Explainable Interaction-driven User Modeling over Knowledge Graph for Sequential Recommendation

Xiaowen Huang[1,2], Quan Fang[1,2], Shengsheng Qian[1,2], Jitao Sang[3,5], Yan Li[4], Changsheng Xu[1,2,5]
[1]National Lab of Pattern Recognition, Institute of Automation, CAS, Beijing 100190, China
[2]University of Chinese Academy of Sciences
[3]School of Computer and Information Technology & Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University
[4]Kuaishou Technology, Beijing, China
[5]Peng Cheng Laboratory, ShenZhen, China
xiaowen.huang77@gmail.com,{shengsheng.qian,qfang}@nlpr.ia.ac.cn,jtsang@bjtu.edu.cn,liyan@kuaishou.com,csxu@nlpr.ia.ac.cn

ABSTRACT

Compared with the traditional recommendation system, sequential recommendation holds the ability of capturing the evolution of users' dynamic interests. Many previous studies in sequential recommendation focus on the accuracy of predicting the next item that a user might interact with, while generally ignore providing explanations why the item is recommended to the user. Appropriate explanations are critical to help users adopt the recommended item, and thus improve the transparency and trustworthiness of the recommendation system. In this paper, we propose a novel Explainable Interaction-driven User Modeling (EIUM) algorithm to exploit Knowledge Graph (KG) for constructing an effective and explainable sequential recommender. Qualified semantic paths between specific user-item pair are extracted from KG. Encoding those semantic paths and learning the importance scores for each path provides the path-wise explanation for the recommendation system. Different from traditional item-level sequential modeling methods, we capture the interaction-level user dynamic preferences by modeling the sequential interactions. It is a high-level representation which contains auxiliary semantic information from KG. Furthermore, we adopt a joint learning manner for better representation learning by employing multi-modal fusion, which benefits from the structural constraints in KG and involves three kinds of modalities. Extensive experiments on the large-scale dataset show the better performance of our approach in making sequential recommendations in terms of both accuracy and explainability.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

explainable recommendation, user modeling, knowledge graph, sequential recommendation

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM '19, October 21–25, 2019, Nice, France
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6889-6/19/10…$15.00
https://doi.org/10.1145/3343031.3350893

ACM Reference Format:
Xiaowen Huang, Quan Fang, Shengsheng Qian, Jitao Sang, Yan Li, Changsheng Xu. 2019. Explainable Interaction-driven User Modeling over Knowledge Graph for Sequential Recommendation. In *Proceedings of the 27th ACM International Conference on Multimedia (MM'19), October 21-25, 2019, Nice, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3343031.3350893

1 INTRODUCTION

Sequential Recommendation (SR) aims to meet the current needs of user according to her/his historical behavior sequence [17]. Since the interest of the user is evolving dynamically, how to *capture the user's **dynamic** interests **accurately** and **explainably*** is the focus of our work. It is equally important to offer a credible explanation for the user as opposed to understand the user's dynamic preference drifting and provide personalized recommendation, so that the user can clearly understand why the recommendation system recommends specific items to her/him.

Some previous work has made a great contribution to the SR task. The Markov Chain (MC) model and its variants, such as FPMC [17] combining the power of MC and Matrix Factorization (MF), are the very early approaches to model the personalized sequential behaviors. With the revival of neural networks (NN), many deep learning models are used to model sequential relationships. Recurrent Neural Network (RNN) is a classical algorithm in SR task, which is able to capture temporal dependencies by encoding user's historical behaviors into a latent vector. However, RNN-based approach is inefficient to model long-dependencies and cannot operate in a parallel way due to its special architecture. Recently, self-attention based sequential recommendation algorithms have attracted increasing attention due to the flexibility and efficiency of the model [12, 33]. Those NN-based methods achieve high accuracy in recommendation task. However, most of those methods do not consider providing users with credible explanations while recommending.

The Knowledge Gragh (KG), which contains comprehensive auxiliary data referring to the facts and connections about items, has gained a lot of attentions in top-N recommendation task [21, 23, 25] and SR task[11]. The entities exist in KG in the form of triples, which can be seamlessly integrated with user-item interactions, endowing recommendation systems the ablility of explainablity [25]. There are usually two forms of incorporating KG into recommender, one is knowledge graph embedding (KGE) based methods, the other is path-based methods. The KGE based methods learn the repre-

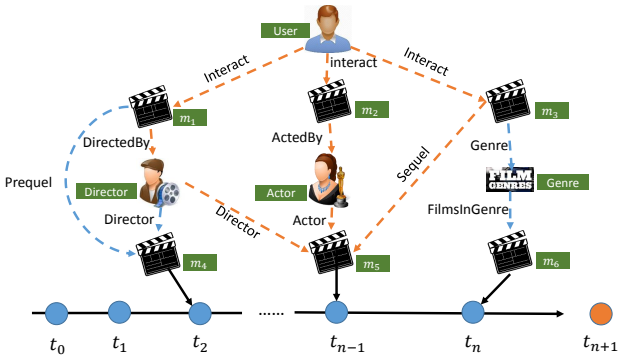


Figure 1: Examples of user-item interaction in movie domain. The orange lines represent paths connecting between the target user and item m_5.

sentations of entities and relations by regularizing the model with structural constraints. It is helpful to obtain high quality representations of entities, but lack of explanation ability because of the disregarding of semantic relations of entities that are connected by paths [21]. Users interact with items for different reasons. For example, as shown in Figure 1, the user watches the movie m_5, because either she/he likes the actor of the movie, or appreciates the director. It could also be because m_5 is the sequel of m_3 with which she/he interacted recently. Each path between a user-item pair contains different semantic information about the interaction. Therefore, it is important and necessary to consider the semantic paths constituting by entities and relations in constructing a convincing recommendation system. Some previous work incorporates meta-path into recommender by benefiting from the paths connecting two entities on different semantics [20, 28]. However, we argue that it is not the best way to model connections between two entities in KG, because meta-paths should be predefined manually which requires domain knowledge. The more recent work [21, 25] achieves more effective recommendations by modeling the sequential semantic paths to infer user interests. However, they do not consider the chronology of user behaviors, resulting the difficulties of modeling user's dynamic preferences with time drifting.

In order to address the problem of introducing semantic paths into SR system for capturing user's dynamic preferences and providing accurate explainable recommendations, we propose an **E**xplainable **I**nteraction-driven **U**ser **M**odeling (**EIUM**) approach for SR task. For improving the **accuracy** of recommender, we introduce the structural information from KG in the way of multi-modal fusion by incorporating textural, visual, structural knowledge into network, where the different modal features satisfy the constraints of the structural information of entities and relations in KG. They are jointly learned in a unified end-to-end way, which is complementary to item representations. To endow the recommendations **explainability**, we introduce the external KG into our model by leveraging the semantic paths extracted from KG. The specific user-item interaction representation can be obtained by modeling the corresponding paths. For integrating all the paths between each user-item pair, we adopt a weighted pooling layer to learn the different contributions of each path, which can facilitate the system to make decisions and provide *path-wise* explainable recommendations. For example: {*The movie m_5 is recommended since it's the sequel to movie m_3 you have watched.*}. Thereafter, a masked self-attention model is adopted to encode the user's sequential interactions for capturing the user's **dynamic** interests, which can be explained at *interaction-level*.

To the best of our knowledge, it is the first time to model user's dynamic preference at explainable interaction-level and path-level by incorporating external KG information. We evaluate our model in a large-scale user-item interaction dataset and the open general knowledge base. Experimental results show that compared with the state-of-the-art methods, our approach not only improves the accuracy of recommendations, but also holds the explanation capacity. The contributions of this work can be summarized as follows:

- We propose a novel explainable user modeling algorithm that captures the **interaction-level** user dynamic preference. It is a high-level representation which contains auxiliary semantic information.
- We endow the recommendation system the ability of **path-wise** explanation by modeling the semantic explicit paths between user-item pairs instead of implicit item embeddings.
- We adopt a joint learning manner for better representation learning by employing multi-modal fusion which benefits from the structural constraints in KG, where three kinds of modalities are involved in.

2 RELATED WORK

Sequential Recommendation (SR). Sequence modeling methods for SR mainly belong to Markov Models [4, 17] and RNNs[8, 9, 15, 27]. The scalable sequential models usually rely on Markov Chain (MC) to capture sequential patterns. L-order MC makes recommendations based on L previous actions. It cannot be directly applied to sequence-aware recommendation in most cases since data sparsity quickly leads to poor estimates of the transition matrices, and it also faces the challenge of the choice of the order [14]. Therefore, many studies are devoted to improving the MC-based approach. Inspired by the great power of Matrix Factorization (MF), Factorized Personalized Markov Chain (FPMC) combines the power of MF and MC to factorize the transition matrix over underlying MC to model personalized sequential behaviors for the problem of next-item recommendation given the last-N interactions of the user [4, 17]. RNN computes the current hidden state from the current input in the sequence and the hidden state outputted by the previous time step. The recurrent feedback mechanism memorizes the influence of each past data sample in the hidden state. It therefore makes RNN and its variants such as LSTM and GRU be able to model the temporal information for user behaviors in recommendation task [8, 9, 15, 27]. Though it is an effective way to encode users' behavior sequences, it still suffers from several difficulties, such as hard to parallelize, time-consuming, hard to preserve long-term dependencies. The emergence of the Transformer [22] architecture tackles the problem of sequence transduction. Some studies abandon the complex and time-consuming RNN structures, and instead construct the sequence model based on self-attention mechanism and apply it to SR system. ATRank [33] takes the lead in using self-attention structure for the SR and achieves encouraging results.

CSAN[12] adopts a feature-wise masked self-attention to construct user preference representations for SR.

Explainable Recommendation (ER). Explainable recommendation algorithm aims to address the problem of not only providing suitable recommendations to users, but also providing explanations for users to understand why they are recommended those items by the system [30]. Knowledge graph has been leveraged for ER recently since knowledge base contains rich external structural information of users and items. The methods that incorporate KG into recommendation system can be roughly categorized into two types: knowledge graph embedding (KGE) based methods and path-based methods. KGE-based methods usually combines content representations of the items themselves with knowledge-aware embeddings to generate a better representation for item [11, 23, 29]. The drawback of these approaches is that, although the accuracy of the recommendation can be improved, it is more difficult to provide an intuitive explanation of why the item is recommended to the user because the introduced KG embeddings are implicit. The disregard of semantic relations of entities that are connected by paths results in lacking the reasoning ability. Thus, many studies extend the general recommendation model with entity similarity derived from paths which refer to the connections between two entities through different semantics in KGs. Meta-path is a typical way of connecting object pairs in graph. It is a relational sequence which is widely used to extract structual features that capture relevant semantics for recommendation [20]. Some previous work has introduced the connectivity patterns into recomendation system [5, 7, 10, 18–20, 28]. However, the meta-path based methods rely heavily on handcrafted features and the quality of selected meta-paths, which additionally requires the domain knowledge. A more recent approach is to automatically capture the semantic associations through the model. The qualified paths between entity pairs from KG are mined automatically and then are encoded via recurrent networks. A recommendation layer is seamlessly integrated at the end of the network that can be trained in an end-to-end way for incorporating semantic structural knowledge into recommendation task [21, 25].

However, the existing ER methods are mainly used in ordinary top-N recommendation problems. There is still little research on the explainable sequential recommendation. We are committed to developing a more efficient and explainable sequence recommendation model to endow accuracy, dynamic and explanation to the recommendation system.

3 PRELIMINARY

Before introducing our proposed models, we first formally detail some concepts related to our work.

Knowledge Graph. Knowledge graph is defined as a graph $\mathcal{G} = (\mathcal{E}, \mathcal{R})$, where $\mathcal{E}$ denotes the set of entities $\mathcal{E} = \{e_1, e_2, ..., e_{|\mathcal{E}|}\}$, and $\mathcal{R}$ denotes the set of relations $\mathcal{R} = \{r_1, r_2, ..., r_{|\mathcal{R}|}\}$. The triplet (h, r, t) indicates a fact that there is a relationship r from head entity h to tail entity t, where $h, t \in \mathcal{E}$ and $r \in \mathcal{R}$.

Semantic Path. Semantic path refers to a sequence of entities connected by relations between entities e_i and e_j, which can be represented as: $p_x = \{e_i \xrightarrow{r_1} e_1 \xrightarrow{r_2} e_2 ... \xrightarrow{r_k} e_j\}$. Each two adjacent entities can be connected by different types of relations, thus forming multiple paths of different semantics.

Examples: As the example shown in Figure 1, the user can interact with the movie m_5 in different paths, such as:

(1) $path_1 = \{user \xrightarrow{Interact} m_2 \xrightarrow{Actedby} Actor \xrightarrow{Actor} m_5\}$

(2) $path_2 = \{user \xrightarrow{Interact} m_3 \xrightarrow{Sequel} m_5\}$

Interaction Representation via Semantic Paths. In previous research work, users' historical behavior is represented as a series of item themselves. Whether using item's ID or multi-modal features or other more complex side information, it is essentially the onefold representation. In fact, the user's behavior has subjective initiative, which makes interaction between the user and the specific item not follow the same pattern. For example, as shown in Figure 1, there are three different ways of *user*-m_5 interaction, containing different semantic information. We aim to mine the high-level interactive representations between users and each item, instead of low-level item-based onefold representation. By introducing external knowledge, we can describe this interaction in a more reasonable, convincible, and explainable way. We define the interaction between user u and item i as the collection of semantic paths: $interact(u, i) = \mathcal{P}(u, i) = \{p_1, p_2, ..., p_{|\mathcal{P}|}\}$ where p is a path, $|\mathcal{P}|$ denotes the total numbers of paths. After the path set of user-item interaction is obtained, the unified representation of user-item interaction can be obtained by learning. The learning model is explained in Section 4.2.

Problem Statement. General user behaviors can be interpreted using the binary relationship between a user and an item. We denote $\mathcal{U} = \{u_1, u_2, ..., u_{|\mathcal{U}|}\}$ as the set of users, and $\mathcal{I} = \{i_1, i_2, ..., i_{|\mathcal{I}|}\}$ as the set of items, where $|\mathcal{U}|, |\mathcal{I}|$ denotes the number of elements in the User and Item collection, respectively. We represent the user-item interaction with a triplet $\tau = \{u, interact, i\}$ where the $interact$ is a pre-defined relation between user and item. It is worth noting that the users in the dataset of the recommendation system are able to connect with the external knowledge graph due to the pre-defined $interact$ relation with the item. Hence, the historical sequential records can be represented as $\mathcal{B} = \{\tau_t, t = 1, 2, ..., T\}$. Based on these preliminaries, we are ready to define the sequential recommendation task. Given $\mathcal{B} = \{\tau_t, t = 1, 2, ..., T\}$ of a user towards items and the semantic paths $\mathcal{P}(u, i) = \{p_1, p_2, ..., p_{|\mathcal{P}|}\}$ of user-item pairs, the task is to predict the next item i_{T+1} that the user may interact with at time $T + 1$.

4 THE PROPOSED MODEL

In this section, we present the details of the proposed **EIUM** algorithm for the sequential recommendation. The overall framework is illustrated in Figure 2. There are three key components: (1) The structural information, which is naturally carried by KG, is incorporated through *multi-modal fusion* to facilitate better and higher-level representations learning for users and items, to improve the performance of recommender via *joint learning*. (2) The *interaction representation module* is to learn the semantic representation of user-item interaction by encoding a set of semantic paths between corresponding user and item. (3) The *sequential interactions modeling module* is to encode each user-item interactions sequentially with the goal of capturing the dynamic user preference drifting.

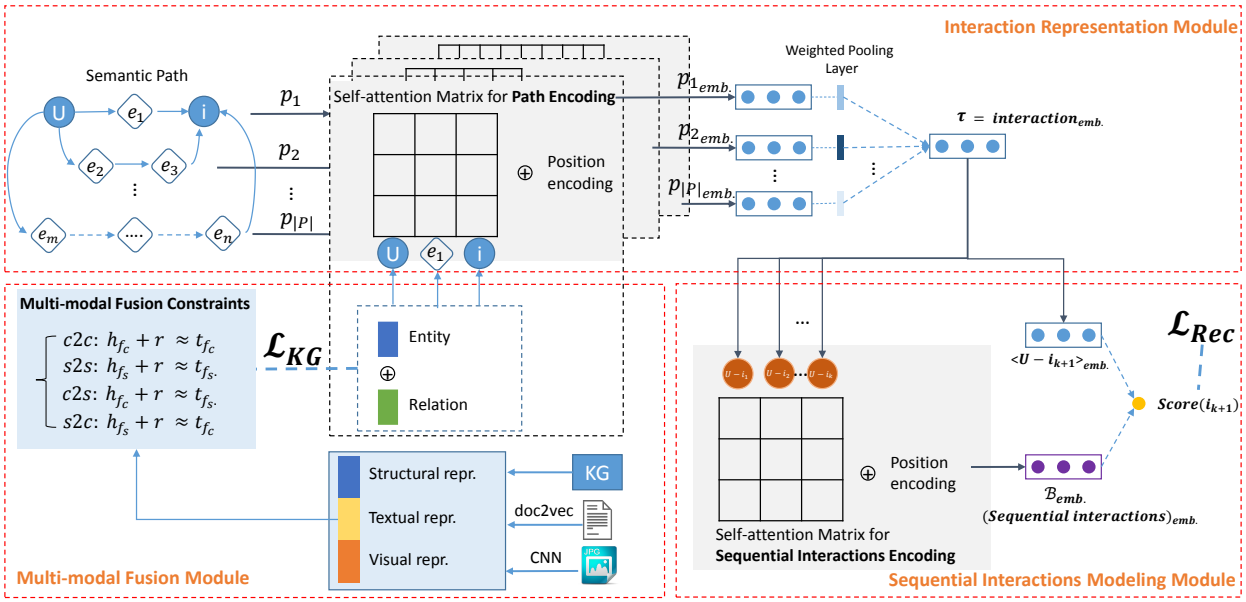


Figure 2: Illustration of the proposed EIUM algorithm. The whole framework contains three main components.

4.1 Multi-modal Fusion

To incorporate more auxiliary information into our model, we adopt the content and structural features to assist in item representation.

Content-features. Content-features include textual features and visual features. For **textual** features, we collect the item's title and description, and then fastText[1] is applied to extract the text representations. Specifically, we generate the each word's 300-d embedding $w \in \mathcal{R}^{300}$ through the pre-trained English word vectors [13], then obtain the complete textual features by $item^{textual} \in \mathcal{R}^{300} = \frac{1}{n}\sum_{i=1}^{n} w_i$, where n denotes word count of title and description. For **visual** features, we collect the visual descriptions of items such as movie's posters from IMDB website [2] through corresponding movie-id. We employ AlexNet pre-trained on ImageNet to extract 4096-dim visual semantic features from those posters. Due to the possible redundancy and noise of visual information, we reduce the high dimensional deep feature to 300-d with PCA [26] to obtain the final visual feature $item^{visual} \in \mathcal{R}^{300}$. Therefore, the content feature can be represented as:

$$f_c = \sigma(W_f[item^{textual}, item^{visual}] + b_f) \tag{1}$$

where $[,]$ is concatenation, W_f and b_f are the learnable parameters, and σ is a nonlinear activation function.

Structural features. We construct the structural representation of items by introducing external KG which contains abundant objective knowledge. The simple KGE method is used to embed entities and relationships into continuous vector space while preserving their structure information [24]. In our proposed model, we learn the structural feature f_s of entities and relations through the structural constraints of KG.

[1]https://fasttext.cc/
[2]https://www.imdb.com/

The multi-modal fusion is implemented by incorporating textual, visual, structural knowledge into network where the different modal features satisfy the constraints of the structure information of entities and relations in the graph[2], which is defined as $h + r \approx t$. For two different item features described above, there are four kinds of constraints by satisfying the corresponding structural constraints. We name it *multi-modal fusion constraints*:

$$\begin{aligned} c2c &: h_{f_c} + r \approx t_{f_c} \quad & s2s &: h_{f_s} + r \approx t_{f_s} \\ c2s &: h_{f_c} + r \approx t_{f_s} \quad & s2c &: h_{f_s} + r \approx t_{f_c} \end{aligned} \tag{2}$$

where relation r shares the same representation among constraints. The structural features are applied on the full graph. It is updated simultaneously during the training of the model by satisfying the constraints of multi-modal fusion. The multi-modal fusion is used to construct a better and more comprehensive user and item preference representation, which is detailed in Section 4.4.

4.2 Interaction Representation

Paths $\mathcal{P}(u, i)$ between a user-item pair contains different semantic information about the interaction between the user and the item. Therefore, the semantic representation of the interaction can be obtained by modeling the corresponding paths. Since a path is a sequence of several entities and relations, we use a sequence model to learn the semantic representation of the path. In order to simplify the structure of the model and improve the efficiency of training, inspired by [12, 22, 33], we adopt self-attention mechanism with position encoding module to model the path in which we can benefit not only from the ability in capturing the long-distance dependencies of the sequence with various lengths, but also from the capability in parallel computing for efficiently learning.

Self-attention layer. For each path $p_l = \{e_1 \xrightarrow{r_1} e_2 \xrightarrow{r_2} e_3 \ldots \xrightarrow{r_{L-1}} e_L\}$ in semantic paths $\mathcal{P}(u, i)$ between the user and the

item, the self-attention module encodes the path by learning the representations of entities and relations that constitute the path, and also obtaining the final semantic representations of the entire path. The self-attention module consists of two parts, one is self-attention matrix for attention score learning, the other is position encoding matrices for incorporating sequence information into path encoding, which is inspired by [12]. Two forward and backward Position Encoding Matrices are combined with self-attention architecture, named "*masked self-attention*", to preserve the temporal information for sequence modeling. The forward and backward matrices are defined as:

$$M_{i,j}^{fw} = \begin{cases} -|d_{i,j}|, & i < j \\ -\infty, & otherwise \end{cases} \tag{3}$$

$$M_{i,j}^{bw} = \begin{cases} -|d_{i,j}|, & i > j \\ -\infty, & otherwise \end{cases} \tag{4}$$

where $d_{i,j} = exp(|i,j|)$, $|i,j| = 1$ if i,j is adjacent, and so on.

The input of self-attention layer is a sequence in which each element consists of two raw feature vectors e and r that describe entity and relation, respectively. e and r are randomly initialized at first, and then will be updated iteratively with the training of the model. We concatenate the embedding of entity $e_i \in \mathcal{R}^d$ and relation $r_i \in \mathcal{R}^d$, and then embed it in a latent space by a fully-connected layer. The output is regarded as the element x_i:

$$x_i = \sigma(W_x[e_i, r_i] + b_x) \tag{5}$$

where $x_i \in \mathcal{R}^d$. Note that, the first entity e_1 in a path of $\mathcal{P}(u,i)$ must be the user u, and the corresponding relation is $interact$. The last entity e_L is the item i which has no relation to the next entity, so we give a 'END' as the relation to the last entity.

Take the path sequence $x = \{x_1, x_2, ..., x_L\}$ as input to the masked self-attention model, $f(x_i, x_j)$ denotes the correlations between the two elements:

$$f(x_i, x_j) = W^T \sigma(W_1 x_i + W_2 x_j) + \gamma M_{i,j} \tag{6}$$

where γ denotes *position scalar* which is a trade-off parameter of relative position difference. (Note that $M_{i,j}$ denotes $M_{i,j}^{fw}$ or $M_{i,j}^{bw}$ which are two independent processes. For simplicity, we only describe one-way processes. The final output is the concatenation of forward and backward encoding representation.)

Then, the attention score between x_i and x_j is defined as:

$$a_{ij} = \frac{e^{[f(x_i, x_j)]}}{\sum_{j=1}^{|L|} e^{[f(x_i, x_j)]}} \tag{7}$$

After obtaining attention scores over all entities, the output for x_j is defined as:

$$o_j = \sum_{i=1}^{L} a_{ij} \odot x_i \tag{8}$$

To obtain the unified embedding of a single path, we perform the mean-pooling operation on the output path sequence:

$$p_{l_{emb.}} = mean\text{-}pooling(o_j)_{j=1}^{L} = \frac{1}{L} \sum_{j=1}^{L} o_j \tag{9}$$

where $p_{l_{emb.}}$ combines the semantics sequence information of the path. Note that, the length L of paths in $\mathcal{P}(u,i)$ are different. Since the number of paths between different user-item pairs is dynamically changing, the number of self-attention layers will also change. The self-attention networks in this module share parameters to avoid overfitting.

Weighted pooling layer. For all paths $\{p_1, p_2, ..., p_{|\mathcal{P}|}\}$ in $\mathcal{P}(u,i)$, we obtain the semantic embeddings $\{p1_{emb.}, p2_{emb.}, ..., p|\mathcal{P}|_{emb.}\}$. Since different paths are generated by the combination of different relations with their corresponding intermediate entities, each path contains different semantic information and plays a different role in modeling user-item interaction. Hence, we use a weighted pooling layer to help distinguish the path importance. Attention mechanism [1] is adopted to address this issue. We use the user u and item i as the query to learn the weighted score :

$$query = \sigma_q(W_q[u,i] + b_q) \tag{10}$$

$$w(\mathcal{P}(u,i)) = [w_1, w_2, ..., w_{|\mathcal{P}|}] \tag{11}$$

where $query \in \mathcal{R}^d$, $w_1 + w_2 + ... + w_{|\mathcal{P}|} = 1$. The unified **interaction representation** is obtained by aggregating the weighted paths, which is also denoted by τ:

$$\tau = ineraction_{emb.} = \sum_{l=1}^{|\mathcal{P}|} w_l \cdot p_{l_{emb.}} \tag{12}$$

where τ combines path representations based on their contributions to reveal the reasoning process when dealing with the user-item pair. Therefore, our proposed model can infer the rationales underlying the user-item interaction to explain the recommended results. Detailed and intuitive visualizations are displayed in Section 6.

4.3 Sequential Interactions Modeling

A user's historical records are a sequence in chronological order, thus her/his subsequent item can be predicted by SR methods. Since the marked self-attention network is able to capture and characterize the temporal dependency in sequence effectively and efficiently, we still adopt the architecture to model sequential interactions.

Given a user's interacted item sequence $\{i_t, t = 1, 2, ..., T\}$, the sequential interactions can be represented as $\mathcal{B} = \{\tau_t, t = 1, 2, ..., T\}$ as detailed above. Similarly, the user preference representation based on sequential interactions is defined as:

$$\begin{aligned} f(\tau_i, \tau_j) &= W_\tau^T \sigma(W_\tau^1 \tau_i + W_\tau^2 \tau_j) + \gamma M_{i,j} \\ a_{\tau_i, \tau_j} &= \frac{e^{[f(\tau_i, \tau_j)]}}{\sum_{j=1}^{T} e^{[f(\tau_i, \tau_j)]}} \\ e_{\tau_j} &= \sum_{\tau=1}^{T} a_{\tau_i, \tau_j} \odot \tau_i \\ \mathcal{B}_{emb.} &= \sum_{t=1}^{T} w_t \cdot e_{\tau_t} \end{aligned} \tag{13}$$

where w_t denotes the attention scores of interactions which is obtained by taking the predicted item as the query. The output embedding $\mathcal{B}_{emb.}$ denotes user dynamic preference which can be used to predict probability of user u engaging item v through a predicting function f, which can be inner product or H-layer MLP:

$$p_{u,v} = \sigma(f(\mathcal{B}_{emb.}, \tau_v)) \tag{14}$$

4.4 Joint Learning of Sequential Recommendation and Multi-modal Fusion

The ultimate goal of the task is to rank the ground-truth item v higher than all other items $v^{'}$ ($v^{'} \in \{\hat{\mathcal{V}} = \mathcal{I}\ w/o\ v\}$), which is defined as follows through cross-entropy function:

$$\mathcal{L}_{rec} = \sum_{u} \sum_{v} \sum_{v' \in \hat{\mathcal{V}}} -log\sigma[D(uv) - D(uv^{'})] \tag{15}$$

where $D(\cdot)$ is the distance function which can be a dot-product or a more complex deep neural network. $\sigma(\cdot)$ is the sigmoid function.

In addition, we also consider the structural information based on KG to facilitate sequential recommendation. As described above, since the items in KG satisfy the multi-modal fusion constraints. The four losses from KG can be included into our model:

$$\begin{aligned} \mathcal{L}_{kg} &= \mathcal{L}_{c2c} + \mathcal{L}_{s2s} + \mathcal{L}_{c2s} + \mathcal{L}_{s2c} \\ &= \frac{1}{4} \sum_{i} ||h + r - t||, i \in \{c2c, s2s, c2s, s2c\} \end{aligned} \tag{16}$$

Therefore, we train our proposed EIUM model by minimizing the overall objective function that jointly evaluates the performance:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda \mathcal{L}_{kg} \tag{17}$$

where λ is the balancing parameter which is analyzed in detail later. And $L2$ regularization is introduced to reduce overfitting.

5 EXPERIMENTS

5.1 Dataset Description

For recommendation data, we use MovieLens-20M[3] which offers the user-item interaction data in movie domain. We filter out the unpopular items which are watched less than 20 times, and inactive users who have fewer than 20 interactions. The threshold of positive rating is 4. We consider the recommendation task targeting for implicit feedback like previous efforts [6, 21, 28]. We define the target value of user-item equals to 1 when user-item interaction is observed, and 0 otherwise. For each positive sample, we follow [6, 11, 25] and randomly sample 100 items that the user has not interacted with to generate negative samples that pair it. Given the user's interaction sequence, we make the first k interactions to predict the $(k+1)$-th interaction in the training set, where $k = 1, 2, ..., n-2$, and we use the first $n-1$ behaviors to predict the last one in the test set. n is the total number of interactions.

For knowledge base, we adopt the Freebase data dumps[4]. Refer to previous work [3], to collect the related facts from Freebase, we retain the triplets associated with those entities which are mapped with items in recommendation data. The mapping relationship between items of MovieLens-20M and entities of Freebase is provided by [11, 32]. We select the most frequently occurring types of relationships, including genre, director, writer, actor, prequel, etc. The statistics of the dataset are presented in Table 1.

5.2 Semantic Path Extraction

To fully explore the linkages between entities through different relationships and the arrangement of entities, we use a series of strategies to mine the semantic associations between entities. It is pointed out in previous work [20, 21, 25] that truncating all paths of a certain length and ignoring remote connections are sufficient to model connections between user-item pairs, and moreover, paths with length greater than six will introduce noisy entities [20]. Therefore, we limit the path length between user-item up to six and extract the valid paths based on this policy. Unlike traditional top-N recommendations, the time when user behavior occurs is a sensitive and important factor in SR task. In order to ensure the preciseness of the experiments, we only seek the paths between the user u and the current item i_t through the items $i_1, i_2, ..., i_{t-1}$ which are interacted by the user before time t.

[3]https://grouplens.org/datasets/movielens/20m/
[4]https://developers.google.com/freebase/

Table 1: Statistics of our dataset

User-Item Interaction	# Users	138,287
	# Items	13,047
	# Interactions	9,971,069
	# Avg.interaction	72
Knowledge Graph	# Entities	40,529
	# Relations	66
	# Triplets	7,422,000
Path	# Paths	146,222,314
	# Avg.path	16.62
	# Avg.path_length	5.04
Train-test Dataset	# Train samples	691,435
	# Test samples	138,287
	# Images	13047

5.3 Experimental Settings

5.3.1 Evaluation Metrics. A variety of widely used evaluation metrics are adopted to evaluate our approach, including AUC [16], Mean Average Precision (MAP), Hit Ratio (Hit@N), and Normalized Discounted Cumulative Gain (NDCG@N).

5.3.2 Training Details. The whole model is trained in an end-to-end way with Adam optimizer. We apply a grid search for the learning rate and find $lr = 0.01$ is the best. A grid search in $\{2^n, n = 5to9\}$ is applied to find out the best setting of the embedding dimension, and finally set $dim. = 64$. The position scalar is searched in $\{0.1to1|interval = 0.1\}$ to find the best scalar equals to 0.6. The batch size is set to 32. In addition, we truncate the user's historical records to reducing the noise. A detailed analysis is illustrated in Section 5.6. For the comparative methods, the parameters are set as suggested by the original papers.

5.4 Comparison Methods

- **BPR**. Bayesian personalized ranking[16] is a pairwise ranking framework which takes Matrix Factorization as the underlying predictor.
- **Bi-LSTM**. Bi-directional Long short-term memory (LSTM) units are building unit for the layer of RNN, which are used to capture sequential dependencies and make predictions[31].
- **Bi-LSTM with attention**. Incorporating attention mechanism into Bi-LSTM methods mentioned above.
- **ATRank**. ATRank[33] is an attention-based user modeling framework which encoding sequential behaviors based only on self-attention mechanism.

Table 2: The next-one recommendation performance of all the methods across all the evaluation metrics. The best performance is boldfaced; the highest score in baseline is labeled with '∗'; the percentage in parentheses represents the relative improvements that baselines achieve w.r.t EIUM.

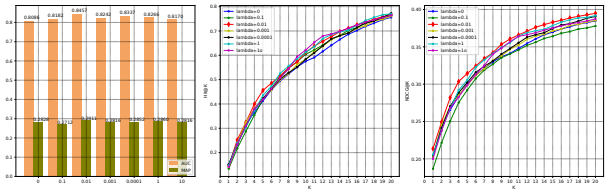
Datasets	Methods	Evaluation Metrics					
		AUC	MAP	Hit@5	Hit@10	NDCG@5	NDCG@10
MovieLens-20M + Freebase	BPR	0.9065 (-0.18%)	0.3312 (-21.05%)	0.4920 (-19.65%)	0.6653 (-16.68%)	0.3578 (-21.95%)	0.3990 (-21.19%)
	Bi-LSTM	0.8800 (-3.09%)	0.3278 (-21.86%)	0.5583 (-8.82%)	0.7180 (-10.08%)	0.3749 (-18.22%)	0.4147 (-18.09%)
	Bi-LSTM+att.	0.8897 (-2.03%)	0.3606 (-14.04%)	0.5944* (-2.92%)	0.7409* (-7.21%)	0.4095 (-10.67%)	0.4460 (-11.91%)
	ATRank	0.8724 (-3.93%)	0.3454 (-17.66%)	0.5561 (-9.18%)	0.7057 (-11.62%)	0.3877 (-15.42%)	0.4250 (-16.06%)
	CKE	0.9054 (-0.30%)	0.3869* (-7.77%)	0.5790 (-5.44%)	0.7175 (-10.14%)	0.4261* (-7.05%)	0.4571* (-9.72%)
	KTUP	**0.9187*** (+1.17%)	0.3829 (-8.72%)	0.5662 (-7.53%)	0.7085 (-11.27%)	0.4196 (-8.46%)	0.4516 (-10.80%)
	EIUM	0.9081	**0.4195**	**0.6123**	**0.7985**	**0.4584**	**0.5063**

- **CKE**. CKE[29] is a recently proposed state-of-the-art method that incorporates KG embedding to improve the recommendation performance.
- **KTUP**. KTUP[3] is a knowledge-enhanced translation-based user preference model. It transfers the relation embeddings as well as entity embeddings learned from KG to user preference model, and simultaneously training two different tasks.
- **EIUM**. It is our proposed model detailed in Section 4.

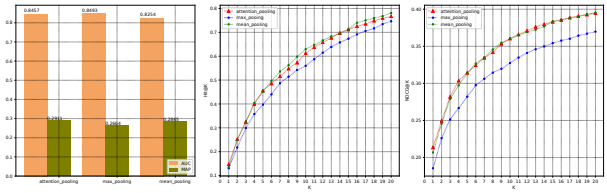
5.5 Performance Analysis

The experimental results of all the methods are illustrated in Table 2, we have the following main findings:

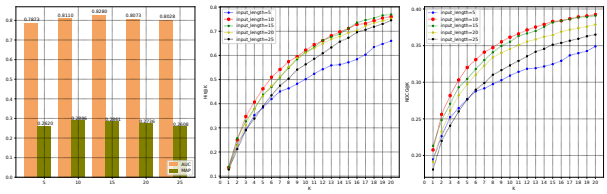
- BPR gets the worst performance because it is a pair-wise ranking method with no additional auxiliary information, which results in failure of capturing user's accurate preferences. This helps to confirm the usefulness of side information, such as time, multi-modality, and KG information.
- As for SR baseline methods which refer to {ATRank, LSTM, Bi-LSTM+attention}, Bi-LSTM with attention approach performs the best because the bi-directional structure uses the information from the past and the future. In addition, the adopted attention layer helps to model the specific relation patterns in the sequence.
- As for Knowledge-based baseline methods which refer to {CKE, KTUP}, CKE achieves better performance than KTUP, because CKE utilizes the KG embeddings as the additional information which preserve the information in different two sources to enhance the recommendation. The reason why KTUP is not doing well may be because the multi-task architecture is more suitable for sparse datasets, but our employed dataset MovieLen-20M and Freebase are large and dense.
- Bi-LSTM and CKE achieve comparable results, which indicates that time information is as important as external knowledge in sequence recommendation task.
- Our proposed EIUM model performs better than other baseline methods in most cases. This is due to the introduction of the external KG information into recommendation model. User's historical behaviors are represented as interaction-level representations learning from the semantic paths, which is a high-level semantic representation instead of traditional low-level item-based embedding representation. Moreover, the structural information is exploited in recommendation by multi-modal fusion way. In addition to improving the effectiveness of recommendations, EIUM also offers highly explainable results, which will be analyzed in Section 6.



(a) Effects of KG loss in EIUM.



(b) Effects of pooling manner in EIUM.



(c) Impact of historical behaviors in EIUM.

Figure 3: Component Analysis.

5.6 Component Analysis

5.6.1 Effects of KG loss. To determine the effectiveness of introducing structural information into recommendation model, we offer ranging trade-off parameters λ to KG loss. From the results shown in Figure 3(a), we observe that the accuracy is not monotone with λ,

since if λ is too large, the recommendation loss will be invalid, and if the λ is too small, the graph structure information constraints will not work for making fully use of structural knowledge from KG. The EIUM achieves the best performance when $\lambda = 0.01$.

5.6.2 Effects of pooling manner. Considering to provide users with a better explanation, attention mechanism is adopted to measure the importance of each path in our proposed algorithm. To analyze its effect, we compare attention manner with two other pooling manners including max-pooling and mean-pooling ways. In Figure 3(b), we find that using attention makes the comparable performance in recommendation. In addition, the interactive paths can be selected according to the attention score which also can provide a more intuitive explanation for users. However, max and mean pooling manners work on each dimension of the hidden vector which result in lacking of explanability.

5.6.3 Impact of length of historical behaviors. In terms of common sense, the next item that user will interact with is affected more by the recent behaviors than by the older behaviors. Many behaviors with long intervals may introduce noise information into the system. Thus, our hypothesis is that appropriate length of user's historical records (input-length) as the input of the model is helpful to accurately predict the user's dynamic preference. To emprically investigate the impact of input-length on recommendation performance, we implement EIUM with input-length in $\{5, 10, 15, 20, 25\}$, the results are shown in Figure 3(c). It demonstrates the rationality of our hypothesis, and the appropriate input-length is 10.

6 RECOMMENDATION EXPLAINABILITY

By incorporating KG into SR for not only improving recommendation performance, but also providing the highly convincible explanations to users. In this section, we present an example to demonstrate the explainability of our proposed EIUM model. As shown in Figure 4, we randomly select a user and import the user's historical records into our model and then the recommended top-3 items are presented. There are several observations as follows: (1) The movie *The Matrix* comes first in the recommendation list, and this is our ground-truth item. It demonstrates the accuracy of our model in sequential recommendation. (2) The recommended top-2 movies *The Matrix* and *Miss Congeniality* share the same genre '*Action*' with the last three historical movies, which indicates that the model is able to capture the short-term interests of the user. Meanwhile, the 3rd recommended movie *Lost in Translation* share the same genre '*Drama*' with the first three historical movies, indicating that our algorithm is able to model the user's long-term taste. The two findings demonstrate that the proposed EIUM can capture the dynamic preference of users effectively. (3) For the first recommended movie *The Matrix*, we show the relative scores (blue numbers) on different user-movie interactions in historical records. The movie *Star Wars: Episode VI* achieves the highest score, which represents that it is the most relative movie of *The Matrix*. It is reasonable because the two movies belong to the same genre, which provides the **interaction-level explanation** to users. (4) Furthermore, we show the extracted qualified paths connecting the user and the most relative movie from KG*Star Wars: Episode VI*. There are six paths with different attention scores which refers to the different

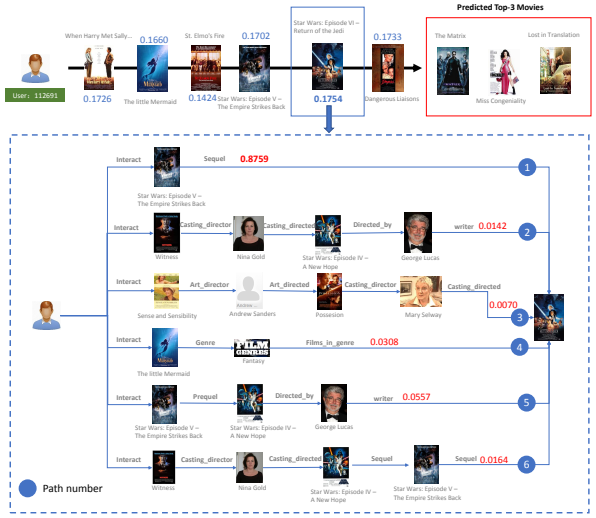


Figure 4: A running example for demonstration of explainability of EIUM model. The blue numbers in historical records denote attention scores for each interaction. The red numbers in each path refer to the attention score for each path between the current user and movie.

importances of paths. The first path gains absolute high score with the strong relations 'Sequel'. This suggests that EIUM can well infer the user's specific preference from different angels with diversified semantic paths, which offers the **path-level explanation** to users. The varied paths carry different semantic information, connecting the user and the movie with different importances, which provides highly explainability for recommendation system.

7 CONCLUSION

We introduce a novel explainable interaction-driven user modeling algorithm to better capture the users' interaction-level dynamic preferences in an explainable way in SR tasks. The user-item interactions are constructed by several semantic paths extracted from knowledge graph, which endows the SR system the ability of accuracy and explainability. Extensive experiments demonstrate the superior ability of our proposed EIUM model on providing effective and convincing recommendations to users. In the future, we will continue to extend EIUM in better incorporating user's profiles and contextual information with external KG for potential preferences seeking in dealing with cold-start recommendation problems.

ACKNOWLEDGMENTS

This work was supported in part by the National Key Research and Development Program of China (No. 2017YFB1002804), the National Natural Science Foundation of China under Grants 61432019, 61720106006, 61572503, 61802405, 61872424, 61702509 and 61832002, the Key Research Program of Frontier Sciences, CAS, Grant NO. QYZDJ-SSW-JSC039, the Beijing Municipal Science & Technology Commission (No. Z181100008918012), and the K.C.Wong Education Foundation.

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[2] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*. 2787–2795.

[3] Yixin Cao, Xiang Wang, Xiangnan He, Tat-Seng Chua, et al. 2019. Unifying Knowledge Graph Learning and Recommendation: Towards a Better Understanding of User Preferences. *arXiv preprint arXiv:1902.06236* (2019).

[4] Chen Cheng, Haiqin Yang, Michael R Lyu, and Irwin King. 2013. Where You Like to Go Next: Successive Point-of-Interest Recommendation.. In *IJCAI*, Vol. 13. 2605–2611.

[5] Li Gao, Hong Yang, Jia Wu, Chuan Zhou, Weixue Lu, and Yue Hu. 2018. Recommendation with multi-source heterogeneous information. *structure* 1, w3 (2018), w4.

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

[7] Benjamin Heitmann and Conor Hayes. 2010. Using linked data to build open, collaborative recommender systems. In *2010 AAAI Spring Symposium Series*.

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

[9] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel recurrent neural network architectures for feature-rich session-based recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, 241–248.

[10] Binbin Hu, Chuan Shi, Wayne Xin Zhao, and Philip S Yu. 2018. Leveraging meta-path based context for top-n recommendation with a neural co-attention model. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1531–1540.

[11] Jin Huang, Wayne Xin Zhao, Hongjian Dou, Ji-Rong Wen, and Edward Y Chang. 2018. Improving sequential recommendation with knowledge-enhanced memory networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 505–514.

[12] Xiaowen Huang, Shengsheng Qian, Quan Fang, Jitao Sang, and Changsheng Xu. 2018. CSAN: Contextual Self-Attention Network for User Sequential Recommendation. In *2018 ACM Multimedia Conference on Multimedia Conference*. ACM, 447–455.

[13] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in Pre-Training Distributed Word Representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

[14] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. 2018. Sequence-Aware Recommender Systems. *ACM Comput. Surv.* 51, 4, Article 66 (July 2018), 36 pages. https://doi.org/10.1145/3190616

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

[16] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*. AUAI Press, 452–461.

[17] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*. ACM, 811–820.

[18] Chuan Shi, Zhiqiang Zhang, Ping Luo, Philip S Yu, Yading Yue, and Bin Wu. 2015. Semantic path based personalized recommendation on weighted heterogeneous information networks. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 453–462.

[19] Yizhou Sun and Jiawei Han. 2013. Mining heterogeneous information networks: a structural analysis approach. *Acm Sigkdd Explorations Newsletter* 14, 2 (2013), 20–28.

[20] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment* 4, 11 (2011), 992–1003.

[21] Zhu Sun, Jie Yang, Jie Zhang, Alessandro Bozzon, Long-Kai Huang, and Chi Xu. 2018. Recurrent knowledge graph embedding for effective recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*. ACM, 297–305.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. 6000–6010.

[23] Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018. Dkn: Deep knowledge-aware network for news recommendation. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1835–1844.

[24] Hongwei Wang, Fuzheng Zhang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2019. Multi-Task Feature Learning for Knowledge Graph Enhanced Recommendation. *arXiv preprint arXiv:1901.08907* (2019).

[25] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. 2018. Explainable Reasoning over Knowledge Graphs for Recommendation. *arXiv preprint arXiv:1811.04540* (2018).

[26] Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and intelligent laboratory systems* 2, 1-3 (1987), 37–52.

[27] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A dynamic recurrent model for next basket recommendation. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. ACM, 729–732.

[28] Xiao Yu, Xiang Ren, Yizhou Sun, Quanquan Gu, Bradley Sturt, Urvashi Khandelwal, Brandon Norick, and Jiawei Han. 2014. Personalized entity recommendation: A heterogeneous information network approach. In *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 283–292.

[29] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM, 353–362.

[30] Yongfeng Zhang and Xu Chen. 2018. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192* (2018).

[31] Yuyu Zhang, Hanjun Dai, Chang Xu, Jun Feng, Taifeng Wang, Jiang Bian, Bin Wang, and Tie-Yan Liu. 2014. Sequential Click Prediction for Sponsored Search with Recurrent Neural Networks.. In *AAAI*, Vol. 14. 1369–1375.

[32] Wayne Xin Zhao, Gaole He, Kunlin Yang, Hongjian Dou, Jin Huang, Siqi Ouyang, and Ji-Rong Wen. 2019. KB4Rec: A Data Set for Linking Knowledge Bases with Recommender Systems. *Data Intelligence* 1, 2 (2019), 121–136.

[33] Chang Zhou, Jinze Bai, Junshuai Song, Xiaofei Liu, Zhengchao Zhao, Xiusi Chen, and Jun Gao. 2018. Atrank: An attention-based user behavior modeling framework for recommendation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.