Unify Local and Global Information for Top-N Recommendation

Xiaoming Liu
Ministry of Education Key Lab for Intelligent Networks
and Network Security
Xi'an Jiaotong University
Xi'an, Shaanxi, China
xm.liu@stu.xjtu.edu.cn

Shaocong Wu
Ministry of Education Key Lab for Intelligent Networks
and Network Security
Xi'an Jiaotong University
Xi'an, Shaanxi, China
shaocong.wsc@stu.xjtu.edu.cn

Zhaohan Zhang
Ministry of Education Key Lab for Intelligent Networks
and Network Security
Xi'an Jiaotong University
Xi'an, Shaanxi, China
zzh1103@stu.xjtu.edu.cn

Chao Shen
Ministry of Education Key Lab for Intelligent Networks
and Network Security
Xi'an Jiaotong University
Xi'an, Shaanxi, China
chaoshen@stu.xjtu.edu.cn

ABSTRACT

Knowledge graph (KG), integrating complex information and containing rich semantics, is widely considered as side information to enhance the recommendation systems. However, most of the existing KG-based methods concentrate on encoding the structural information in the graph, without utilizing the collaborative signals in user-item interaction data, which are important for understanding user preferences. Therefore, the representations learned by these models are insufficient for representing semantic information of users and items in the recommendation environment. The combination of both kinds of data provides a good chance to solve this problem, but it faces the following challenges: i) the inner correlations in user-item interaction data are difficult to capture from one side of the user or item; ii) capturing the knowledge associations on the whole KG would introduce noises and variously influence the recommendation results; iii) the semantic gap between both kinds of data is hard to alleviate.

To tackle this research gap, we propose a novel duet representation learning framework named KADM to fuse local information (user-item interaction data) and global information (external knowledge graph) for the top-N recommendation, which is composed of two separate sub-models. One learns the local representations by discovering the inner correlations in local information with a knowledge-aware co-attention mechanism, and another learns the global representations by encoding the knowledge associations in global information with a relation-aware attention network. The two sub-models are jointly trained as part of the semantic fusion network to compute the user preferences, which discriminates the contribution of the two sub-models under the special context. We conduct experiments on two real-world datasets, and the evaluations show that KADM significantly outperforms state-of-art methods. Further ablation studies confirm that the duet architecture performs significantly better than either sub-model on the recommendation tasks.

CCS CONCEPTS

• **Information systems** → **Online advertising**; *Collaborative and social computing systems and tools*; • **Computing methodologies** → *Knowledge representation and reasoning*.

KEYWORDS

Recommendation System, Duet Representation Learning, Collaborative Signals, Knowledge Graph, Local and Global Information

1 INTRODUCTION

With the rapid development of Internet technology, the amount of online data has increased sharply. The massive information would overwhelm users so that it is time-consuming for them to filter out their favorite information among a large number of choices. To alleviate this effect, recommendation systems have become a vital and indispensable tool to assist users in making decisions.

Prior Works and Limitations. The recommendation system attracts intensive research interest and derives broad applications [29]. Conventional collaborative filtering (CF) methods [12, 50], which provide recommendation based on the user-item interaction data, have made a significant success. However, CF-based methods usually suffer from the data sparsity and cold-start issues [41] due to the fact that even the most active users just have interacted with a small percentage of items in the recommendation environment. Meanwhile, most of them are unable to thoroughly encode the collaborative signals in interaction data, because they can only capture one-side influence from users or items and model the shallow
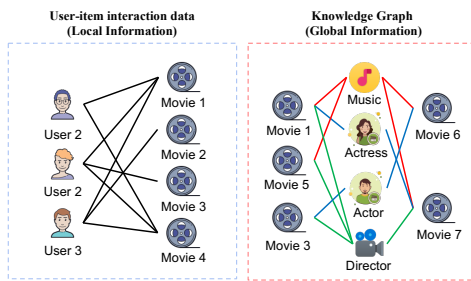
Figure 1: Illustration of local Information and global information. Local information is the user-item interaction data, which is a bipartite graph with users and items and interactions. Global information is a heterogeneous knowledge graph with multiple types of relationships and entities.

correlations between users and items. To solve these issues, many methods try to exploit different types of side information (*e.g.*, item description [5], user profile [10] and social network [16]). For instance, a number of trust-aware recommendation methods [9, 23] are proposed based on the assumption that users may share similar preferences with their trusted users.

Recently, with the development of the semantic web, introducing knowledge graphs (KGs) into recommendation systems as side information has attracted extensive attention. In contrast with other forms of side information (*e.g.*, social network), which are generally limited to capturing features with homogeneous information, KGs are heterogeneous graphs connecting various types of features related to users or items in a unified global representation space [33]. The structural information in KGs helps to explore the potential connections between users or items from different perspectives, which is beneficial for improving the performance of recommendation algorithms [3, 22, 34]. However, existing KG-based methods still share several common limitations: First, most methods focus on knowledge associations in KG without exploiting the user-item interaction data, which are insufficient to represent user preferences; Second, most studies conduct information propagation on the whole graph, which may introduce negative noise from irrelevant entities; Thirdly, most works equally treat information from different relation-paths, which is contrary to the real recommendation scenarios.

Motivations and Relations. Comprehensive analysis reveals that user-item interaction data and KG could indicate the people's decision patterns from different perspectives. Concretely, the collaborative signals in user-item interaction could introduce two kinds of effects from both item and user levels. For item level, users tend to buy items similar to their historical items, which reflects historical unchanged and independent user preferences. For another, people may be more likely to choose such item that is frequently purchased by other users together with the item they like. Moreover, there are rich knowledge associations in KG, which could indicate semantic relations between users or items from the attribute level. For example, if two movies have the same director or actor, a user who has watched one may be also willing to watch the other. Furthermore, there are multiple kinds of relation-paths in KG. The different

relation-paths would have varying degrees of impact on user preferences. For example, a user is more likely to choose movies from the perspective of movie casting rather than the director.

In this paper, we respectively define the user-item interaction data and knowledge graph as the local and global information according to their characteristics and different impacts for the recommendation, as illustrated in Fig. 1. Following the above analysis, the two-fold information can be combined and complemented each other to jointly infer a user's decision on the target item, but with the following challenges: 1) How to discover the inner correlations between users and items in local information? 2) How to eliminate the negative effect of noise triplets and model the influence of different relation-paths in global information? 3) How to alleviate the semantic gap between local and global information?

Methodologies and Results. To address the above challenges, we propose a **K**nowledge-**A**ware **D**uet **M**odel (named KADM) comprised of two sub-models to respectively formulate user preferences based on local and global information: i) Local model, which computes the local representation of each user or item by encoding the collaborative signals in local information. It firstly represents each user and item with their collaborative neighbors. Secondly, it captures the inner correlations by calculating the attention matrix based on the collaborative neighbors and further obtains the attention vectors with pooling operation. Finally, it could generate the local representations of users and items by attentively aggregating adjacent neighbors' representation that reflects users' preferences from both user and item side. ii) Global model, which learns the global representation of each user or item by capturing the rich knowledge associations in global information. It firstly extracts the enclosing subgraph of the user-item pair from global information. Secondly, it attentively aggregates the semantic information propagated by different relation-paths in subgraph to update the representations of nodes. Finally, it could generate the global representations of users and items by combining the aggregated features with their own features.

Then, since local information as well as global information could complement each other to jointly influence a user's decision on target item and the importance of them would vary for distinct user-item pairs, we utilize a gating network to dynamically weigh the two sub-models, which has been proven useful to control importance of different information flows in the network, *i.e.*, the two sub-models are jointly trained as part of the gating network.

The major contributions in this paper are summarized as follows:

- **General Concepts:** We distinguish the concepts of the user-item interaction data from KG according to their characteristics and different impacts for the recommendation. Then we respectively define them as local and global information.
- **Duet Recommendation Architecture:** We propose a novel duet architecture model for top-N recommendation, which takes advantage of both global and local views to investigate user preferences. In the duet recommendation architecture, local model takes the knowledge-aware co-attention mechanism to discover the inner correlations by encoding the collaborative signals in local information, and global model utilizes the relation-aware GNN to capture the knowledge

associations in the enclosing subgraph extracted from global information.

- **Gating network Semantic Fusion:** To eliminate the semantic gap between the two types of information, the gating network, based on a linear unit, dynamically weighs different impacts from two information according to specific contexts.
- **Outstanding Performance:** We deploy KADM on two real-world datasets. The experiment results demonstrate the state-of-the-art performance of KADM, the effectiveness of the proposed components, as well as its possible interpretability for modeling user preferences.

2 RELATED WORK

This section reviews the related works that are relevant to our work.

CF-based recommendation. Collaborative filtering (CF) is a technique widely used in recommender systems, which leverages the user-item feedback data to model the user preference. It mainly consists of neighbor-based methods [19, 29] and matrix factorization (MF) methods [20, 26]. And recently, many CF approaches are combined with new deep learning techniques [7, 48]. However, although these methods could sometimes achieve good recommendation performance, most of them still suffer from data sparsity and cold-start problems, and can only model shallow relationships between users and items.

KG-based recommendation. Knowledge graph (KG) is widely used as auxiliary information to enhance recommendation systems, and has achieved effective results. It mainly consists of three categories, *i.e.*, embedding-based methods [15, 38, 49], path-based methods [11, 14, 31] and propagation-based methods [37, 39, 51]. Among them, the propagation-based methods usually could achieve state-of-the-art results by recursively propagating the information from multi-hop nodes to refine the representation of users and items over the entire KG. For instance, Wang *et al.* [40] propose the concept of collaborative knowledge graph (CKG) to encode user behaviors and item knowledge as a unified relational graph, and further explore high-order connectivity with semantic relations in CKG for the recommendation. MVIN [35] gathers personalized knowledge information in the KG (user view) and further considers the difference among layers (entity view) to ultimately enhance item representations. CKAN [42] explicitly encodes the user-item interactions and naturally combines them with knowledge associations in an end-to-end manner.

Compared to previous methods, there are several key advantages in our proposed model: i) KADM combines the collaborative signals from the user-item interaction data with knowledge associations in a dual structure. Both kinds of information could complement each other to achieve better recommendation performance. ii) With relation-aware attention mechanism, KADM captures the various influence from different relation-paths during the information propagation process, while most of the previous methods are node-based and treat them equally. iii) KADM is based on the computation of enclosing subgraph extracted from the specific context while most methods work directly on the whole graph. It can alleviate the negative effects of irrelevant nodes in the propagation process and reduce the size of the graph to save computing resources.

Dual Mechanism. There are many dual phenomenons in real-life which inspire several dual structures in model design. For instance, Xia *et al.* [44] propose a model-level dual learning framework to merge the training of two dual tasks. DGCN [53] extends GCN to dual structures, jointly considering both the local and global consistencies in the graph. Cheng *et al.* [6] propose a novel deep latent factor model named DELF with dual embeddings of users and items for recommendation. DANSER [43] includes two dual graph attention networks to learn deep representations for social effects in recommendation systems.

Compared to previous methods, our model possesses several key differences: i) KADM designs the specific sub-model for each type of data, while others utilize sub-models of symmetric structures. ii) In the semantic fusion module, KADM dynamically weighs the importance of the representations from different sub-models, while others usually straightforwardly concatenate different features vectors.

3 PROBLEM FORMULATION

Assume that there are m users and n items, we represent historical data as the user-item interaction matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$, where $\mathbf{Y}_{ij} = 1$ indicates an observed interaction between user i and item j, *e.g.*, a user reads a book or a user clicks a news; otherwise $\mathbf{Y}_{ij} = 0$. In addition, we introduce knowledge graph $\mathcal{G} = \{(h, r, t) | h, t \in \xi, r \in \mathcal{R}\}$ as side information in the recommendation process, in which each triple (h, r, t) indicates that there is a relationship r from head entity h to tail entity t, ξ and $\mathcal{R}$ are the set of entities and relations in KG.

Input: User-item interaction matrix $\mathbf{Y}$ and knowledge graph $\mathcal{G}$.
Output: A novel duet representation learning framework for the recommendation task to correctly predict the probability $Pr(u, v)$ that user u would adopt item v.

4 METHODOLOGY

In this section, we will provide details of the proposed knowledge-aware duet model KADM, whose framework is illustrated in Fig. 2. The proposed model KADM comprises four key components for thoroughly learning the inherent characteristic of the local information and the global information. First, knowledge complement linkage module is used for mapping items to external entities in KG to explore rich semantic information; Then, local model based on a knowledge-aware co-attention mechanism and global model based on a relation-aware GNN are designed to learn the representations of the users and items from the local and global information, respectively; Finally, semantic fusion based on a gating network is developed to alleviate the semantic gap between both kinds of information, and the outputs are taken to compute the final predicted value in turn. The remaining part of this section proceeds in terms of the four components mentioned above.

4.1 Knowledge Complement Linkage

The goal of knowledge complement linkage is to link each item with its corresponding entity in KG. During this process, we first utilize entity linking [25, 32] technology to retrieve related entities with items' titles as queries. Occasionally, an item may have more than one entity returned during the linkage procedure. To address this
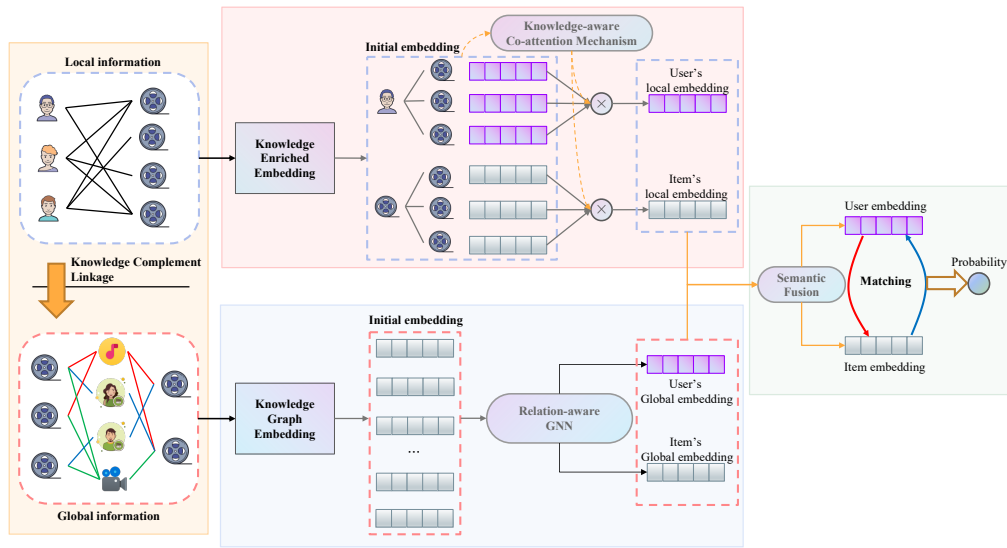
Figure 2: Illustration of the knowledge-aware duet model KADM, which compromises four components: i) Knowledge Complement Linkage (orange background, left), which maps items to external entities to capture rich semantic information in KG. ii) Local Model (red background, middle), which learns local representations of users and items from the local information. iii) Global Model (blue background, middle), which learns global representations of users and items based on the global information. iv) Prediction (green background, right), which alleviates the semantic gap between the local and global information by a gated network and calculates the final predicted probability.
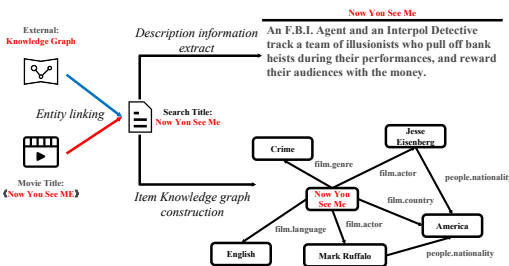

Figure 3: The process of knowledge complement linkage for each item, including entity linking, disambiguating entities, extracting items' description information and construct items' knowledge graph from external KGs.

Figure 4: Illustration of Knowledge Enriched representation (KEE), which generates the enriched representation of items with their textual descriptions.

problem, we further incorporate other items' attribute information to identify the accurate linkage entity (e.g., *IMDB ID* and *writer name* are used for the movie and the book, respectively). Based on the linked result, we can further extract the textual description of each entity and its centered subgraph as side information to enhance the recommendation process. For instance, as shown in Fig. 3, we take the title *Now You See Me* as a query to retrieve the linked entity *Now You See Me* in the external KG of a movie and further extract the textual description and subgraph based on its linked entity.

4.2 Local Representation Learning Model

The local model is proposed to explore the inner correlations between items and users based on local information. As shown in

Fig. 2, it mainly includes three components: i) the Knowledge Enriched representation (KEE), which computes the initial representations for items; ii) the encoding of users and items, which constructs the collaborative neighbor set and calculates the initial representation matrix; iii) the Knowledge-aware Co-attention Mechanism (KCM), which selects the most informative local neighbors for each user and item respectively.

Knowledge Enriched representation (KEE). Unlike the previous works [13, 49] using one-hot encoding or randomly generating representations, KEE enriches each item initial representation with its textual description and can be learned as part of local model.

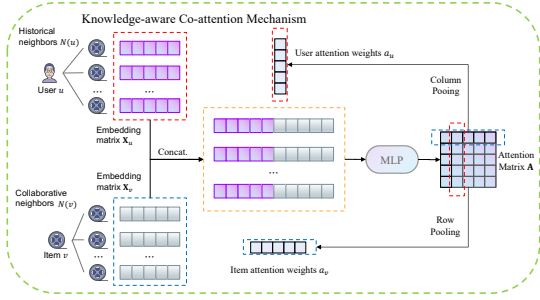Figure 5: Illustration of Knowledge-aware Co-attention Mechanism (KCM), which computes the corresponding local representation according to the historical and collaborative neighbors of users and items.
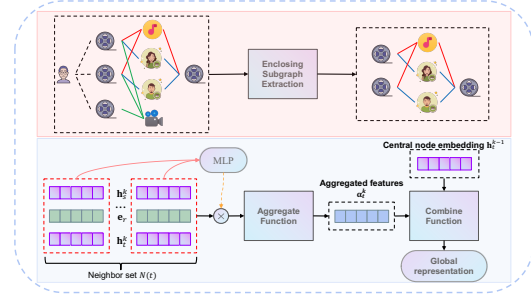

Figure 6: Illustration of Global Model, which contains three components the enclosing subgraph extraction, neural encoding of subgraph and relation-aware GNN.

As shown in Fig. 4, given the item v and its textual description $D = \{d_1, d_2, \ldots, d_m\}$ composed of word sequences, KEE firstly uses a representation layer that maps the item into a representation with a lookup table [45], then utilizes a CNN model to process the word sequence, which performs remarkably well for modeling sentence representation [17, 46]. Concretely, it embeds each word d_i into $\mathbf{d_i}$ using representation layer, composes the word representations using CNN filters, and the maximum value of each dimension is obtained through max-pooling to generate the description representation $\mathbf{D}$. And finally it is combined with the item representation $\mathbf{v}$ to learn the KEE representation $\mathbf{v}^{\text{KEE}}$ with a two-layer DNN $\mathcal{H}$. The mathematical definition of KEE is shown as follows:

$$v, d_i \xrightarrow{\mathbf{V}} \mathbf{v}, \mathbf{d}_i \qquad \textbf{representation Layer}$$
$$F_q = \mathbf{W_i}\mathbf{d}_{p:p+h} \qquad \textbf{CNN}$$
$$\mathbf{D}_i = \max(F_1, \ldots, F_q, \ldots, F_{m-h}) \qquad \textbf{Max-Pooling}$$
$$\mathbf{D} = \{\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_n\} \qquad \textbf{Description representation}$$
$$\mathbf{v}^{\text{KEE}} = \mathcal{H}(\mathbf{v} \sqcup \mathbf{D}) \qquad \textbf{KEE representation}$$

where $\mathbf{V}$ is the parameters of the representation layer, $\mathbf{W_i}$ and h are weights and size of the i-th filter, n is the number of filters, and $\sqcup$ is the concatenation operator. Furthermore, with KEE, we can also obtain the initial representation for cold-start items without interaction information using their textual description.

Encoding user and item. Rather than traditionally using an independent latent vector, we represent each user $u \in \mathcal{U}$ with a sequence of interacted items, which can be defined as $N(u) \in \mathbb{R}^{K_1 \times 1}$, where K_1 represent the size of each user's neighbor set. Due to the unbalanced neighbor distribution between users and items, we utilize a neighbor selecting approach to select the top K_1 neighbors as users' neighbor collection, which is based on mutual information [52] for ranking direct neighbors. The benefit of this approach is that it can reduce the redundancy in the neighbor set and keep the maximum retention of all neighbor features.

Analogously, items that have interacted with the same users in history can be considered similar to each other, which could be defined as collaborative neighbors. Rather than treat items independently like most existing methods, we represent each item $v \in \mathcal{V}$ with its collaborative neighbor set $N(v) \in \mathbb{R}^{K_2 \times 1}$, where K_2 represent the size of each item's neighbor set. During this process,

the similarity or relevance between two items is calculated by the common users who have interacted with them [43]. Specifically, for any item pair v_i and v_j, we define their similarity coefficient s_{ij} as the proportion of users who interact with both items. These coefficients induce an equivalence relation over items, i.e., item v_i is related to item v_j if $s_{ij} > \tau$ with τ a fixed threshold.

For encoding collaborative signals in local information, unlike most methods considering one-side influence from users or items, we respectively represented each user and item with its historical and collaborative neighbors extracted from local information. Concretely, for one user-item pair (u, v), we can represent them as $\mathbf{U} \in \mathbb{R}^{K \times 1}$ and $\mathbf{V} \in \mathbb{R}^{K \times 1}$, where K is the size of neighbor set. Following KEE, we transform each neighbor into a low-dimensional dense vector. Therefore, we encode local neighbors of user u and item v into $\mathbf{X_u} \in \mathbb{R}^{K \times d}$ and $\mathbf{X_v} \in \mathbb{R}^{K \times d}$, where d is the dimension for representation.

Knowledge-aware Co-attention Mechanism (KCM). To select the most informative local neighbors for each user and item respectively and generate more meaningful representations of users and items, we propose a knowledge-aware co-attention module, which is shown in Fig. 5. Given the local neighbors representation matrix of a user $\mathbf{X_u} \in \mathbb{R}^{K \times d}$ and an item $\mathbf{X_v} \in \mathbb{R}^{K \times d}$, we design an attention network with multiple layers to calculate an attention matrix $\mathbf{A} \in \mathbb{R}^{K \times K}$ as

$$\mathbf{A}_{ij} = \textbf{Attention}(\mathbf{X}_u^i, \mathbf{X}_v^j), \qquad (1)$$

where $\mathbf{X}_u^i$ is the i-th neighbor of user u_i, $\mathbf{X}_v^j$ is the j-th neighbor of item v_i, and $\mathbf{A}_{ij}$ is the calculated correlation value between them. $\mathbf{A}$ contains the inner correlations among different collaborative neighbors. Furthermore, we respectively calculate the attention vectors of user or item by performing the mean-pooling operation along column or row on the attention matrix $\mathbf{A}$, which comprehensively considers the influence of different neighbors and can be defined as follows,

$$a_u = \textbf{Mean-Pooling}(\{A_{i\cdot}\}_{i=1}^K),$$
$$a_v = \textbf{Mean-Pooling}(\{A_{\cdot j}\}_{j=1}^K), \qquad (2)$$

where $a^u \in \mathbb{R}^{K \times 1}$ and $a^v \in \mathbb{R}^{1 \times K}$ are the importance vectors for user u and item v. After that, we take the normalization values of importance vectors as weights to calculate the local representations

by attentively aggregating neighbor representation.

$$
\begin{aligned}
\mathbf{u}^{\text{Local}} &= \mathbf{a}_u'^{\,\mathrm{T}} \mathbf{X}_u, \ \mathbf{a}_u' = \sigma(\mathbf{a}_u), \\
\mathbf{v}^{\text{Local}} &= \mathbf{a}_v' \mathbf{X}_v, \ \mathbf{a}_v' = \sigma(\mathbf{a}_v),
\end{aligned}
\tag{3}
$$

where σ is the softmax function, $\mathbf{u}^{\text{Local}}$ and $\mathbf{v}^{\text{Local}}$ are the local representations of user u and item v, respectively.

4.3 Global Representation Learning Model

Unlike most previous methods, which ignore the different impacts from different relation paths and may bring negative noises during conducting the information propagation process on the whole KG, the proposed global model consists of three components: i) Enclosing subgraph extraction, which extracts the enclosing subgraph for the given user-item pair from knowledge graph; ii) Neural encoding of subgraph, which learns a low-dimensional representation vector for each entity and relation that preserves the structural information of the graph; iii) Relation-aware GNN, which reveals the different importance of relation-paths during information propagation on the subgraph.

Enclosing subgraph extraction. Rather than existing methods capturing the knowledge associations on the whole graph, in this work, we only take the neighborhood of particular triplet in KG into consideration to eliminate noise during propagation. To do so, we define *enclosing subgraph* as the graph induced by all the entities that occur on a path between the two target entities. For instance, $(e_0 \xrightarrow{\textit{produced by}} e_1 \xrightarrow{\textit{produce}} e_2)$ is a path included in enclosing subgraph around (e_0, e_2).

Hence, for extracting the enclosing subgraph of the given user-item pair (u, v), we first represent user u with its interacted items set $N(u)$. Secondly, we construct the entity set $E(u)$ for user u and target entity e for item v by mapping items into their corresponding entities in KG. Thirdly, for each entity $e_i \in E(u)$, we compute the enclosing subgraph for e_i and e by taking the intersection of $\mathcal{N}_k(e_i)$ and $\mathcal{N}_k(e)$, which are set of k-hop neighbors, and further pruning the entities that are isolated from either entity e_i or e. Finally, as shown in Fig. 6, we construct the enclosing graph for (u, v) by merging all the enclosing subgraphs between every entity in $E(u)$ and target entity e.

Neural encoding of subgraph. Knowledge graph representation (KGE) is an effective way to learn dense low-dimensional vector representations for entities and relations, which preserves the structural information of the graph. Recently, translation-based KGE methods have received great attention due to their concise models and superior performance [38]. Therefore, in this paper, we employ a widely used translation-based method TransR [21], which introduces a projection matrix for each relation to map entity representations to their corresponding relation space. Concretely, for each triple (h, r, t) in the graph, we define that $\mathbf{e}_h, \mathbf{e}_t \in \mathbb{R}^{d \times 1}$, $\mathbf{e}_r \in \mathbb{R}^{k \times 1}$ and $\mathbf{M}_r \in \mathbb{R}^{k \times d}$ as the representations of entities h, t, relation r and the projection matrix for relation r respectively. The representation is learned by optimizing the translation principle $\mathbf{e}_h^r + \mathbf{e}_r \approx \mathbf{e}_t^r$ under the constraints of $\mathbf{e}_h^r = \mathbf{e}_h \mathbf{M}_r$ and $\mathbf{e}_t^r = \mathbf{e}_t \mathbf{M}_r$, which is the mapping representations of $\mathbf{e}_h$ and $\mathbf{e}_t$ in the relation r's space. Then, the

plausibility score for the triplet (h, r, t) is defined as

$$
g_r(h, t) = ||\mathbf{e}_h^r + \mathbf{e}_r - \mathbf{e}_t^r||_2^2.
\tag{4}
$$

The lower the plausibility score $g_r(h, t)$, the more valid the triplet is. To encourage the discrimination between true and false triplets, we use the following margin-based ranking loss for training,

$$
\mathcal{L}_{KG} = \sum_{(h, h', r, t, t') \in \Gamma} \max(0, g_r(h, t) + \gamma - g_r(h', t')),
\tag{5}
$$

where γ is the margin, and $\Gamma = \{(h, r, t, t') | (h, r, t) \in \mathcal{G}, (h', r, t') \notin \mathcal{G}\}$ is the training set for KGE.

Following this way, we can initialize the representations of entities and relations on the granularity of triplets by exploiting the structural information in the extracted enclosing subgraph.

Relation-aware GNN for Representation Learning. The goal of global model is to compute the global representations of the user-item pair (u, v) based on the enclosing subgraph extracted from global information. We adopt the general message-passing scheme described in [47], where a node representation is iteratively updated by combining it with aggregation of its neighbors' representation. Concretely, the k-th layer of GNN is given by

$$
\begin{aligned}
\mathbf{a}_i^k &= \text{AGGREGATE}^k(\mathbf{h}_s^{k-1} : s \in \mathcal{N}(t), \mathbf{h}_t^{k-1}), \\
\mathbf{h}_t^k &= \text{COMBINE}^k(\mathbf{h}_t^{k-1}, \mathbf{a}_t^k),
\end{aligned}
$$

where $\mathcal{N}(t)$ is the set of immediate neighbors of node t, a_t^k denotes the aggregated message from the neighbors, and h_t^k denotes the latent representation of node t in the k-th layer. During the message-passing process, the initial latent representation h_i^0 of any node i is initialized by neural encoding of subgraph with KGE in previous components. Furthermore, this framework gives the flexibility to plug in different **AGGREGATE** and **COMBINE** functions resulting in various GNN architectures [36].

Inspired by the fact that message passed from different relation-path in graph may have different influence and the multi-relational R-GCN [30], we design a relation-aware GNN, as shown in Fig. 6, whose **AGGREGATE** function is defined as

$$
\mathbf{a}_t^{k+1} = \sum_{r=1}^{\mathcal{R}} \sum_{s \in \mathcal{N}_r(t)} \omega_{rst}^{k+1} \mathbf{W}_r^{k+1} \mathbf{h}_s^k,
\tag{6}
$$

where $\mathcal{R}$ is the total number of relations present in graph, $\mathcal{N}_r(t)$ is the neighbors of node t connected with relation r, $\mathbf{W}_r^{k+1}$ is the transformation matrix for relation r used to propagate messages in the k-th layer, and ω_{rst}^{k+1} is the relation-path attention weight at the k-th layer corresponding to the triplet (s, r, t), which is calculated via a two-layer MLP as follow:

$$
\begin{aligned}
\mathbf{c} &= \text{ReLU}(\mathbf{W}_1[\mathbf{h}_s^k \oplus \mathbf{h}_t^k \oplus \mathbf{e}_r] + \mathbf{b}_1), \\
\omega_{rst}^{k+1} &= \sigma(\mathbf{W}_2 \mathbf{c} + \mathbf{b}_2),
\end{aligned}
\tag{7}
$$

where $\mathbf{h}_s^k$ and $\mathbf{h}_t^k$ are the latent representation of head and tail node of a triplet at k-th layer of the GNN, $\mathbf{e}_r$ is the learned representation of relation r in graph, and σ is a sigmoid function which regulates the information aggregated from each neighbor. In practice, to avoid overfitting on rare relations caused by the rapid growth in the amount of parameters with the number of relations in the graph, we adopt the basis sharing mechanism among the relation-specific

transformation matrices $\mathbf{W}_r^{k+1}$ of each layer in GNN and also implement a form of triplet dropout, where triplets are randomly dropped from the graph while aggregating information from the neighborhood. Furthermore, given the aggregated information $\mathbf{a}_t^{k+1}$ of node t, we implement the **COMBINE** function with a self-connection of a special relation type to compute the updated representation for it, derived from [30]. It is given by

$$\mathbf{h}_t^{k+1} = \text{LeakyReLU}(\mathbf{W}_3 \mathbf{h}_t^k + \mathbf{a}_t^{k+1}). \tag{8}$$

Following the relation-aware GNN architecture as described above, we obtain the global representations of items after L layers of message passing on the corresponding subgraph. Then, we compute the global representation of user with its interacted items as follows,

$$
\begin{aligned}
\mathbf{v}^{\text{Global}} &= \mathbf{h}_v^L, \\
\mathbf{u}^{\text{Global}} &= \frac{1}{|\mathcal{N}(u)|} \sum_{v_i \in \mathcal{N}(u)} \mathbf{h}_{v_i}^L,
\end{aligned}
\tag{9}
$$

where $|\mathcal{N}(u)|$ is the size of item set, $\mathbf{v}^{\text{Global}}$ and $\mathbf{u}^{\text{Global}}$ are global representations of item v and user u, respectively.

4.4 Prediction with Semantic Fusion

Note that local and global information could jointly indicate user preference on item, but for distinct user-item pairs, the importance of the two-fold effects could be different. Therefore, inspired by [52], we design a gating network for semantic fusion to dynamically allocate weights to the four semantic features ($\mathbf{v}^{\text{Local}}$, $\mathbf{u}^{\text{Local}}$, $\mathbf{v}^{\text{Global}}$, $\mathbf{u}^{\text{Global}}$) according to specific user-item pair.

Semantic fusion with gating network. For a user-item pair (u, v), given the local and global representations of them from the above two sub-models, we could derive the final representations with the gating network as follows,

$$
\begin{aligned}
\mathbf{u}^{\text{Final}} &= \alpha \cdot \mathbf{u}^{\text{Global}} + (1 - \alpha) \cdot \mathbf{u}^{\text{Local}}, \\
\alpha &= \sigma(\mathbf{W}_{\text{gate}}[\mathbf{u}^{\text{Global}} \sqcup \mathbf{u}^{\text{Local}}]),
\end{aligned}
\tag{10}
$$

where $\mathbf{u}^{\text{Final}}$ is the final representation of user u, α denote the weight for global features and $\mathbf{W}_{\text{gate}}$ is a learnable transformation matrix of the linear unit. The final representation $\mathbf{v}^{\text{Final}}$ of item v could also be computed in similar way. Taking the final representation as input, we can compute the probability that user u will adopt item v,

$$Pr(u, v) = nn(\mathbf{u}^{\text{Final}}, \mathbf{v}^{\text{Final}}), \tag{11}$$

where $nn(\cdot)$ can be a fully-connected network with a sigmoid activation function.

Loss function. To optimize the recommendation model, we adopt the loss function of BPR loss [27], which assumes that the observed interactions that indicate more user preferences should be assigned higher predictions values than unobserved ones:

$$Loss = -\sum_{(u, v_i, v_j) \in O} -\ln \sigma(Pr(u, v_i) - Pr(u, v_j)) + \lambda ||\Theta||_2^2, \tag{12}$$

where $O = \{(u, v_i, v_j)|(u, v_i) \in \mathcal{I}^+, (u, v_j) \in \mathcal{I}^-\}$ denotes the training set, $\mathcal{I}^+$ indicates the positive interactions between user u and items while $\mathcal{I}^-$ is the sampled negative interaction set, $\sigma(\cdot)$ is the sigmoid function, and Θ is the model parameter set. L_2 regularization parameterized by λ on Θ is conducted to prevent overfitting.

To optimize the loss function, we adopt the mini-batch Adam in our implementation for its ability to adaptively control the learning rate.

4.5 Time Complexity of KADM

For the local model, the computational complexity is $O(n(m - h)hd + d(n + d))$ for KEE operation, and $O(2K(d + 1))$ for KCM operation, where n and m are the number of filters and fixed length of description content, h is the filter size in KEE, K and d are the fixed neighbor size and length of embeddings, respectively, so the computational complexity in this part is $O(n(m - h)hd + d(n + d) + 2K(d + 1))$. For the global model, the computational complexity for enclosing subgraph extraction is $O((K+1)(|\xi| + |E|))$ while utilizing Breadth-First Search algorithm to prune the isolated entities, and $O(|\mathcal{R}|dk)$ for representation learning. where $|E|$, $|\xi|$ and $|\mathcal{R}|$ are the size of edges, nodes and relations in the graph, respectively, d is the size of node/relation embeddings. It can be seen that the computational cost of the global model depends on the size of the graph, which can be greatly reduced by extracting enclosing subgraphs. In the semantic fusion part, the computation complexity of gating network is $O(d)$, and the computational complexity of the last fully-connected layer is $O(d^2)$.

5 EXPERIMENTS

To comprehensively evaluate the proposed model KADM, we conduct experiments to answer the following research questions:

RQ1. How does KADM perform compared with state-of-the-art models for recommendations, especially the KG-based recommendation models?

RQ2. How do hyper-parameters and the key components in KADM impact the recommendation performance?

RQ3. Could KADM provide some reasonable explanations about user preferences benefiting from the knowledge graph and attention mechanism?

5.1 Dataset Description

To evaluate the effectiveness of KADM, we apply our model to two public benchmark datasets Movielens and Last-FM. The statistics of these two datasets are shown in Table 1. The basic descriptions about them are summarized as follows:

- **MovieLens-1M**[1] contains approximately 1 million explicit ratings (ranging from 1 to 5) on the MovieLens website. We extract the 10-core data to ensure data quality.
- **Last.FM**[2] contains musician listening information from a set of 2 thousand users from Last.FM online music system. Similarly, we use the 10-core setting to ensure that each user and item pair has at least ten interactions.

In order to be consistent with the implicit feedback setting, we transform them into implicit feedback where each user-item pair is marked with 1 indicating that the user has rated the item positively. The threshold of positive for MovieLens-1M is 4, while no threshold is set for Last.FM due to their sparsity.

---

[1]https://grouplens.org/datasets/movielens/

[2]https://grouplens.org/datasets/hetrec-2011/

Table 1: Basic statistics of the datasets.

		MovieLens-1M	Last.FM
User-Item Feedback	#Users	6,040	1,851
	#Items	3,389	2,315
	#Interactions	997,024	59,781
Knowledge Graph	#Entities	392,966	10,367
	#Relations	49	63
	#Triplets	2,112,838	245,043

Besides the user-item interactions, we need to construct a knowledge graph for each dataset. Concretely, we follow the way in Knowledge Complement Linkage (Sec. 4.1) to map items to Freebase entities[3] via title matching. For those items that failed to link, we simply discard them. Furthermore, for identified entities, we consider the triplets that are immediate neighbors of the entities aligned with items, no matter which role (*i.e.*, subject or object) it serves as. To ensure the KG quality, we then filter out infrequent entities expected for entities aligned with items (*i.e.*, lower than 10 in both datasets) and retain the relations appearing in at least 50 triplets. The basic statistics of the extracted knowledge graph information for the two datasets are also summarized in Table 1.

For each dataset, we randomly select 80% of interaction history of each user to constitute the training set, and treat the remaining as the test set. From the training set, we randomly select 20% of interactions as validation set to tune hyper-parameters. For each observed user-item interaction, we treat it as a positive instance, and then conduct the negative sampling strategy to pair it with one negative item that the user did not rate before.

5.2 Experiment Setup

Evaluation Metrics. For each user in the test set, we randomly sample 100 items that the user has not interacted with as the negative items, considering the computational efficiency. Then each method outputs the user's preference scores over all the items in test environment. To evaluate the effectiveness of top-K recommendation and preference ranking, we adopt two widely-used evaluation protocols: recall@K and ndcg@K [4]. By default, we set K = 20. We report the average metrics for all users in the test set.

Baselines. To evaluate the effectiveness of KADM, we compare the proposed model with CF-based (FM and NFM), regularization-based (CKE and CFKG), and GNN-based (KGAT, MVIN and CKAN):

- **FM** [28]: A basic factorization method for modeling the second-order feature interactions between inputs. In our evaluations, we treat ID of a user, an item, and the related KG knowledge as input features.
- **NFM** [12]: The method is a state-of-the-art factorization model, which subsumes FM under neural network. Specially, we enrich the representation of an item with the embeddings of connected entities in KG and employ one hidden layer on input features as suggested in [12].
- **CKE** [49]: It combines CF with various information, including structural, textual, and visual knowledge in a unified

---

[3]https://developers.google.com/freebase/

framework for the recommendation. We implement CKE as CF plus structural knowledge in this paper.
- **CFKG** [1]: It applies TransE [2] on the unified graph including users, items, entities, and relations, transforming the recommendation task into the plausibility prediction of (user, Interact, item) triplets.
- **KGAT** [40]: It employs a graph attention network on a unified graph, which includes the knowledge graph and user-item graph, to discriminate the importance of neighbors in graph.
- **MVIN** [35]: It learns the representations of items from both the user view and the entity view. MVIN gathers the knowledge in KG and the different interactions between entities to model user preference.
- **CKAN** [42]: It encodes the collaborative signals that are latent in user-item interactions and combines them with KG in an end-to-end manner. CKAN initialize the entity set of user and item with the collaborative signals.

Implement Details. All models are implemented based on PyTorch, in which the hyper-parameters are configured following popular choices or previous research. In detail, we optimize all models with Adam [18] optimizer, where the batch size is fixed at 512. The default Xavier initializer [8] is used to initialize the model parameters. We apply a grid search for some common hyper-parameters in all models: the learning rate is tuned amongst {0.05, 0.01, 0.005, 0.001} with the decay rate of 0.9, the coefficient of L_2 normalization is searched in $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$, and the dropout ratio is tuned in {0.2, 0.3, 0.4, ..., 0.8}. For other hyper-parameters of baselines, the settings are the same as reported in their original papers or as default in their codes. Then for the hyper-parameters specific to our proposed model, they are set as follows: The size of neighbors is set at 40 for MovieLens-1M and 20 for Last.FM according to their distribution. For the local model, the dimension of word embeddings is 64 and is initialized with Word2vec [24] using wiki corpus. The length of the CNN with 64 filters used to encode description is set to 3, which refers to tri-gram. The fixed size of descriptions is set 30 for MovieLens-1M and 40 for Last.FM according to their distribution. For the global model, we choose TransR [21] to learn the pre-trained entity and relation embedding with the dimension of 128. In consideration of computational efficiency, we limit the number of triples where the entity is located to 1,000 for Last.FM and 100 for MovieLens-1M and preserve 2-hop neighbors during constructing the enclosing subgraph. The codes of this paper are available at https://github.com/scwu1008/KADM.

5.3 Performance Comparison (RQ1)

The experiment results for algorithm overall comparison are shown in Table 2, we have some observations from it:

- **Our proposed KADM has the best performance in all metrics on both two datasets.** Overall, KADM surpasses others significantly by around 0.05 and 0.04 on Last.FM dataset and by 0.02 and 0.03 on MovieLens-1M dataset in metrics of recall and ndcg at least, respectively. It indicates that KADM has the significant power of explicitly encoding collaborative signals with the co-attention manner and capturing the rich knowledge associations contained in global

Table 2: Comparative results of MovieLens-1M and Last.FM. For Recall, NDCG, the larger value is better.

Model	Last.FM		MovieLens-1M	
	recall	ndcg	recall	ndcg
FM	0.568	0.448	0.534	0.610
NFM	0.535	0.412	0.590	0.620
CKE	0.553	0.483	0.635	0.670
CFKG	0.577	0.484	0.621	0.672
KGAT	0.657	0.550	0.652	0.701
MVIN	0.672	0.583	0.658	0.713
CKAN	0.686	0.590	0.673	0.721
KADM	**0.736**	**0.625**	**0.694**	**0.752**

information with the relation-aware attention mechanism. From another aspect, the combination of collaborative signals in local information and knowledge associations in global information can obviously improve the recommendation performance. Meanwhile, KADM has better performance than CKAN on both datasets, which indicates that the dual mechanism is effective for modeling different kinds of information.

- **GNN-based models have better performances than other kinds of baselines, but are affected by introduced noise.** It illustrates that capturing the information propagation on KG with GNN can be effective to model user preference. However, when the graph becomes denser and larger, more noise would be introduced to the propagation process [35]. Therefore, directly computing on the enclosing subgraph between user and item in KADM, which only contains the relevant entities and relation-paths in large-scale KG, can be more effective.

- **Most KG-based methods perform better than traditional CF-based methods on all datasets.** It demonstrates that the usage of KG is of great help for the recommendation. Meanwhile, it is worth noting that the performances of the GNN-based models are better than regularized-based models, which indicates that modeling the first-order relationship might not fully utilize the structural information in KG.

- **In most situations, the model performance on movie data is better than music data.** One possible assumption is that there are more interactions between users and items in local information and links among nodes in global information, which provides sufficient information for learning the latent embeddings.

5.4 Study Of KADM (RQ2)

To study the performance variation for our model, we conduct some experiments on Last.FM with different hyper-parameter settings.

Effect of dimension of embedding. In KADM, we explore the impact of different dimensions on the model performance, including word embedding in local model and node embedding in global model. The results are shown in Fig. 7, which enlightens us that a proper embedding dimension is needed. If it is too small, it would lack expressiveness; while if it is too large, it might cause a little overfitting, which leads to performance decline.
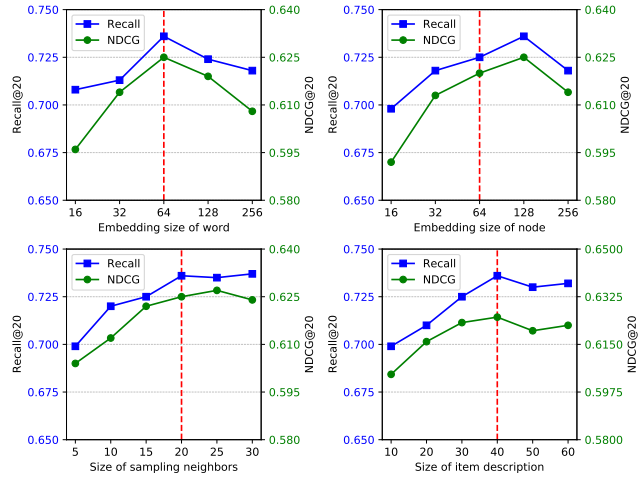

Figure 7: Evaluation of KADM on Last.FM w.r.t different hyper-parameters.

Table 3: Effect of different network configurations.

Model	Last.FM		MovieLens-1M	
	recall	ndcg	recall	ndcg
KADM-co	0.704	0.613	0.669	0.719
KADM-rel	0.695	0.607	0.662	0.714
KADM-local	0.674	0.572	0.654	0.708
KADM-global	0.689	0.603	0.663	0.716
KADM	0.736	0.625	0.694	0.752

Effect of sampling neighbor size. We vary the size of sampling neighbor to investigate the impact of usage of the local and global information, which influences the initial neighbor set of user and item in KADM. From Fig. 7, we observe that the performance would be better with the sample size increasing, but the rate of increase is decreasing and the computational cost increases as well. Therefore, we can select an appropriate sample size that can keep a good balance between model performance and complexity.

Effect of item description size. The change of description size may influence the learned initial embedding of items in local model. To investigate its impact, we experiment with different sizes of description D. From Fig. 7, it illustrates that the model performance first increases and then decreases as the length increases. A reasonable explanation is that a too small D lacks enough capacity to describe the item, while a too large D is prone to be misled by noises.

Effect of different network configurations. In order to verify the effectiveness of some components in our model, we conduct some ablation studies and the results are shown in Table 3. There are three different network configurations used for ablation experiments: 1) Co-attention mechanism, which can compute the inner correlation in local information. In KADM-co, we fix the attention weight in the local model to $\frac{1}{K}$, where K is the size of the sampling neighborhood; 2) Relation-aware attention mechanism, which models the different influence of relation paths during the information
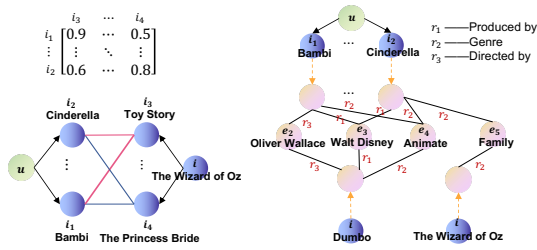
Figure 8: A real example from MovieLens-1M, including local information (left) and global information (right).

propagation process; In KADM-rel, we modify the **AGGREGATE** function in the global model to an averaging function, and take the mean of the immediate neighbors as the aggregated feature. 3) Combination of local and global models, which complement each other for modeling user preferences. We conduct experiments on the local model and the global model after removing gate network settings, separately. As we can see, the results enlighten us that: i) The attention mechanisms are both effective to help us filter useful features from local and global information, which is beneficial for modeling user preference; ii) This supports our underlying hypothesis that modeling user preference with both local and global information could complement each other to achieve a better performance, and hence a combination of them is more appropriate.

5.5 Case Study (RQ3)

Benefiting from the attention mechanism and knowledge graph, we can reason on the attention matrix in local model and the high-order connectivity of enclosing subgraph in global model to infer the user preferences on the candidate item, offering some potential explanations. Specially, we conduct a case study of one user-item pair and show the computed attention matrix and the extracted higher-order connectivity of enclosing subgraph between user u and item i in Fig. 8. There are three key observations: a) For local information, KADM computes the attention matrix from it, which is beneficial to infer the user preference. The attention weights between collaborative neighbors of user u and item i can be considered as evidence of why the item meets the user preference. As we can see, most neighbors of item i are similar to neighbors of user u, which denotes that the target item matches the user taste. b) For global information, KADM captures the high-order connectivity of enclosing subgraph extracted from it, which can also play an important role to infer user preference. The connected paths in the subgraph are useful for inferring user preference. For instance, the connectivities ($Bambi \xrightarrow{r_3} OliverWallace \xrightarrow{r_3} Dumbo$) and ($Bambi \xrightarrow{r_1} WaltDisney \xrightarrow{r_1} Dumbo$) indicate that the target item $Dumbo$ has the same director and publisher as the user's favorite movie $Bambi$ in the past. Hence, we can generate the explanation as $Dumbo$ is recommended since you have watched $Bambi$ directed by the same director $Oliver Wallace$ and produced by the same producer $Walt Disney$. c) For the whole model, the local model and the global model can complement each other. Concretely, even if the target item itself has almost no related path connected with the user in global information, we can still recommend the item through

its collaborative neighbor similarity with the user, and vice versa. For example, we would recommend *The Wizard of Oz* even though the subgraph is sparse, because the reason that its collaborative neighbors are very similar to the user's.

6 CONCLUSION AND FUTURE WORK

In this work, we study the task of combining the user-item interaction data and knowledge graph information for top-N recommendation. We propose a duet representation learning framework KADM to unify the local and global information, in which a local model with a knowledge-aware co-attention mechanism is developed to learn the local representation of items and users by discovering the inner correlations from their collaborative neighbors, and a global model with relation-aware GNN is designed to learn the global representation of items and users by capturing the knowledge associations in the enclosing subgraph from knowledge graph. Extensive experiments on two real-world datasets verify the effectiveness of KADM. Further evaluations on hyper-parameters and case studies also demonstrate the advantages of KADM.

As for future work, we would make focus on the time complexity reduction of KADM based on graph parallel computing. We also plan to fuse more heterogeneous data to enrich the recommendation performance while both local and global information are extremely sparse, or consider the condition of privacy protection or dynamical modeling problems in recommendation systems.

7 ACKNOWLEDGMENTS

REFERENCES

[1] Qingyao Ai, Vahid Azizi, Xu Chen, and Yongfeng Zhang. 2018. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms* 11, 9 (2018), 137.

[2] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems* 26 (2013).

[3] Rose Catherine and William Cohen. 2016. Personalized recommendations using knowledge graphs: A probabilistic logic programming approach. In *Proceedings of the 10th ACM conference on recommender systems*. 325–332.

[4] Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2021. Sequential Recommendation with Graph Neural Networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 378–387.

[5] Tianqi Chen, Weinan Zhang, Qiuxia Lu, Kailong Chen, Zhao Zheng, and Yong Yu. 2012. SVDFeature: a toolkit for feature-based collaborative filtering. *The Journal of Machine Learning Research* 13, 1 (2012), 3619–3622.

[6] Weiyu Cheng, Yanyan Shen, Yanmin Zhu, and Linpeng Huang. 2018. DELF: A Dual-Embedding based Deep Latent Factor Model for Recommendation.. In *IJCAI*, Vol. 18. 3329–3335.

[7] Gintare Karolina Dziugaite and Daniel M Roy. 2015. Neural network matrix factorization. *arXiv preprint arXiv:1511.06443* (2015).

[8] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international*

conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 249–256.

[9] Guibing Guo, Jie Zhang, and Daniel Thalmann. 2012. A simple but effective method to incorporate trusted neighbors in recommender systems. In *International conference on user modeling, adaptation, and personalization*. Springer, 114–125.

[10] Qing Guo, Zhu Sun, and Yin-Leng Theng. 2019. Exploiting side information for recommendation. In *International Conference on Web Engineering*. Springer, 569–573.

[11] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2020. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering* (2020).

[12] 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.

[13] Binbin Hu, Chuan Shi, Wayne Xin Zhao, and Tianchi Yang. 2018. Local and global information fusion for top-n recommendation in heterogeneous information network. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 1683–1686.

[14] 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*. 1531–1540.

[15] 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*. 505–514.

[16] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*. 135–142.

[17] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, 1746–1751. https://doi.org/10.3115/v1/D14-1181

[18] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[19] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 426–434.

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

[21] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. (2015), 2181–2187.

[22] Chen Luo, Wei Pang, Zhe Wang, and Chenghua Lin. 2014. Hete-cf: Social-based collaborative filtering recommendation using heterogeneous relations. In *2014 IEEE International Conference on Data Mining*. IEEE, 917–922.

[23] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*. 287–296.

[24] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[25] David Milne and Ian H Witten. 2008. Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*. 509–518.

[26] Andriy Mnih and Russ R Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in neural information processing systems*. 1257–1264.

[27] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[28] Steffen Rendle, Zeno Gantner, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2011. Fast context-aware recommendations with factorization machines. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 635–644.

[29] 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.

[30] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*. Springer, 593–607.

[31] 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.

[32] Avirup Sil and Alexander Yates. 2013. Re-ranking for joint named-entity recognition and linking. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 2369–2374.

[33] Zhu Sun, Qing Guo, Jie Yang, Hui Fang, Guibing Guo, Jie Zhang, and Robin Burke. 2019. Research commentary on recommendations with side information: A survey and research directions. *Electronic Commerce Research and Applications* 37 (2019), 100879.

[34] 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*. 297–305.

[35] Chang-You Tai, Meng-Ru Wu, Yun-Wei Chu, Shao-Yu Chu, and Lun-Wei Ku. 2020. MVIN: Learning Multiview Items for Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 99–108.

[36] Komal Teru, Etienne Denis, and Will Hamilton. 2020. Inductive relation prediction by subgraph reasoning. In *International Conference on Machine Learning*. PMLR, 9448–9457.

[37] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 417–426.

[38] 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*. 1835–1844.

[39] Hongwei Wang, Fuzheng Zhang, Mengdi Zhang, Jure Leskovec, Miao Zhao, Wenjie Li, and Zhongyuan Wang. 2019. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 968–977.

[40] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tatseng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. (2019), 950–958.

[41] Xiang Wang, Xiangnan He, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. 2018. Tem: Tree-enhanced embedding model for explainable recommendation. In *Proceedings of the 2018 World Wide Web Conference*. 1543–1552.

[42] Ze Wang, Guangyan Lin, Huobin Tan, Qinghong Chen, and Xiyang Liu. 2020. CKAN: Collaborative Knowledge-aware Attentive Network for Recommender Systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 219–228.

[43] Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Peng He, Paul Weng, Han Gao, and Guihai Chen. 2019. Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. In *The World Wide Web Conference*. 2091–2102.

[44] Yingce Xia, Xu Tan, Fei Tian, Tao Qin, Nenghai Yu, and Tie-Yan Liu. 2018. Model-level dual learning. In *International Conference on Machine Learning*. PMLR, 5383–5392.

[45] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval*. 55–64.

[46] Chenyan Xiong, Zhengzhong Liu, Jamie Callan, and Tie-Yan Liu. 2018. Towards better text understanding and retrieval through kernel entity salience modeling. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 575–584.

[47] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How Powerful are Graph Neural Networks?. In *International Conference on Learning Representations*.

[48] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 974–983.

[49] 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*. 353–362.

[50] Hanwang Zhang, Fumin Shen, Wei Liu, Xiangnan He, Huanbo Luan, and Tat-Seng Chua. 2016. Discrete collaborative filtering. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 325–334.

[51] Jun Zhao, Zhou Zhou, Ziyu Guan, Wei Zhao, Wei Ning, Guang Qiu, and Xiaofei He. 2019. Intentgc: a scalable graph convolution framework fusing heterogeneous information for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2347–2357.

[52] Kun Zhou, Wayne Xin Zhao, Shuqing Bian, Yuanhang Zhou, Ji-Rong Wen, and Jingsong Yu. 2020. Improving conversational recommender systems via knowledge graph based semantic fusion. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1006–1014.

[53] Chenyi Zhuang and Qiang Ma. 2018. Dual graph convolutional networks for graph-based semi-supervised classification. In *Proceedings of the 2018 World Wide Web Conference*. 499–508.