The Truyen, Dinh Q. Phung, and Svetha Venkatesh. Ordinal boltzmann machines for collaborative filtering. In *UAI*, pages 548–556, 2009.

[Wang and Wang, 2014] Xinxi Wang and Ye Wang. Improving content-based and hybrid music recommendation using deep learning. In *ACM MM*, pages 627–636, 2014.

[Wang *et al.*, 2015] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *KDD*, pages 1235–1244. ACM, 2015.

[Wang *et al.*, 2019] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Exploring high-order user preference on the knowledge graph for recommender systems. *ACM Transactions on Information Systems*, 37(3):1–26, 2019.

[Wu *et al.*, 2017] Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J. Smola, and How Jing. Recurrent recommender networks. In *WSDM*, pages 495–503, 2017.

[Yao and Huang, 2017] Sirui Yao and Bert Huang. Beyond parity: Fairness objectives for collaborative filtering. In *NIPS*, pages 2921–2930, 2017.

[Zhang *et al.*, 2018] Quangui Zhang, Longbing Cao, Chengzhang Zhu, Zhiqiang Li, and Jinguang Sun. Coupledcf: Learning explicit and implicit user-item couplings in recommendation for deep collaborative filtering. In *IJCAI*, pages 3662–3668, 2018.

[Zhang *et al.*, 2019] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)*, 52(1):5, 2019.

[Zhao *et al.*, 2015] Handong Zhao, Zhengming Ding, Ming Shao, and Yun Fu. Part-level regularized semi-nonnegative coding for semi-supervised learning. In *ICDM*, pages 1123–1128, 2015.

[Zhao *et al.*, 2017] Handong Zhao, Zhengming Ding, and Yun Fu. Multi-view clustering via deep matrix factorization. In *AAAI*, pages 2921–2927, 2017.

[Zheng *et al.*, 2017] Lei Zheng, Vahid Noroozi, and Philip S Yu. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 425–434, 2017.

[Zhu *et al.*, 2017] Yu Zhu, Hao Li, Yikang Liao, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. What to do next: Modeling user behaviors by time-lstm. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI, pages 3602–3608, 2017.