Leveraging Meta-path based Context for Top-N Recommendation with A Neural Co-Attention Model

Binbin Hu, Chuan Shi*
Beijing University of Posts and Telecommunications, Beijing, China
{hubinbin,shichuan}@bupt.edu.cn

Wayne Xin Zhao*
School of Information, Renmin University of China, Beijing, China
batmanfly@gmail.com

Philip S. Yu†
University of Illinois at Chicago
IL, USA
psyu@cs.uic.edu

ABSTRACT

Heterogeneous information network (HIN) has been widely adopted in recommender systems due to its excellence in modeling complex context information. Although existing HIN based recommendation methods have achieved performance improvement to some extent, they have two major shortcomings. First, these models seldom learn an explicit representation for path or meta-path in the recommendation task. Second, they do not consider the mutual effect between the meta-path and the involved user-item pair in an interaction. To address these issues, we develop a novel deep neural network with the co-attention mechanism for leveraging rich meta-path based context for top-N recommendation. We elaborately design a three-way neural interaction model by explicitly incorporating meta-path based context. To construct the meta-path based context, we propose to use a priority based sampling technique to select high-quality path instances. Our model is able to learn effective representations for users, items and meta-path based context for implementing a powerful interaction function. The co-attention mechanism improves the representations for meta-path based context, users and items in a mutual enhancement way. Extensive experiments on three real-world datasets have demonstrated the effectiveness of the proposed model. In particular, the proposed model performs well in the cold-start scenario and has potentially good interpretability for the recommendation results.

CCS CONCEPTS

• Information systems → Collaborative filtering; **• Computer systems organization → Heterogeneous (hybrid) systems**;

KEYWORDS

Recommender System; Heterogeneous Information Network; Deep Learning; Attention Mechanism

*Corresponding author
†Philip S. Yu is also with the Institute for Data Science, Tsinghua University, Beijing, China.

KDD 2018, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219965

ACM Reference Format:
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 *KDD 2018: 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3219965

1 INTRODUCTION

In the era of information overload, recommender systems play a pivotal role in various online services, which aim to match user interests with resource items [25]. Classic recommendation methods, *e.g.,* matrix factorization [15], mainly model users' preference towards items using historical user-item interaction records. Nowadays, various kinds of auxiliary data become increasingly available in online services. Many methods further propose to leverage these context information for improving recommendation performance [1, 24]. Due to the heterogeneity and complexity of auxiliary data, it is still challenging to effectively utilize such context information in recommender systems.

As a promising direction, *heterogeneous information network* (HIN), consisting of multiple types of nodes and links, has been proposed as a general information modeling method [28, 30]. Since HIN is flexible to characterize various kinds of heterogenous data, it has been adopted in recommender systems to model rich auxiliary data [40, 41]. In particular, *meta-path*, a relation sequence connecting object pairs in HIN, is widely used to extract structural features that capture relevant semantics for recommendation [30]. We present an example for movie recommendation characterized by HIN in Fig. 1(a). Roughly speaking, existing HIN based recommendation methods can be categorized into two types. The first type leverages path based semantic relatedness over HIN as *direct features* for recommendation relevance [5, 29, 40]. As a comparison, the second type performs some transformation on path based similarities (*e.g.,* matrix factorization on the path based similarity matrix) for learning effective *transformed features*, which are subsequently used to enhance original user or item representations in the recommendation methods [40, 41]. These two types of methods both extract meta-path based features for improving the characterization of two-way user-item interaction, as illustrated in Fig. 1(b).

Although the above methods have achieved performance improvement to some extent, there are two major shortcomings with existing HIN based methods. First, they do not learn an explicit representation for path or meta-path in the recommendation method,

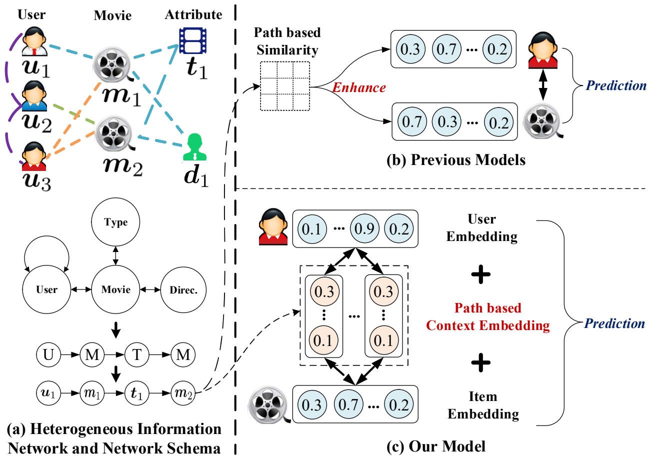


Figure 1: The illustration for HIN based recommendation setting (network schema, meta-path, path instance) and the comparison between our model and previous methods (two-way interaction *v.s.* three-way meta-path based interaction).

or the learned representations are not tailored for the recommendation task. Second, they still characterize two-way user-item interactions, and seldom consider the mutual effect between the meta-path and the involved user-item pair in an interaction. Without considering the above two aspects, path based features may not be optimal for recommendation, and lack direct explanations of why they work or not. In addition, existing HIN based recommendation [26, 41] methods mainly focus on the task of rating prediction, while we may only have implicit feedback available in practice.

To address these issues, we aim to leverage rich meta-path information from HIN for top-N recommendation in a more principled way. For convenience, given a target user-item pair, we call the aggregate meta-paths (together with their corresponding path instances) connecting the user with the item *meta-path based context* of the interaction. Our main idea is to (1) learn explicit representations for meta-path based context tailored for the recommendation task, and (2) characterize a three-way interaction of the form: ⟨*user*, *meta-path*, *item*⟩. The paradigm of our work is illustrated in Fig. 1(c). By explicitly incorporating meta-paths into the interaction model, our approach is able to effectively mine and extract useful information from meta-path based context for improving the recommendation performance. In this way, the characterization of meta-path based context will be more flexible to adapt to different interaction scenarios, providing a good interpretability on the recommendation results, *i.e., who* will adopt *what* given the *context*. The idea is appealing, while the solution is challenging. We have to consider three key research problems: (1) how to design the base architecture that is suitable for the complicated HIN based interaction scenarios; (2) how to generate meaningful path instances for constructing high-quality meta-path based context; and (3) how to capture the mutual effect between the involved user-item pair and meta-path based context in an interaction.

In this paper, for tackling various complicated interaction scenarios, we adopt deep neural networks to build the base architecture for our recommender, since it has been shown that deep neural networks are more capable of learning arbitrary interaction function from data [11]. We elaborately design a three-way neural interaction model by explicitly incorporating meta-path based context into the interaction. To construct the meta-path based context, we propose to use a priority based sampling technique to select high-quality path instances for recommendation, since a simple meta-path guided sampling strategy like that in [4] is likely to generate low-quality path instances or even noise for our task. Our model needs to learn the representations for users, items and meta-path based context. We propose a novel co-attention mechanism to mutually improve the representations for meta-path based context, users and items. Specially, the representations for meta-path based context are first improved according to the information from a user-item pair in an interaction, and then user and item representations are further enhanced conditioned on the improved representations of meta-path based context. In this way, the meta-path based context are transformed into a form that is directly useful for a specific interaction, which is supposed to be more effective in recommendation than original representations. Meanwhile, the improved user and item representations also embody useful evidence from calibrated meta-path based context for the current interaction. By comparing Fig. 1(b) and (c), we can see that, different from previous methods, our proposed model explicitly incorporates the meta-path based context into the interaction, and learns interaction-specific representations for these useful information.

To our knowledge, it is the first time that meta-path based context has been explicitly modeled in a three-way neural interaction model, *i.e.,* ⟨*user*, *meta-path*, *item*⟩, for the task of top-N recommendation in HIN. We propose a novel deep neural network with the co-attention mechanism by leveraging rich meta-path based context, which is able to learn interaction-specific representations for users, items and meta-path context. We do extensive experiments on three real-world datasets, which demonstrate the effectiveness of the proposed model compared to the state of arts. In addition, we also validate that the proposed model has the potential to alleviate cold-start problem and provide interpretable recommendation results.

2 RELATED WORK

In the literature of recommender systems, early works mainly adopt collaborative filtering (CF) methods (*e.g.,* matrix factorization) to utilize historical interactions for recommendation [16]. Since CF methods usually suffer from cold-start problem, many works attempt to leverage additional information for recommendation, such as social information [33, 42, 43], location information [39], and heterogeneous information [5]. In addition, there are also some general feature based frameworks for incorporating context information for recommendation [3, 10, 22]. Recently, deep network models are also employed to extract refined latent features [11] from the user-item interaction data.

As a newly emerging direction, heterogeneous information network [28] can naturally model complex objects and their rich relations in recommender systems, in which objects are of different types and links among objects represent different relations [27, 30]. Due to the flexibility of HIN in modeling various kinds of heterogeneous data, it has been adopted in recommender systems to model rich auxiliary data. Most of HIN based methods usually utilize path based similarity to enhance the representations of users and items, including meta-path based latent features [40], meta-path based user similarity [18, 29] and meta-graph based latent features [41]. However, they seldom learn explicit representation for path or meta-path tailored to the recommendation task

On the other hand, network embedding has shown its potential in structure feature extraction and has been successfully applied in many data mining tasks. For example, Deepwalk [19] and node2vec [7] combine random walk and skip-gram to learn network representations. Most of network embedding methods focus on homogeneous networks, and thus they cannot directly be applied to heterogeneous networks. Recently, attention is increasingly shifting towards heterogeneous networks. Xu et al. [38] propose an Embedding of Embedding model to encode the intra-network and inter-network edges for the coupled heterogeneous network. Dong et al. [4] obtain the neighbors of a node via meta-paths and learn the HIN embedding by skip-gram with negative sampling. Furthermore, Fu et. al [6] learn node embedding to capture rich relation semantics in HIN via neural network model. Although these HIN embedding methods has shown their effectiveness in some tasks, they usually focus on general node embeddings, seldom considering the path embedding for the recommendation task.

Our work is inspired by the recent progress of neural attention mechanism in the fields of computer vision [37] and natural language processing [21]. In particular, co-attention or cross-attention mechanisms have been applied to solve complicated NLP tasks [8, 36]. We are also aware of the application of attention mechanism in recommender systems [2, 34, 35]. We borrow the idea of co-attention mechanisms for modeling mutual effect between the meta-path and the involved user-item pair in an interaction. To our knowledge, it is the first time that meta-path based context has been explicitly modeled in a three-way neural interaction model with the co-attention mechanism.

3 PRELIMINARIES

In this paper, we consider the recommendation task targeting for implicit feedback. With n users $\mathcal{U} = \{u_1, ..., u_n\}$ and m items $\mathcal{I} = \{i_1, ..., i_m\}$, we define each entry $r_{u,i}$ in the user implicit feedback matrix $\mathbf{R} \in \mathbb{R}^{n \times m}$ as follows: $r_{u,i} = 1$ when $\langle u, i \rangle$ interaction is observed, and $r_{u,i} = 0$ otherwise. Here the value of 1 in the matrix $\mathbf{R}$ indicates the interaction result between a user and an item, *e.g.*, whether a user has watched or rated a movie.

We frame our recommendation task in the setting of heterogeneous information network, which can be defined as follows:

Definition 1. ***Heterogeneous Information Network*** *[30]. A HIN is defined as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with an entity type mapping function $\phi : \mathcal{V} \to \mathcal{A}$ and a link type mapping function $\varphi : \mathcal{E} \to \mathcal{R}$. $\mathcal{A}$ and $\mathcal{R}$ denote the sets of predefined entity and link types, where $|\mathcal{A}| + |\mathcal{R}| > 2$.*

In HIN, **network schema** is proposed to describe the meta structure of a network, which illustrates the object types and their interaction relations.

Example 1. *Fig. 1(a) illustrates a HIN example and its corresponding network schema for movie recommender system. We can see that the network consists of multiple types of objects (*e.g.*, User (U), Movie (M), Director (D)) and their semantic relations (*e.g.*, watching relation between users and movies, friend relation among users, and directed by relation between movies and directors).*

In HIN, two objects can be connected via different semantic paths, which are defined as meta-paths.

Definition 2. ***Meta-path*** *[30]. A meta-path ρ is defined as a path in the form of $\mathcal{A}_1 \xrightarrow{\mathcal{R}_1} \mathcal{A}_2 \xrightarrow{\mathcal{R}_2} \cdots \xrightarrow{\mathcal{R}_l} \mathcal{A}_{l+1}$ (abbreviated as $\mathcal{A}_1\mathcal{A}_2 \cdots \mathcal{A}_{l+1}$), which describes a composite relation $\mathcal{R}_1 \circ \mathcal{R}_2 \circ \cdots \circ \mathcal{R}_l$ between object $\mathcal{A}_1$ and $\mathcal{A}_{l+1}$, where "$\circ$" denotes the composition operator on relations.*

Giving a meta-path ρ, there exist multiple specific paths under the meta-path, which is called a **path instance** denoted by p. As we have illustrated above, the implicit feedback matrix $\mathbf{R}$ indicates the interaction result between a user and an item. We are particularly interested in the meta-paths that connect a user and an item in HIN, which can reveal semantic context for a user-item interaction.

Definition 3. ***Meta-path based Context****. Giving a user u and an item i, the meta-path based context is defined as the aggregate set of path instances under the considered meta-paths connecting the two nodes on the HIN.*

Example 2. *Take Fig. 1(a) as an example. The user u_1 and movie m_2 can be connected via multiple meta-paths,* e.g., *"u_1-m_1-u_3-m_2" ($UMUM$) and "u_1-m_1-t_1-m_2" ($UMTM$), which constitute the context of the interaction $\langle u_1, m_2 \rangle$. Different meta-paths usually convey different interaction semantics of $\langle u_1, m_2 \rangle$. For example, $UMUM$ and $UMTM$ paths indicate that user u_1 has watched movie m_2 since (1) a user u_2 sharing the same watching records (*i.e.*, m_1) has watched m_2 and (2) she/he has previously watched movie m_1 with the same type of movie m_2 respectively. These meta-path based contexts reveal varying interaction semantics through aggregating different meth-paths.*

Given the above preliminaries, we are ready to define our task.

Definition 4. ***HIN based Top-N Recommendation****. Given a heterogeneous information network $\mathcal{G}$ with user implicit feedback matrix $\mathbf{R}$, for each user $u \in \mathcal{U}$, we aim to recommend a ranked list of items that are of interest to u.*

Many efforts have been made for HIN based recommendation. While, most of these works focus on the rating prediction task, which predicts the absolute preference score of a user to a new item [29, 41]. Top-N recommendation task is more common in practice, since implicit feedback is easier to obtain.

4 THE PROPOSED MODEL

In this section, we present the proposed model that leverages Meta-path based Context for RECommendation, called **MCRec**.

4.1 Model Overview

Different from existing HIN based recommendation models, which only learn the representations for users and items, we explicitly incorporate meta-paths as the context in an interaction between a user and an item. Instead of modeling the two-way interaction $\langle user, item \rangle$, we aim to characterize a three-way interaction $\langle user,$

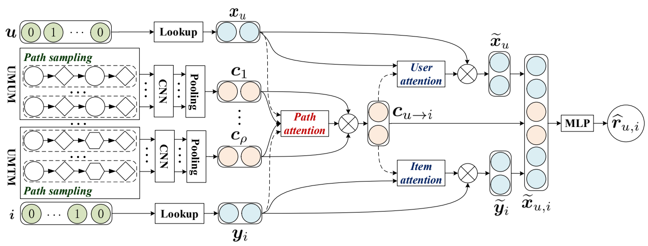


Figure 2: The overall architecture of the proposed model.

meta-paths, item⟩. For learning a better interaction function that generates the recommendations, we learn the representations (*i.e.,* embedding) for users, items and their interaction contexts, which is the aggregated meta-paths. We present the overall architecture for the proposed model in Fig. 2. As we can see, besides the components for learning user and item embeddings, the most important part lies in the embedding of *meta-path based context*. The meta-path based context is first modeled into a low-dimensional embedding using a hierarchical neural network. With the initially learned embeddings for users, items and meta-path based context, the co-attention mechanism further improves the three representations through alternative enhancement. Due to the incorporation of meta-path based context, our model is expected to yield a better performance and also improve the interpretability for recommendation results.

4.2 User and Item Embedding

Following [11], we set up a lookup layer to transform the one-hot representations of users and items into low-dimensional dense vectors, called *embeddings*. Formally, given a user-item pair $\langle u, i\rangle$, let $\mathbf{p}_u \in \mathbb{R}^{|\mathcal{U}|\times 1}$ and $\mathbf{q}_i \in \mathbb{R}^{|\mathcal{I}|\times 1}$ denote their one-hot representations. The lookup layers correspond to two parameter matrices $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}|\times d}$ and $\mathbf{Q} \in \mathbb{R}^{|\mathcal{I}|\times d}$, which store the latent factors for users and items respectively. Here d is the dimension size of user and item embeddings, and $|\mathcal{U}|$ and $|\mathcal{I}|$ are the total number of users and items respectively. The lookup operation is implemented as follows:

$$\mathbf{x}_u = \mathbf{P}^\top \cdot \mathbf{p}_u, \quad (1)$$

$$\mathbf{y}_i = \mathbf{Q}^\top \cdot \mathbf{q}_i. \quad (2)$$

4.3 Characterizing Meta-path based Context for Interaction

A major novelty of our work is to explicitly characterize meta-path based context for improving the modeling of the interaction. In this part, we first study how to generate high-quality path instances, and then present how to learn effective representations (*i.e.,* embeddings) for meta-path based context.

4.3.1 Sampling Path Instances via Priority based Random Walk based on Meta-paths. Existing HIN embedding models mainly adopt a meta-path guided random walk strategy to generate path instances [4], relying on a uniform sampling over the out-going nodes. However, the path instances generated by such a simple random walk strategy are usually of low quality and even accompanied with much noise, which makes the sampling strategy unsuitable for recommendation. Intuitively, at each step, the walker should wander to a neighbor of a higher "priority" score with a larger probability, since such an out-going node can reflect more reliable semantics by forming a more close link. Hence, the key problem is how to define the priority score of an out-going node. Inspired by the tricks for training neural networks [11, 12], we propose to use a similar *pretrain* technique to measure the priority degree of each candidate out-going nodes. The basic idea is to first learn a latent vector for each node with historical user-item interaction records by using traditional matrix factorization method without meta-path information. Then, we can measure the priority degree by the similarity between the current node and candidate out-going nodes. Such a priority score directly reflects the association degree between two nodes. To pretrain the latent factors for all the entities in HIN, we train the feature based matrix factorization framework SVDFeature [3], adapted to implicit feedback with the pairwise loss in [23], on all the available historical interaction records. SVDFeature is able to characterize various kinds of context information. We can incorporate the entities from HIN related to an interaction as the context of a training instance. With the learned latent factors, we can compute the pairwise similarities between two consecutive nodes along a path instance, and then average these similarities for ranking the candidate path instances. Finally, given a meta-path, we only keep top K path instances with the highest average similarities.

4.3.2 Meta-path based Context Embedding. After obtaining path instances from multiple meta-paths, we continue to study how to model these meta-path based context as an informative embedding. Our method naturally follows a hierarchical structure: embedding a single path instance → embedding a single meta-path → embedding the aggregate meta-paths.

Path Instance Embedding. A path instance is essentially a sequence of entity nodes. To embed such a node sequence into a low-dimensional vector, many methods can apply. Here, we adopt the commonly used Convolution Neural Network (CNN) to deal with sequences of variable lengths. Formally, given a path p from some meta-path ρ, let $\mathbf{X}^p \in \mathbb{R}^{L\times d}$ denote the embedding matrix formed by concatenating node embeddings, where L is the length of the path instance and d is the embedding dimension for entities. The structure of CNN consists of a convolution layer (producing new features with the convolution operation) and a max pooling layer. We learn the embedding of a path instance p using CNN as follows:

$$\mathbf{h}_p = CNN(\mathbf{X}^p; \Theta) \quad (3)$$

where $\mathbf{X}^p$ denotes the matrix of the path instance p and Θ denotes all the related parameters in CNNs.

Meta-path Embedding. Since a meta-path can produce multiple path instances, we further apply the max pooling operation to derive the embedding for a meta-path. Let $\{\mathbf{h}_p\}_{p=1}^K$ denote the embeddings for the K selected path instances from meta-path ρ. The embedding $\mathbf{c}_\rho$ for meta-path ρ can be given

$$\mathbf{c}_\rho = \text{max-pooling}(\{\mathbf{h}_p\}_{p=1}^K). \quad (4)$$

Our max pooling operation is carried out over K path instance embeddings, which aim to capture the important dimension features from multiple path instances.

Simple Average Embedding for Meta-path based Context. Finally, we apply the average pooling operation to derive the embedding for modeling the aggregate meta-path based context

$$\mathbf{c}_{u\to i} = \frac{1}{|\mathcal{M}_{u\to i}|} \sum_{\rho \in \mathcal{M}_{u\to i}} \mathbf{c}_{\rho}, \tag{5}$$

where $\mathbf{c}_{u\to i}$ is the embedding for meta-path based context and $\mathcal{M}_{u\to i}$ is the set of the considered meta-paths for the current interaction. In this naive embedding method, each meta-path indeed receives equal attention, and the representation of meta-path based context fully depends on the generated path instances. It does not incorporate the involved user and item into consideration, which lacks the ability of capturing varying semantics from meta-paths in different interaction scenarios.

4.4 Improving Embeddings for Interaction via Co-Attention Mechanism

Intuitively, different users are likely to have different preferences over the meta-paths. Even for the same user, a meta-path may have varying semantics in her multiple interactions with different items. A good embedding method for modeling meta-path based context should be interaction-specific, which is able to provide highly discriminative semantics in various complicated recommendation scenarios. Furthermore, in a user-item interaction, meta-paths provide important context information, and hence the involved user and item are likely to be affected by such interaction context, too. Based on these discussions, if we could improve the embeddings of users, items and meta-paths in a mutual enhancement way, it may be possible to develop a more effective representation learning method. Inspired by the recent progress of attention mechanism made in computer vision and natural language processing [21, 37], we propose a novel co-attention mechanism to achieve this goal.

4.4.1 Attention for Meta-path based Context. Since different meta-paths may have different semantics in an interaction, we learn the interaction-specific attention weights over meta-paths conditioned on the involved user and item. Given the user embedding $\mathbf{x}_u$, item embedding $\mathbf{y}_i$, the context embedding $\mathbf{c}_\rho$ for a meta-path ρ, we adopt a two-layer architecture to implement the attention

$$\boldsymbol{\alpha}^{(1)}_{u,i,\rho} = f(\mathbf{W}^{(1)}_u \mathbf{x}_u + \mathbf{W}^{(1)}_i \mathbf{y}_i + \mathbf{W}^{(1)}_\rho \mathbf{c}_\rho + \mathbf{b}^{(1)}), \tag{6}$$

$$\alpha^{(2)}_{u,i,\rho} = f(\mathbf{w}^{(2)\top} \boldsymbol{\alpha}^{(1)}_{u,i,\rho} + b^{(2)}), \tag{7}$$

where $\mathbf{W}^{(1)}_*$ and $\mathbf{b}^{(1)}$ denote the weight matrix and the bias vector for the first layer, and the $\mathbf{w}^{(2)}$ and $b^{(2)}$ denote the weight vector and the bias for the second layer. $f(\cdot)$ is set to the ReLU function.

The final meta-path weights are obtained by normalizing the above attentive scores over all the meta-paths using the softmax function,

$$\alpha_{u,i,\rho} = \frac{\exp(\alpha^{(2)}_{u,i,\rho})}{\sum_{\rho' \in \mathcal{M}_{u\to i}} \exp(\alpha^{(2)}_{u,i,\rho'})}. \tag{8}$$

which can be interpreted as the contribution of the meta-path ρ to the interaction between u and i. After we obtain the meta-path attention scores $\alpha_{u,i,\rho}$, the new embedding for aggregate meta-path context can be given as the following weighted sum:

$$\mathbf{c}_{u\to i} = \sum_{\rho \in \mathcal{M}_{u\to i}} \alpha_{u,i,\rho} \cdot \mathbf{c}_\rho, \tag{9}$$

where $\mathbf{c}_\rho$ the learned embedding for the meta-path ρ in Eq. 4. Since the attention weights $\{\alpha_{u,i,\rho}\}$ are generated for each interaction, they are interaction-specific and able to capture varying interaction context.

4.4.2 Attention for Users and Items. Given a user and an item, the meta-path connecting them provide important interaction context, which is likely to affect the original representations of users and items. Giving original user and item latent embeddings $\mathbf{x}_u$ and $\mathbf{y}_i$, and the meta-path based context embedding $\mathbf{c}_{u\to i}$ for the interaction between u and i, we use a single-layer network to compute the attention vectors $\boldsymbol{\beta}_u$ and $\boldsymbol{\beta}_i$ for user u and item i as,

$$\boldsymbol{\beta}_u = f(\mathbf{W}_u \mathbf{x}_u + \mathbf{W}_{u\to i} \mathbf{c}_{u\to i} + \mathbf{b}_u), \tag{10}$$

$$\boldsymbol{\beta}_i = f(\mathbf{W}'_i \mathbf{y}_i + \mathbf{W}'_{u\to i} \mathbf{c}_{u\to i} + \mathbf{b}'_i), \tag{11}$$

where $\mathbf{W}_*$ and $\mathbf{b}_u$ denote the weight matrix and bias vector for user attention layer, $\mathbf{W}'_*$ and $\mathbf{b}'_i$ denote the weight matrix and bias vector for item attention layer. Similarly, $f(\cdot)$ is set to the ReLU function. Then, the final representations of user and item are computed by using an element-wise product "$\odot$" with the attention vectors

$$\tilde{\mathbf{x}}_u = \boldsymbol{\beta}_u \odot \mathbf{x}_u, \tag{12}$$

$$\tilde{\mathbf{y}}_i = \boldsymbol{\beta}_i \odot \mathbf{y}_i. \tag{13}$$

The attention vectors $\boldsymbol{\beta}_u$ and $\boldsymbol{\beta}_i$ are used for improving the original user and item embeddings conditioned on the calibrated meta-path based context $\mathbf{c}_{u\to i}$ (Eq. 9).

By combining the two parts of attention components, our model improves the original representations for users, items and meta-path based context in a mutual enhancement way. We call such an attention mechanism *Co-Attention*. To our knowledge, few HIN based recommendation methods are able to learn explicit representations for meta-paths, especially in an interaction-specific way.

4.5 The Complete Model

Until now, given an interaction between user u and item i, we have the embeddings for user u, item i and the meta-path connecting them. We combine the three embedding vectors into a unified representation of the current interaction as below:

$$\widetilde{\mathbf{x}}_{u,i} = \tilde{\mathbf{x}}_u \oplus \mathbf{c}_{u\to i} \oplus \tilde{\mathbf{y}}_i, \tag{14}$$

where "$\oplus$" denotes the vector concatenation operation, $\mathbf{c}_{u\to i}$ (Eq. 9) denotes the embedding of the meta-path based context for $\langle u, i\rangle$, $\tilde{\mathbf{x}}_u$ (Eq. 12) and $\tilde{\mathbf{y}}_i$ (Eq. 13) denote the improved embeddings of user u and item i respectively. $\widetilde{\mathbf{x}}_{u,i}$ encodes the information of an interaction from three aspects: the involved user, the involved item and the corresponding meta-path based context. Following [11], we feed $\widetilde{\mathbf{x}}_{u,i}$ into a MLP component in order to implement a nonlinear function for modeling complicated interactions

$$\hat{r}_{u,i} = \mathrm{MLP}(\widetilde{\mathbf{x}}_{u,i}), \tag{15}$$

where the MLP component is implemented with two hidden layers with ReLU as the activation function and an output layer with the sigmoid function. With the premise that neural network models can learn more abstractive features of data via using a small number of hidden units for higher layers [9], we empirically implement a tower structure for the MLP component, halving the layer size for each successive higher layer.

Defining a proper objective function for model optimization is a key step for learning a good recommendation model. Traditional point-wise recommendation models for the rating prediction task usually adopt the squared error loss [16]. While, in our task, we have only implicit feedback available. Following [11, 31], we learn the parameters of our model with negative sampling and the objective for an interaction $\langle u, i \rangle$ can be formulated as follows:

$$\ell_{u,i} = -\log \hat{r}_{u,i} - E_{j \sim P_{neg}}[\log(1 - \hat{r}_{u,j})], \tag{16}$$

where the first term models the observed interaction, and the second term models the negative feedback drawn from the noise distribution P_{neg}. In this paper, we set the distribution P_{neg} as uniform distribution, which is flexible to extend to other biased distributions, *e.g.,* popularity based distribution.

As shown in Eq. 14, MCRec is a flexible approach to leveraging heterogeneous information. If we remove the term $\mathbf{c}_{u \to i}$ from Eq. 14, MCRec degrades into the neural collaborative filtering model without HIN based information, *i.e.,* NeuMF [11]. Our approach is general to work with any methods that are able to learn $\mathbf{c}_{u \to i}$. For a recommendation method, we are more concerned about the online prediction time than the offline training time. Compared with NeuMF, the additional online-recommendation time cost from our approach mainly lies in the step of generating path instances, since it is relatively cheap to compute path embeddings using CNN and max pooling operations which is easy to be accelerated in parallel. For efficiency consideration, we use a small number l of meta-paths. With the network schema, we can offline obtain all the possible link strengths based on the training data using the pretrain technique in Section 4.3.1. Given a node, we offline filter out all the out-going nodes with a low priority score, and further build an alias table [17] for $O(1)$-time node sampling. In this way, generating path instances for an interaction can be done in $O(l \cdot L \cdot N + l \cdot N \cdot \log N)$ online time, where L is the path length and N is the maximum number of considered path instances for a meta-path. In practice, we require $l \leq 5$, $L \leq 5$ and $N \leq 1000$, which costs a small computation time and is efficient for online prediction.

5 EXPERIMENTS

In this section, we evaluate the effectiveness of the proposed model for top-N recommendation.

Table 1: Statistics of the three datasets. The first row of each dataset corresponds to the number of users, items and interactions.

Datasets	Relations (A-B)	#A	#B	#A-B
Movielens	*User-Movie*	943	1,682	100,000
	User-User	943	943	47,150
	Movie-Movie	1,682	1,682	82,798
	Movie-Genre	1,682	18	2861
LastFM	*User-Artist*	1,892	17,632	92,834
	User-User	1,892	1,892	18,802
	Artist-Artist	17,632	17,632	153,399
	Artist-Tag	17,632	11,945	184,941
Yelp	*User-Business*	16,239	14,284	198,397
	User-User	16,239	16,239	158,590
	Business-City (Ci)	14,267	47	14,267
	Business-Category (Ca)	14,180	511	40,009

5.1 Experimental Setup

Evaluation Datasets. We adopt three widely used datasets from different domains, namely Movielens [1] movie dataset, LastFM [2] music dataset and Yelp [3] business dataset. LastFM dataset contains listening records of users, which can be directly transformed into implicit feedback. For the other two datasets, we follow [11, 32] to treat a rating as an interaction record, indicating whether a user has rated an item. The detailed descriptions of the three datasets are shown in Table 1. The first row of each dataset corresponds to the numbers of users, items and interactions, while the other rows correspond to the statistics of other relations. We also report the selected meta-paths for each dataset in the last column of Table 2. We only select short meta-paths of at most four steps, since long meta-paths are likely to introduce noisy semantics [30].

Evaluation Protocol and Metrics. To evaluate the recommendation performance, we randomly split the entire user implicit feedback records of each dataset into training and test set, *i.e.,* we use 80% feedback records to predict the remaining 20% feedback records [4]. Since it is time consuming to rank all items for each user in the evaluation procedure, following the strategy in [11], for each positive item in the test set, we randomly sample 50 negative samples that have no interaction records with the target user. Then, we rank the list consisting of the positive item and 50 negative items. The top-N recommendation task usually adopts similar evaluation metrics in information retrieval. Following [11, 40], we use Precision at rank K (Prec@K), Recall at rank K (Recall@K) and Normalized Discounted Cumulative Gain at rank K (NDCG@K) as the evaluation metrics. The final results are first averaged over all the test items of a user and then averaged over all the users. For stability, we perform ten runs using different random-splitting training/test sets and report the average results.

Comparison Methods. We consider two kinds of representative recommendation methods: CF based methods (ItemKNN, BPR, MF,

[1] https://grouplens.org/datasets/movielens/
[2] https://www.last.fm
[3] http://www.yelp.com/dataset-challenge
[4] We hold out 10% training data as the validation set for parameter tuning.

Table 2: The selected meta-paths used in each dataset.

Dataset	Meta-paths
Movielens	UMUM, UMGM, UUUM, UMMM
LastFM	UATA, UAUA, UUUA, UUA
Yelp	UBUB, UBCaB, UUB, UBCiB

and NeuMF) only utilizing implicit feedback, and HIN based methods utilizing rich heterogeneous information (SVDFeature$_{hete}$, SVDFeature$_{mp}$, HeteRS and FMG$_{rank}$). To examine the effectiveness of our priority based sampling strategy and co-attention mechanism, we prepare three variants of MCRec (MCRec$_{rand}$, MCRec$_{avg}$ and MCRec$_{mp}$). The comparison methods are given below:

• **ItemKNN** [25]: It is a classic collaborative filtering method by recommending similar items based on the adopted items previously. We adapt it for implicit feedback data by following the setting of [13].

• **BPR** [23]: It is the Bayesian Personalized Ranking model that minimizes the pairwise ranking loss for implicit feedback.

• **MF** [16]: It is the standard matrix factorization method, and we modify its optimization loss with the cross entropy loss [11] for top-N recommendation.

• **NeuMF** [11]: It is the state-of-the-art neural network method for top-N recommendation using only implicit feedback, which consists of a generalized MF component and a MLP component.

• **SVDFeature$_{hete}$**: SVDFeature [3] is a feature based matrix factorization model. Here we extract heterogeneous relations as one-hot features to feed into SVDFeature.

• **SVDFeature$_{mp}$**: It is another variant of SVDFeature, which utilizes the HIN embedding method meta-path2vec++ [4] to extract user and item embeddings as features for SVDFeature.

• **HeteRS** [20]: It is a strong HIN based ranking method, which employs multivariate Markov chains to model user preferences.

• **FMG$_{rank}$**: FMG [41] is a state-of-the-art HIN based model for rating prediction. We modify its optimization objective as pairwise ranking loss as in BPR [23] for top-N recommendation.

•**MCRec$_{rand}$**: It is a variant of MCRec, which employs the random meta-path guided sampling strategy [4] for path generation.

• **MCRec$_{avg}$**: It is a variant of MCRec, which employs the naive context embedding strategy (see Eq. 5) for meta-paths.

• **MCRec$_{mp}$**: It is another variant of MCRec which only reserves the attention components for meta-paths and removes the attention component for users and items.

• **MCRec**: It is our complete model.

Implementation Details. We implement the MCRec model using the python library of Keras [5]. For our model, we randomly initialize model parameters with Gaussian distribution and optimize the model with Adaptive Moment Estimation (Adam) [14], and set the batch size to 256, the learning rate to 0.001, the regularization parameter to 0.0001, the CNN filter size to 3, the dimension of user and item embeddings to 128, and the dimension of predictive factors to 32. In addition, the number of sampled path instances is 5, and the meta-paths in Table 2 are used for those HIN based methods. For MF and NeuMF, we follow the optimal configuration and architecture reported in [11]. For the other comparison methods, we optimize their parameters using 10% training data as the validation set. All the experiments are conducted on a machine with two GPUs (NVIDIA GTX-180 * 2) and two CPUs (Intel Xeon E5-2690 * 2).

[5]https://keras.io/

5.2 Experimental Results

The comparison results of our proposed model and baselines on three datasets are reported in Table 3. The major findings from the experimental results are summarized as follows:

(1) Our complete model MCRec is consistently better than all the baselines on the three datasets. The results indicate the effectiveness of MCRec on the task of top-N recommendation, which has adopted a more principled way to leverage heterogeneous context information for improving recommendation performance.

(2) Considering the three variants of MCRec, we can find that the overall performance order is as follows: MCRec > MCRec$_{mp}$ > MCRec$_{avg}$ > MCRec$_{rand}$. The results show that the co-attention mechanism is able to better utilize the meta-path based context for recommendation in two aspects. First, the importance of each meta-path should depend on a specific interaction instead of being treated equal (*i.e.,* MCRec$_{avg}$). Second, meta-paths provide important context for the interaction between users and items, which has a potential influence on the learned representations of users and items. Ignoring such influence may not be able to achieve the optimal performance for utilizing meta-path based context information (*i.e.,* MCRec$_{mp}$). In addition, although MCRec$_{rand}$ achieves competitive performance compared to baselines, it is still significantly worse than the complete MCRec. Recall our complete model adopts the priority based sampling strategy to generate path instances, while MCRec$_{rand}$ adopts a random sampling strategy. It indicates the effectiveness of the proposed sampling strategy in Section 4.3.1.

(3) Among the two kinds of baselines, most of HIN based methods (SVDFeature$_{hete}$, SVDFeature$_{mp}$ and FMG$_{rank}$) outperform CF methods (ItemKNN, BPR and MF) in most cases, which indicates the usefulness of heterogeneous information. It is noteworthy that NeuMF model works well among these baselines. An intuitive explanation is that NeuMF utilizes the multi-layer perceptron to model the complex interaction between users and items, which implies the excellence of deep neural network in capturing complex interaction relations for recommendation.

(4) Among HIN based baselines, the recently proposed method FMG$_{rank}$ overall performs best, which tries to learn effective features from the similarity evidence reflected in meta-graphs. As the two variants of SVDFeature, it can seen that SVDFeature$_{mp}$ method does not work very well, although it takes the embeddings learned from metapath2vec [4] as features. A possible reason is that the learned embeddings from metapath2vec mainly reflects the structural node features of HIN, rather than path features, and may not be directly useful for the recommendation task.

5.3 Detailed Analysis of The Proposed Model

In this part, we perform a series of detailed analysis to better understand the traits of MCRec.

5.3.1 Qualitative Analysis for the Recommendation Interpretability. A major contribution of MCRec is the incorporation of co-attention mechanism, which takes the interaction relation into

Table 3: Results of effectiveness experiments on three datasets. We use "*" to mark the best performance from the baselines for each comparison. We use "#" to indicate the improvement of MCRec over the best performance from the baselines is significant based on paired t-test at the significance level of 0.01. Our performance improvement is more significant with NDCG@10.

Model	Movielens			LastFM			Yelp		
	Prec@10	Recall@10	NDCG@10	Prec@10	Recall@10	NDCG@10	Prec@10	Recall@10	NDCG@10
ItemKNN	0.2578	0.1536	0.5692	0.4160	0.4513	0.7981	0.1386	0.5421	0.5378
BRP	0.3010	0.1946	0.6459	0.4129	0.4492	0.8099	0.1474	0.5504	0.5549
MF	0.3247	0.2053	0.6511	0.4364	0.4634	0.7921	0.1503	0.5350	0.5322
NeuMF	0.3293*	0.2090	0.6587	0.4540	0.4678	0.8104	0.1504	0.5857	0.5713
SVDFeature$_{hete}$	0.3171	0.2021	0.6445	0.4576	0.4841	0.8290*	0.1404	0.5613	0.5289
SVDFeature$_{mp}$	0.3109	0.1929	0.6536	0.4391	0.4651	0.8116	0.1524	0.5932	0.5974*
HeteRS	0.2485	0.1674	0.5967	0.4276	0.4489	0.8026	0.1423	0.5613	0.5600
FMG$_{rank}$	0.3256	0.2165*	0.6682*	0.4630*	0.4916*	0.8263	0.1538*	0.5951*	0.5861
MCRec$_{rand}$	0.3223	0.2104	0.6650	0.4540	0.4795	0.8002	0.1510	0.5842	0.5718
MCRec$_{avg}$	0.3270	0.2111	0.6631	0.4645	0.4914	0.8311	0.1595	0.5933	0.6021
MCRec$_{mp}$	0.3401	0.2200	0.6828	0.4662	0.4924	0.8428	0.1655	0.6303	0.6228
MCRec	**0.3451#**	**0.2256#**	**0.6900#**	**0.4807#**	**0.5068#**	**0.8526#**	**0.1686#**	**0.6326#**	**0.6301#**

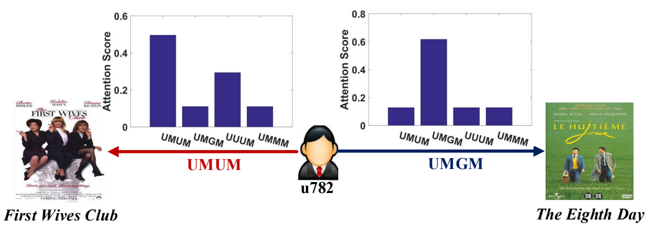


Figure 3: An illustrative example of the interpretability of interaction-specific attention distributions for MCRec.

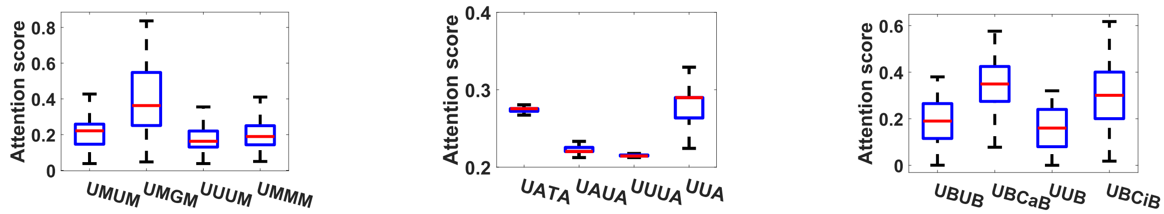
consideration in learning effective representations for recommendation. Besides the performance effectiveness, another benefit of the co-attention mechanism is that it makes the recommendation results highly interpretable. Recall that we have learned the attention weights $\{\alpha_{u,i,\rho}\}_{\rho\in\mathcal{M}}$ for an interaction between user u and item i. Since meta-paths serve as important interaction context, the attention weights provide explicit evidence to understand why an interaction happens.

To see this, we select the user $u782$ in the Movielens dataset as an illustrative example. Two interaction records of this user have been used here, namely movies of "*First Wives Club*" and "*The Eighth Day*". In Fig. 3, we can see that each interaction corresponds to a unique attention distribution, summarizing the contributions of the meta-paths. The first interaction mainly relies on the meta-paths of $UMUM$ and $UUUM$, while the second interaction mainly relies on the meta-path of $UMGM$. By inspecting into the dataset, it is found that at least five friends of $u782$ have watched the movie of "*First Wives Club*", which explains why user-oriented meta-paths $UMUM$ and $UUUM$ play the key role in the first interaction. As for the second interaction, we find that the movie of "*The Eighth Day*" is with the genre of *Drama*, which is the favorite movie genre of $u782$. This explains why genre-oriented meta-path $UMGM$ plays the key role in the second interaction. Our model is able to produce interaction-specific attention distributions, providing a good interpretability for the recommendation results.

After examining how the co-attention works at the user level, we further present the macro-level analysis of the attention distributions for the entire dataset. We present the box-plot figure of the attention weight distributions from all the interaction records of a dataset in Fig. 4. We can see that the distributions of attention weights are indeed very skew, indicating some meta-paths are more important to consider than the others.

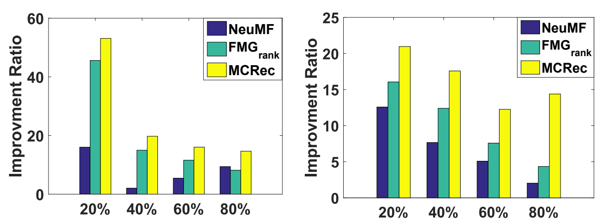
5.3.2 Cold-start Recommendation. HINs are particularly useful to alleviate cold-start problem in recommendation, where there are very few rating records but heterogeneous context information is available. We study the recommendation performance *w.r.t.* different cold-start degrees, *i.e.,* the feedback sparsity. We divide the entire dataset into five equal folds. The last fold is held out for test, and then we vary the amount of training data from one fold to four folds, corresponding to 20%, 40%, 60%, 80% data as training sets. For comparison, we select three representative baselines: BPR, NeuMF, FMG$_{rank}$. For convenience, we report the improvement ratios *w.r.t.* BPR by other comparison methods. Due to space limit, for the remainder of the paper, we will only show the results from Movielens and Yelp, since LastFM shows similar findings. As shown in Fig. 5, we can see that MCRec yields the most improvement over BPR in all the cases. Overall, the improvement ratios decrease with the increasing of training data. The results indicate that HIN based information is effective to improve the recommendation performance, and the proposed MCRec method can effectively utilize heterogeneous information in a more principled way.

5.3.3 Impact of Different Meta-Paths. In this section, we analyze the impact of different meta-paths on the recommendation performance through gradually incorporating meta-paths into the proposed model. For ease of analysis, we include the NeuMF as the reference baseline. In Fig. 6, we can observe that the performance



(a) Movielens (b) LastFM (c) Yelp

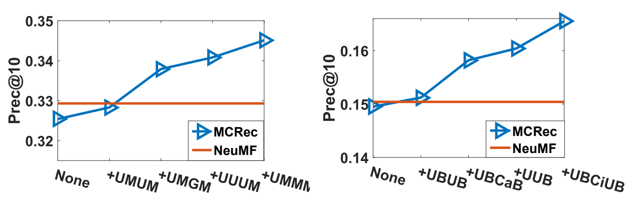
Figure 4: The distribution of attention weights of MCRec on the three datasets.



(a) Movielens (b) Yelp

Figure 5: Performance comparison of different methods for cold-start recommendation on Movielens and Yelp datasets. y**-axis denotes the improvement ratio over BPR for** $Prec@10$.

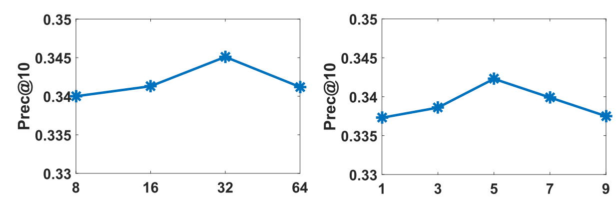
of MCRec overall improves with the incorporation of more meta-paths. Meanwhile, meta-paths seem to have different effects on the recommendation performance. Particularly, we can find that, when adding $UMGM$, MCRec has a significantly performance boost in Movielens dataset, while adding $UBCaB$ and $UBCiB$ is more important for Yelp dataset. These findings are consistent with previous observations for attention weights distributions of meta-paths in Fig. 4, where these meta-paths have higher attention scores.



(a) Movielens (b) Yelp

Figure 6: Performance change of MCRec when gradually incorporating meta-paths.

5.3.4 Parameter Tuning. Our models include a few important parameters to tune. Here, we examine the performance effect of two parameters, *i.e.,* the number of predictive factors (the embedding size for the output layer) and the number of negative samples (in Eq. 16). For the number of predictive factors, we vary it in the set of {8, 16, 32, 64}. For the number of negative samples, we vary it in the set of {1, 3, 5, 7, 9}. As shown in Fig. 7, overall MCRec is not sensitive to these two parameters. The optimal performance is obtained with 32 predictive factors and five negative samples.



(a) #predictive factors (b) #negative samples

Figure 7: Parameter tuning of MCRec on Movielens dataset.

6 CONCLUSION

In this paper, we proposed a novel deep neural network model with the co-attention mechanism for top-N recommendation in HIN. We elaborately designed a three-way neural interaction model by explicitly incorporating meta-path based context. To construct the meta-path based context, we used a priority based sampling technique to select high-quality path instances. Our model learned effective representations for users, items and meta-path based context for implementing a powerful interaction function. The co-attention mechanism mutually improved the representations for meta-path based context, users and items. Extensive experimental results have demonstrated the superiority of our model in both recommendation effectiveness and interpretability. We believe the proposed three-way neural interaction model provides a promising approach to utilize HIN information for improving recommender systems.

Currently, our approach is able to effectively select high-quality path instances, and learn the attention weights of meta-paths. While, the selection of meta-paths is completed manually. As future work, we will consider how to develop a more principled way for automatically selecting meta-paths in HINs. We will also consider adapting our approach to deal with more complicated structure patterns in HIN, *e.g.,* meta-graphs. In addition, we also will extend our model to deal with explicit feedback as well as weighted edges.

7 ACKNOWLEDGEMENT

This work is supported in part by the National Natural Science Foundation of China (No. 61772082, 61502502, 61320106006, 61375058), the National Key Research and Development Program of China (2017YFB0803304), and the Beijing Municipal Natural Science Foundation (4182043, 4162032). This work is also supported in part by NSF through grants IIS-1526499, IIS- 1763325, and CNS-1626432, and NSFC 61672313. We also thank the anonymous reviewers for their valuable suggestions for helping improve the quality of this manuscript.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2015. Context-aware recommender systems. In *Recommender systems handbook*. 191–226.

[2] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. 2017. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. 335–344.

[3] Tianqi Chen, Weinan Zhang, Qiuxia Lu, Kailong Chen, Zhao Zheng, and Yong Yu. 2012. SVDFeature: a toolkit for feature-based collaborative filtering. *Journal of Machine Learning Research* 13 (2012), 3619–3622.

[4] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 135–144.

[5] Wei Feng and Jianyong Wang. 2012. Incorporating heterogeneous information for personalized tag recommendation in social tagging systems. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1276–1284.

[6] Tao Yang Fu, Wang Chien Lee, and Zhen Lei. 2017. HIN2Vec: Explore Meta-paths in Heterogeneous Information Networks for Representation Learning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 1797–1806.

[7] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.

[8] Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. 2017. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. 221–231.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[10] Xiangnan He and Tat-Seng Chua. 2017. Neural factorization machines for sparse predictive analytics. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. 355–364.

[11] 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*. 173–182.

[12] Geoffrey E Hinton and Ruslan R Salakhutdinov. 2012. A better way to pretrain deep boltzmann machines. In *Advances in Neural Information Processing Systems*. 2447–2455.

[13] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *IEEE International Conference on Data Mining*. 263–272.

[14] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *Third International Conference on Learning Representations* (2015).

[15] Yehuda Koren and Robert Bell. 2015. Advances in collaborative filtering. In *Recommender systems handbook*. 77–118.

[16] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42 (2009).

[17] Aaron Q. Li, Amr Ahmed, Sujith Ravi, and Alexander J. Smola. 2014. Reducing the sampling complexity of topic models. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 891–900.

[18] Jian Liu, Chuan Shi, Binbin Hu, Shenghua Liu, and S Yu Philip. 2017. Personalized ranking recommendation via integrating multiple feedbacks. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. 131–143.

[19] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 701–710.

[20] Tuan-Anh Nguyen Pham, Xutao Li, Gao Cong, and Zhenjie Zhang. 2016. A general recommendation model for heterogeneous networks. *IEEE Transactions on Knowledge and Data Engineering* 28 (2016), 3140–3153.

[21] Minh C Phan, Aixin Sun, Yi Tay, Jialong Han, and Chenliang Li. 2017. NeuPL: Attention-based Semantic Matching and Pair-Linking for Entity Disambiguation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 1667–1676.

[22] Steffen Rendle. 2010. Factorization machines. In *IEEE 10th International Conference on Data Mining*. 995–1000.

[23] 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*.

[24] Royi Ronen, Elad Yom-Tov, and Gal Lavee. 2016. Recommendations meet web browsing: enhancing collaborative filtering using internet browsing logs. In *IEEE 32nd International Conference on Data Engineering*. 1230–1238.

[25] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. 285–295.

[26] Chuan Shi, Binbin Hu, Wayne Xin Zhao, and Philip S Yu. 2018. Heterogeneous Information Network Embedding for Recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2018).

[27] Chuan Shi, Xiangnan Kong, Yue Huang, S Yu Philip, and Bin Wu. 2014. Hetesim: A general framework for relevance measure in heterogeneous networks. *IEEE Transactions on Knowledge and Data Engineering* 26 (2014), 2479–2492.

[28] Chuan Shi, Yitong Li, Jiawei Zhang, Yizhou Sun, and S Yu Philip. 2017. A survey of heterogeneous information network analysis. *IEEE Transactions on Knowledge and Data Engineering* 29 (2017), 17–37.

[29] 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*. 453–462.

[30] 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. *Very Large Data Base Endowment* 4 (2011), 992–1003.

[31] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*. 1067–1077.

[32] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Latent Relational Metric Learning via Memory-based A ention for Collaborative Ranking. In *Proceedings of the 27th International Conference on World Wide Web*.

[33] Xin Wang, Wei Lu, Martin Ester, Can Wang, and Chun Chen. 2016. Social recommendation with strong and weak ties. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. 5–14.

[34] Xuejian Wang, Lantao Yu, Kan Ren, Guanyu Tao, Weinan Zhang, Yong Yu, and Jun Wang. 2017. Dynamic attention deep model for article recommendation by learning human editors' demonstration. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2051–2059.

[35] Jun Xiao, Hao Ye, Xiangnan He, Hanwang Zhang, Fei Wu, and Tat-Seng Chua. 2017. Attentional factorization machines: Learning the weight of feature interactions via attention networks. *arXiv preprint arXiv:1708.04617* (2017).

[36] Caiming Xiong, Victor Zhong, and Richard Socher. 2017. Dynamic coattention networks for question answering. *Fifth International Conference on Learning Representations* (2017).

[37] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*. 2048–2057.

[38] Linchuan Xu, Xiaokai Wei, Jiannong Cao, and Philip S Yu. 2017. Embedding of embedding (eoe): Joint embedding for coupled heterogeneous networks. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. 741–749.

[39] Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, and Ling Chen. 2013. LCARS: a location-content-aware recommender system. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 221–229.

[40] 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*. 283–292.

[41] Huan Zhao, Quanming Yao, Jianda Li, Yangqiu Song, and Dik Lun Lee. 2017. Meta-graph based recommendation fusion over heterogeneous information networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 635–644.

[42] Wayne Xin Zhao, Sui Li, Yulan He, Edward Y Chang, Ji-Rong Wen, and Xiaoming Li. 2016. Connecting social media to e-commerce: Cold-start product recommendation using microblogging information. *IEEE Transactions on Knowledge and Data Engineering* 28, 5 (2016), 1147–1159.

[43] Xin Wayne Zhao, Yanwei Guo, Yulan He, Han Jiang, Yuexin Wu, and Xiaoming Li. 2014. We know what you want to buy: a demographic-based system for product recommendation on microblogs. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1935–1944.