IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 29 March 2024, accepted 10 May 2024, date of publication 14 May 2024, date of current version 4 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3400788

RESEARCH ARTICLE

Hierarchically Coupled View-Crossing Contrastive Learning for Knowledge Enhanced Recommendation

SHUAI CHEN AND ZHOUJUN LI
State Key Laboratory of Software Development Environment, Beihang University, Beijing 100083, China
Corresponding author: Zhoujun Li (lizj@buaa.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62276017, Grant U1636211, and Grant 61672081; and in part by the Fund of the State Key Laboratory of Software Development Environment under Grant SKLSDE-2021ZX-18.

ABSTRACT Knowledge enhanced recommendation algorithms focus on how to leverage auxiliary information from knowledge graphs to enhance recommendation performance. However, existing methods for knowledge enhanced recommendation often overlook the issues of the non-uniform distribution of task-relevant information: (1) Item nodes often have neighbors unevenly distributed across interaction graphs and knowledge graphs. This uneven distribution of neighbor nodes might lead to the information from certain sources being ignored during message passing, thereby reducing the quality of the learned node embeddings. (2) The implicit inclusion of noise within graph data exacerbates the aforementioned issues, further hindering the effective utilization of knowledge graph information. In this paper, we introduce a novel algorithm called hierarchically coupled view-crossing contrastive learning for knowledge enhanced recommendation to address the challenges mentioned above. Specifically, we controllably couple knowledge graph information into each layer of message passing, and then use a weighted sum of the embeddings learned hierarchically as the final node representation. In addition, we devised a view-crossing contrastive learning approach to construct two additional contrastive learning loss functions for joint training with the main task and more effectively mitigate the adverse impact of noise than the traditional contrastive learning paradigms. Extensive experiments on three real-world graph datasets show that our proposed model performs significantly better than the state-of-the-art baselines and the results of experiments involving adversarial samples indicate the robustness of our model.

INDEX TERMS Contrastive learning, graph neural network, knowledge graph, recommender systems.

I. INTRODUCTION

Recommendation systems play a vital role in modern society. Initially, recommendation systems primarily relied on classic Collaborative Filtering (CF) techniques [1], [2], [3], [4], which analyze user-item interaction data to transform this information into latent representations for predicting user preferences. Collaborative filtering has achieved tremendous success in various recommendation scenarios. Traditional CF methods typically focus only on direct interactions between users and items. In contrast, Graph Neural Networks (GNNs) [5], [6] can simulate the complex relationships between users and items, including indirect interactions. GNNs consider multi-hop relationships between users and items, thereby revealing deeper levels of user preferences and item attributes. As a result, in recent years, GNNs have been widely applied in recommendation systems [7], [8], [9] due to their powerful capability in modeling relational data. Knowledge Graphs (KG), as graphical structures that describe entities and their rich relationships, provide additional semantic and contextual information for recommendation systems. Knowledge graphs contain a wealth of structured information about items, such as categories, attributes, brands, and user reviews. This information can help recommendation systems better understand the characteristics and context of items. By leveraging the entity relationships in knowledge graphs, recommendation systems can more accurately capture user interests and preferences, leading to more personalized recommendations. Consequently, knowledge-enhanced recommendation algorithms have received substantial research

The associate editor coordinating the review of this manuscript and approving it for publication was Mu-Yen Chen.

attention in recent years [10], [11], [12], [13]. Knowledge-enhanced recommendation algorithms primarily focus on effectively utilizing the structured information in knowledge graphs to augment recommendation systems. Graph Neural Networks (GNNs) are used to aggregate neighborhood information from both interaction graph and knowledge graph to generate richer representations of users and items. This includes using GNNs to process knowledge graphs and combining user-item interactions with knowledge graphs in hybrid graph models. knowledge-enhanced recommendation algorithms often encounter issues with data sparsity and noise interference. In such scenarios, contrastive learning has been introduced into recommendation algorithms [14], [15], [16], [17]. By emphasizing the intrinsic structure within graph data, contrastive learning helps models learn richer information from limited interaction data, thereby maintaining relatively good recommendation performance even in sparse data environments. Additionally, knowledge graphs may contain erroneous or inaccurate information. Contrastive learning, by emphasizing the similarities and differences between nodes, aids to identify and resist this noise.

Current contrastive learning based methods have achieved commendable results in knowledge-enhanced recommendations [18], [19], [20], yet challenges persist in effectively utilizing information from the knowledge graph. Knowledge graphs store a vast array of diverse associative relationships between items and entities, playing a crucial role in understanding user preferences and achieving superior recommendation result. In other words, with the augmentation of knowledge graphs, each item node has two types of neighboring information: one part originates from the user-item interaction graph, while the other is derived from the knowledge graph. However, the task-relevant information carried by the same item node from two different graphs is not balanced, significantly affecting the learning of item node embeddings. On one hand, there might be a substantial disparity in the number of user neighbors compared to entity neighbors for a given item node. We refer to the proportion of the neighbors of an item node originating from the user-item interaction graph as the interaction neighbor ratio, which ranges between 0 and 1. An item node has an interaction neighbor ratio value around 0.5 indicates that the neighbors of the item are almost evenly distributed between the interaction graph and the knowledge graph, and vice versa. As shown in Figure 1, we have compiled the distribution of the number of item nodes with varying interaction neighbor ratios across three commonly used datasets in knowledge enhanced recommendation research. The chart reveals that a substantial number of item nodes have an interaction neighbor ratio that deviates significantly from the 0.5 value, which represents a balance between interaction and knowledge levels. This implies that many item nodes do not have neighbors evenly distributed between the interaction graph and the knowledge graph. However, in message passing on the graph data, the representation of a deeper layer node depends on all its neighbors, including those in both the interaction graph and the knowledge graph, denoted as $h_i^{l+1} = \mathrm{Agg}\left(\{h_j^l \mid j \in \mathcal{N}_i^{(ui)} \cup \mathcal{N}_i^{(kg)}\}\right)$, where $\mathcal{N}_i^{(ui)}$ and $\mathcal{N}_i^{(kg)}$ represents the set of neighbors of node i in user-item interaction graph and knowledge graph, h_i^{l+1} is the embedding of node i in layer $l+1$ and Agg represents the aggregation function. Recently, recommendation models based on contrastive learning have effectively improved the recommendation performance for low-degree items on the interaction graph. However, these methods do not address the issue of uneven distribution of neighbor nodes across the interaction graph and the knowledge graph from the perspective of the aggregation function. Specifically, conventional aggregation methods target each neighbor node, resulting in the information contained within $\mathcal{N}_i^{(ui)}$ being easily overwhelmed in message passing when the cardinality of $\mathcal{N}_i^{(kg)}$ is much larger than that of $\mathcal{N}_i^{(ui)}$, and vice versa. This leads to a challenge in balancing the information derived from both graphs during message passing, resulting in a decline in recommendation effectiveness. On the other hand, noise in graph data implicitly increases the difficulty of filtering and utilizing effective information and exacerbated the aforementioned issues. Such noise may exist in either the user-item interaction graph or the knowledge graph, with no prior knowledge of its specific location.

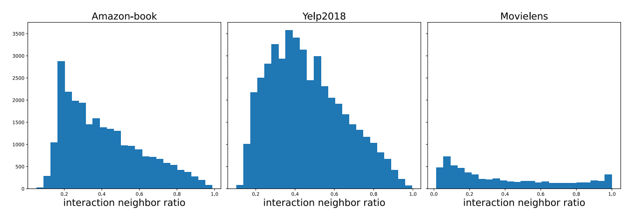


FIGURE 1. Distribution chart of item quantities with different interaction neighbor ratios in three datasets.

To address these challenges, we have developed a model called **H**ierarchically **C**oupled **V**iew-Crossing **C**ontrastive **L**earning for Knowledge Enhanced Recommendation(HCVCL). To balance information from the interaction graph and the knowledge graph during inter-layer message passing, we designed a hierarchically knowledge coupled representation learning method to achieve improved node embedding. Before entering the message passing mechanism, we first align the initial node embedding space of the knowledge graph with the interaction graph through a nonlinear transformation. During layer-by-layer message passing in the interaction graph, we impartially integrate the item node representations learned exclusively from the knowledge graph into the aggregation function designed for the interaction graph, and then use a weighted sum of the embeddings learned at each layer as the final node representation. Furthermore, we devised a view-crossing contrastive learning paradigm to effectively mitigate the negative impact of noise from various sources on recommendation performance. After generating an enhanced view for both the interaction graph and the knowledge graph, we cross-combine the augmented interaction (knowledge) view with the original knowledge (interaction) view. Then, the node embeddings learned from these two combinations are utilized to construct two contrastive loss functions, each in contrast with the embeddings learned from the original view combinations of the interaction and knowledge graphs. In summary, our Hierarchically Coupled View-crossing Contrastive Learning (HCVCL) model makes the following contributions:

- This work addresses the issue of uneven distribution of task-relevant information in knowledge enhanced recommendation systems. On one hand, there is often a significant disparity in the number of neighbors from different sources for item nodes, which leads to knowledge (or interaction) information being difficult to learn in message passing under some circumstances. On the other hand, noise exists in both the interaction graph and the knowledge graph greatly increases the likelihood of the aforementioned issues occurring. These two factors together make it challenging to learn useful information for the knowledge enhanced recommendation task.
- We proposed the hierarchically coupled view-crossing contrastive learning model for knowledge enhanced recommendation. First, a hierarchically knowledge coupled representation learning method is proposed. It mitigates the issue of uneven sources of item node neighbors by controllably coupling knowledge graph information during message passing, which avoids the information from one side being overwhelmed in the message passing process and enhances the recommendation performance. Secondly, we propose a view-crossing contrastive learning paradigm. Compared to conventional contrastive learning methods, it constructs additional contrastive loss functions under the same data augmentation setting, which more efficiently leverages the principles of contrastive learning to enhance the resistance to noise of our model.
- We conducted experiments on three real-world datasets, and the results showed that our model achieved better recommendation performance compared to the current state-of-the-art models. An ablation study confirmed the effectiveness of each module designed in our model. Additionally, we demonstrated the robustness of our model by introducing adversarial samples to training data.

II. RELATED WORK

A. KNOWLEDGE ENHANCED RECOMMENDATION

Knowledge-enhanced recommendation algorithms are mainly categorized into three types: embedding-based methods, connection-based methods, and graph neural network based methods. Embedding-based Methods: These methods utilize rich semantic information in knowledge graph to enrich the representations of user and item nodes. Two-stage embedding-based method [21], [22], [23] initially acquires node embeddings on the knowledge graph independently. Joint training approach train the knowledge graph node embeddings and the recommendation task together. CKE [10] integrates collaborative filtering with knowledge base embeddings to enhance recommendation systems by combining user-item interactions with structured knowledge graph information. SHINE [24] develops an embedding method for signed heterogeneous information networks, focusing on predicting sentiment links by capturing both structural and sentiment relationships in complex network data. The multi-task training approach aims to simultaneously optimize both the knowledge graph embeddings and the recommendation task by treating them as related but distinct tasks. Reference [25] present a multitask learning framework that enhances recommendation systems by integrating feature learning from knowledge graphs and user-item interaction data for more accurate and informative recommendations. Reference [26] propose a novel framework that combines knowledge graph learning with recommendation systems, aiming to enhance the understanding and prediction of user preferences through a unified model. The connection-based method focuses on leveraging knowledge graphs to mine the relationships between various entities within the graph. This approach typically utilizes the rich inter-entity connections present in knowledge graphs to enhance the recommendation system's understanding of item characteristics and user preferences. Meta-structure based methods [27], [28] utilize meta-structures, such as meta-paths and meta-graphs, to learn relationships between nodes. These methods focus on capturing the complex and higher-order relationships in knowledge graphs by constructing and exploiting these meta-structures. Path-based methods learn the explicit embedding of paths in knowledge graphs [29], [30], [31]. Graph Neural Network (GNN)-based methods leverage graph neural network technology to mine deep inter-node relationships in knowledge graphs and enhance recommendation effectiveness. Some GNN-based methods focus on refining the representations of user nodes. RippleNet [32] propagates user preferences over a knowledge graph to capture the potential interests of users more comprehensively and accurately. AKUPM [33] develops an advanced recommendation model that integrates attention mechanisms with knowledge-aware user preference modeling, aiming to improve the accuracy and relevance of recommendations by effectively capturing user interests. Some GNN-based methods focus on refining the representations of item nodes. KGCN [11] employs knowledge graph convolutional networks to enhance recommender systems by leveraging the rich relational information embedded in knowledge graphs for more accurate and personalized recommendations. KGCN-LS [11] adds a label smoothness (LS) regularization on the KGCN model to mitigate overfitting. Some GNN-based methods refine the representations of user and item nodes together [34], [35], [36].

B. CONTRASTIVE LEARNING FOR RECOMMENDATION

Contrastive learning for recommendations aims to enhance recommendation outcomes by introducing additional self-supervised signals through contrastive learning. Some methods based on contrastive learning are designed to augment general recommendation algorithms. SGL [14] presents a novel approach for recommendation systems, employing self-supervised learning techniques within a graph learning framework to enhance the capability of the model in capturing complex user-item interactions and relationships. Reference [15] investigates the use of contrastive learning in large-scale recommendation systems to generate unbiased candidate selections, addressing the issue of bias in recommendations. Some contrastive learning based methods are designed to augment knowledge enhanced recommendation algorithms. MCCLK [18] introduces an innovative approach for recommendation systems that leverages multi-level cross-view contrastive learning, integrating knowledge graph information to enhance the recommendation process by effectively capturing and utilizing the rich semantic relationships in the data. KACL [17] proposes a model based on contrastive learning that can identify task-irrelevant connections in knowledge graphs and encode the shared information across interaction graph views and knowledge graph views. KGCL [19] presents a new framework that integrates contrastive learning with knowledge graphs to improve recommender systems, focusing on leveraging the structural and semantic relationships within knowledge graphs to enhance the quality and relevance of recommendations. KGRec [20] introduces a self-supervised learning framework within knowledge graphs for recommendation systems, focusing on enhancing the recommendation process by leveraging internal rationalizations and relationships in knowledge graphs for more accurate and context-rich user preferences prediction.

III. METHODOLOGY

In this section, we present our HCVCL framework for knowledge-enhanced top-K recommendation task. We will introduce the construction of our model in three sections, namely hierarchically knowledge coupled representation learning, view-crossing contrastive learning and joint training and prediction. Figure 2 presents the model architecture of our proposed HCVCL model.

A. HIERARCHICALLY KNOWLEDGE COUPLED REPRESENTATION LEARNING

When the node representation learning of knowledge-enhanced recommendation algorithms faces the scenario where item node neighbors are unevenly distributed across interaction graphs and knowledge graphs, we alleviate the negative effects of this issue by controlling the proportion of information from the knowledge graph entering the message passing stage. Finally, we use weighted summation to combine the embeddings learned at different layers to form the final embedding of the node.

1) KNOWLEDGE COUPLED MESSAGE PASSING

In graph representation learning, the update of the target node embedding often depends on the features of its neighboring nodes. When the neighboring nodes of the target node incorporate external knowledge, we control the intensity of knowledge coupling, allowing both user-item interaction information and external knowledge to be transmitted to the next layer simultaneously. Specifically, given a user-item interaction graph $\mathcal{G}_u$ and a knowledge graph $\mathcal{G}_k$, we perform information aggregation and representation updates for each node from a global perspective. We denote the merged global graph resulting from the combination of $\mathcal{G}_u = (\mathcal{V}_u, \mathcal{E}_u)$ and $\mathcal{G}_k = (\mathcal{V}_k, \mathcal{E}_k)$ as $\mathcal{G}_g$ which is defined as $\mathcal{G}_g = (\mathcal{V}_g, \mathcal{E}_g)$ where the node set $\mathcal{V}_g = \mathcal{V}_u \cup \mathcal{V}_k$ involves all the nodes in $\mathcal{G}_u$ and $\mathcal{G}_k$, and the edge set $\mathcal{E}_g = \mathcal{E}_u \cup \mathcal{E}_k$ represents all the edges in $\mathcal{G}_u$ and $\mathcal{G}_k$. For each node in $\mathcal{G}_g$, we define the knowledge coupled message passing procedure as follows:

$$h_n^{l+1} = I_{\mathcal{V}_u}(n) \cdot \mathrm{Agg}\left(\left\{h_i^l \middle| \, i \in \mathcal{N}_n \cap \mathcal{V}_u\right\}\right) + I_{\mathcal{V}_k}(n) \cdot \hat{h}_n \tag{1}$$

where h_n^l is the embedding of node n in $l(l = 0, 1, \cdots)$ layer, $I_{\mathcal{V}}(n)$ is the indicator function which returns 1 if node $n \in \mathcal{V}$ else 0, $\mathcal{N}_n$ denotes the set of all the neighbors of node n, $\hat{h}_n$ is the coupled vector for node n in aggregation of every layer.

Due to the efficiency and effectiveness of LightGCN [9], when the neighbors of node n belong to the user-item interaction graph $\mathcal{G}_u$, we apply a similar aggregation method Agg to these part of neighbors as in LightGCN. We define Agg as follows:

$$\mathrm{Agg}\left(\left\{h_i^l \middle| \, i \in \mathcal{N}_n \cap \mathcal{V}_u\right\}\right) = \sum_{i \in \mathcal{N}_n \cap \mathcal{V}_u} \frac{1}{|\mathcal{N}_n \cap \mathcal{V}_u| \cdot |\mathcal{N}_i \cap \mathcal{V}_u|} h_i^l \tag{2}$$

where $|\mathcal{N}_n \cap \mathcal{V}_u|$ means the number of elements in set $\mathcal{N}_i \cap \mathcal{V}_u$.

The knowledge graph $\mathcal{G}_k$ exists independently from the interaction graph $\mathcal{G}_u$, and the node embeddings in the knowledge graph do not share the same feature space with the interaction graph. Therefore, before entering the message passing phase, it is necessary to project the node embeddings from the knowledge graph to maintain consistency in the feature space. For node n in $\mathcal{G}_k$, we have:

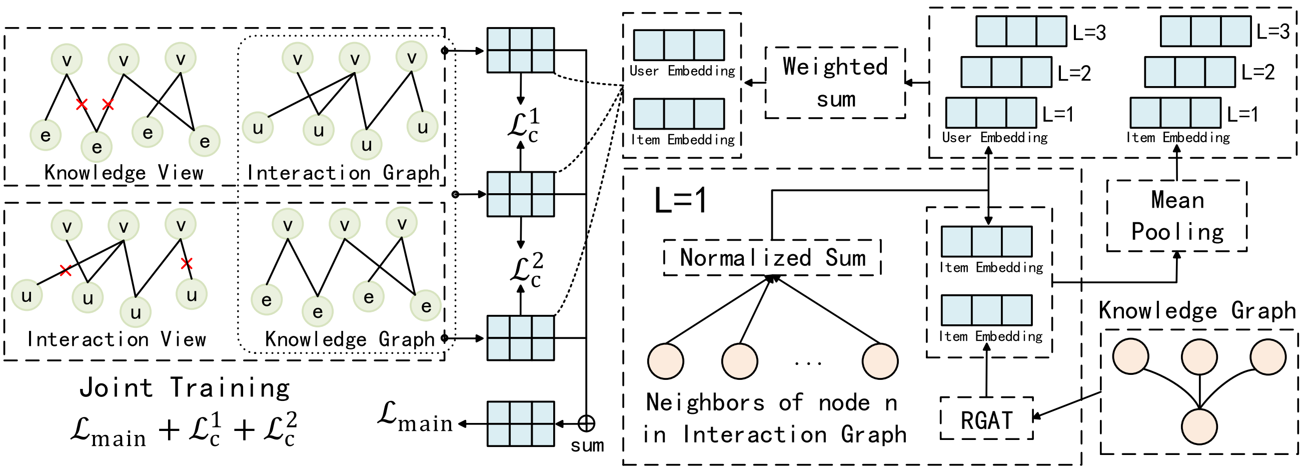
$$e'_n = \mathbf{W_k} \cdot e_n \tag{3}$$

where e'_n is the projected embedding of node n, $\mathbf{W_k}$ is the weight matrix to be learned. Inspired by [34], we use an attention-based aggregation method for nodes belonging to $\mathcal{G}_k$ to obtain the coupled vector $\hat{h}_n$ by the steps outlined below:

$$h_n^{l+1} = \sigma \cdot \sum_{i \in \mathcal{N}_n \cap \mathcal{V}_k} \alpha^l(i, r_{n,i}, n) \cdot h_i^l \tag{4}$$

$$\alpha^l\left(i, r_{n,i}, n\right) = \frac{exp\left(-LeakyReLU\left(\beta_{n,i}\right)\right)}{\sum_{m \in \mathcal{N}_n \cap \mathcal{V}_k} exp\left(-LeakyReLU\left(\beta_{m,i}\right)\right)} \tag{5}$$

$$\beta_{n,i} = h_{r_{n,i}}^T \cdot \mathbf{W} \cdot \left[h_n^l \parallel h_i^l\right] \tag{6}$$



(a) View-Crossing Contrastive Learning (b) Hierarchically Knowledge Coupled Representation Learning

FIGURE 2. The overall framework of our proposed HCVCL model. After constructing two enhanced global views through the view-crossing method, we obtain the node embeddings of nodes in each view via hierarchically knowledge coupled representation learning. Finally, we construct two contrastive loss functions and conduct joint training with the primary recommendation task.

where $r_{n,i}$ represents the relation between node n and i in $\mathcal{G}_k$, *LeakyReLU* is the activate function, $h_{r_{n,i}}$ is the embedding vector of $r_{n,i}$, $W \in \mathbb{R}^{d\times 2d}$ is the weight matrix to be trained, $\|$ represents the concatenation function and h_n^0 is the initial embedding of node n, i.e., $h_n^0 = e'_n$. It is worth noting that we have added a negative sign in front of the activation function to enhance the numerical stability of the attention model.

2) HIERARCHICAL INTEGRATED REPRESENTATION LEARNING

After message passing at each layer, we obtain a node vector representation that has been coupled with knowledge information from the knowledge graph. We use a weighted sum of the node representations from each layer to serve as the final vector representation of the node, as shown below:

$$h_n = \sum_{l=1}^{L} \frac{1}{l+1} \cdot h_n^l \tag{7}$$

where h_n^l is the representation vector of node n in layer l. We believe that message passing at each layer can learn useful information integrated with the knowledge graph to varying degrees. Therefore, drawing on the design of LightGCN, we set different weights for the embeddings at each layer. However, unlike LightGCN, our method does not include the embeddings from the initial (0th) layer, as these initial embeddings have not learned any information from the knowledge graph, making them not aligned with the knowledge-coupled node embeddings from other layers in the same feature space. Our experiments have also shown that including the embeddings of initial layer leads to a significant decline in recommendation performance. Thus, after several layers of message passing, we obtain the final representations for each node, which will be used to construct the loss functions for the recommendation task and contrastive learning.

B. VIEW-CROSSING CONTRASTIVE LEARNING

After determining the graph embedding computation method, we employ the contrastive learning paradigm to alleviate graph sparsity issues. Existing contrastive learning methods often suffer from the inefficiency of utilizing the additional generated views. We designed a view-crossing approach, which constructs additional contrastive loss function to better mine both interaction graph view and knowledge graph view information.

1) AUGMENTATION ON GRAPH STRUCTURE

Following the approach in [14], we utilize edge dropout to augment data on both interaction graph $\mathcal{G}_u$ and knowledge graph $\mathcal{G}_k$. The augmented graph is referred to as a view of the original graph and denoted as $\mathcal{G}'_u$ and $\mathcal{G}'_k$. Specifically, we obtain $\mathcal{G}'_u$ and $\mathcal{G}'_k$ according to the following formula:

$$\mathcal{G}'_u = (\mathcal{V}_u, M_u \odot \mathcal{E}_u), \mathcal{G}'_k = (\mathcal{V}_k, M_k \odot \mathcal{E}_k) \tag{8}$$

where $M_i \in \{0, 1\}^{|\mathcal{E}_i|}$, $i \in \{u, k\}$ are masking vectors on the edge set $\mathcal{E}_i$.

2) VIEW-CROSSING CONTRASTIVE LOSS CONSTRUCTION

After data augmentation on the graph, we obtain the augmented view $\mathcal{G}'_u, \mathcal{G}'_k$ as well as the original graph $\mathcal{G}_u, \mathcal{G}_k$. By employing the view-crossing approach to fully utilize the existing perspectives, we are able to construct additional contrastive loss function, further enhancing the resilience to noise interference of our model. The first step involves cross-matching the original graph with the graph augmented through data enhancement. concretely, we match the original interaction graph $\mathcal{G}_u$ with the augmented knowledge graph $\mathcal{G}'_k$ to obtain a knowledge enhanced global view $\mathcal{G}_g^{(k)}$, as well as a interaction enhanced global view $\mathcal{G}_g^{(u)}$ by matching the augmented interaction graph $\mathcal{G}'_u$ with the original knowledge graph $\mathcal{G}_k$. For original global view $\mathcal{G}_g$

and the knowledge enhanced global view $\mathcal{G}_g^{(k)}$, we can obtain user and item embeddings by utilizing the method from the previous knowledge coupled aggregation module, denoted as $\mathcal{H}$ and $\mathcal{H}^{(k)}$, respectively. According to the principle of contrastive learning for recommendation [14], the embedding representations of the same user (or item) nodes in different views should be as similar as possible, while the distance from the embedding representations of different nodes should be as distinct as possible. Formally, we construct the InfoNCE [37] loss function to maximize the agreement of positive nodes pairs and minimize the agreement of different pairs as shown as the following formula:

$$\mathcal{L}_c^{(k)} = \sum_{n\in\mathcal{V}_u} -\log \frac{\exp(s(h_n, h_n^{(k)})/\tau)}{\sum_{n'\in\mathcal{V}_u, n'\neq n} \exp(s(h_{n'}, h_{n'}^{(k)})/\tau)} \tag{9}$$

where $\mathcal{L}_c^{(k)}$ is the knowledge enhanced InfoNCE loss over the original global view $\mathcal{G}_g$ and the knowledge enhanced global view $\mathcal{G}_g^{(k)}$. $h_n^{(k)} \in \mathcal{H}^{(k)}, h_n \in \mathcal{H}$ is the node embedding vector in global view $\mathcal{G}_g$ and $\mathcal{G}_g^{(k)}$. $\mathcal{V}_u$ is the set of all user and item nodes and τ is the temperature parameter. We adopt the cosine function $s(\cdot)$ as the similarity function between positive node pairs and negative pairs. Following the same procedure, we can obtain the interaction enhanced contrastive loss function $\mathcal{L}_c^{(u)}$ over global view $\mathcal{G}_g$ and interaction enhanced global view $\mathcal{G}_g^{(u)}$ by matching the augmented interaction graph $\mathcal{G}_u'$ with the original knowledge graph $\mathcal{G}_k$. Finally, we obtain the ultimate contrastive loss function $\mathcal{L}_c$ by summing the knowledge-enhanced contrastive loss function and the interaction-enhanced contrastive loss function, as shown in the following equation:

$$\mathcal{L}_c = \mathcal{L}_c^{(k)} + \mathcal{L}_c^{(u)} \tag{10}$$

C. JOINT TRAINING AND PREDICTION

In this section, we will provide a detailed explanation of the training and prediction processes of our entire model. Note that the contrastive learning loss learns node embeddings solely in a self-supervised manner, disregarding collaborative filtering signals. Therefore, we supplement the contrastive InfoNCE loss as a complementary loss function for the main recommendation task and train them jointly, thereby robustly learning user preferences. For the main recommendation task, we use the mean pooling of embeddings from each global view as the final node embedding.

$$z_n = \text{mean}\left(h_n, h_n^{(u)}, h_n^{(k)}\right) \tag{11}$$

where $h_n, h_n^{(u)}, h_n^{(k)}$ represents the node embedding for node n in global view $\mathcal{G}_g$, interaction enhanced global view $\mathcal{G}_g^{(u)}$ and knowledge enhanced global view $\mathcal{G}_g^{(k)}$ separately. We employ the Bayesian Personalized Ranking (BPR) loss [38] as the main recommendation loss to encourage a higher inner product of embeddings between user nodes and historically interacted item nodes, compared to that with unobserved item nodes.

$$\mathcal{L}_{\text{main}} = \sum_{u\in\mathcal{U}} \sum_{i^+\in\mathcal{N}_u} -\ln\sigma\left(z_u^T \cdot z_{i^+} - z_u^T \cdot z_{i^-}\right) \tag{12}$$

where $\mathcal{U}$ is the set of all user nodes, $\mathcal{N}_u$ is the set of all item neighbors of node u, σ represents the sigmoid activation function. z_{i^+} is the embedding vector of the item node that has interacted with the user node u, while z_{i^-} is the embedding vector of the item node that has not interacted with u, obtained by randomly sampling from items that have no interaction with u. By integrating the BPR collaborative filtering loss with the contrastive loss, we minimize the following objective function to conduct joint training:

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \lambda_1\mathcal{L}_c + \lambda_2\|\Theta\|_2^2 \tag{13}$$

where λ_1 controls the strengths of contrast learning, λ_2 determines the strength of regularization for the joint loss function. Θ represents the set of all model parameters. During the inference phase, we use the node representations learned on the original global view $\mathcal{G}_g$ to make predictions, treating the inner product of the embeddings of user nodes and item nodes as their matching scores.

IV. EXPERIMENTS

A. EXPERIMENTAL SETTINGS

1) DATASETS

We conducted experiments on three publicly available datasets derived from real-world scenarios. Detailed information about the three datasets is listed in Table 1. Similar to [17], [19], and [34], we employ a 10-core setting for user and item nodes, wherein we retain the nodes with interaction frequencies exceeding 10 occurrences in original data. For each user node within the user-item interaction graph, we randomly select 80% of the item nodes that interact with it as training data, while the remaining 20% serve as testing data.

TABLE 1. Statistics of three experimented datasets.

Dataset	Amazon-book	Yelp2018	Movielens
# Users	70679	45919	37385
# Items	24915	45538	6182
# Interactions	846434	1183610	539300
# Entities	29714	47472	24536
# Relations	39	42	20
# Triplets	686516	869603	237155

2) BASELINES

We compared the HCVCL model with various baseline models:

a: GENERAL RECOMMENDATION

- **BPR** [38] introduces a bayesian approach to personalized ranking, specifically designed for generating personalized recommendations from implicit feedback like clicks or purchases by users.

- **GC-MC** [39] introduces a novel method using graph convolutional networks for the task of matrix completion, aiming to enhance recommendation systems by leveraging relational data represented as graphs.
- **LightGCN** [9] introduces a simplified and efficient graph convolution network model designed specifically to enhance the performance of recommendation systems.

b: KNOWLEDGE-AWARE RECOMMENDATION:

- **CKE** [10] explores the integration of collaborative filtering with knowledge base embeddings to enhance the accuracy and quality of recommendations in recommender systems.
- **KGCN** [11] introduces the use of Knowledge Graph Convolutional Networks (KGCNs) in recommender systems to effectively incorporate relational information from knowledge graphs, thereby enhancing the recommendation quality and accuracy.
- **KGAT** [34] introduces a novel framework that integrates knowledge graph-based information with an attention mechanism to improve the performance and effectiveness of recommender systems.
- **KGIN** [12] focuses on understanding user intents in recommender systems by leveraging knowledge graphs, proposing a novel approach to model and learn from the complex interactions and intentions behind user behaviors for more accurate and personalized recommendations.
- **CKAN** [13] is a novel framework that combines collaborative filtering with a knowledge-aware attentive mechanism, aiming to enhance recommender systems by effectively integrating both user-item interactions and knowledge graph information for more precise and insightful recommendations.

c: CONTRASTIVE LEARNING BASED METHODS FOR RECOMMENDATION

- **SGL** [14] explores a self-supervised learning approach within graph learning frameworks for recommendation systems, aiming to enhance recommendation quality by leveraging self-supervised signals to capture complex user-item interactions and relationships more effectively.
- **MCCLK** [18] introduces an innovative approach that integrates multi-level cross-view contrastive learning into knowledge-aware recommender systems, aiming to enhance recommendation performance by effectively leveraging the rich semantic information from knowledge graphs and user-item interaction data.
- **KGCL** [19] proposes a novel framework that combines knowledge graph-enhanced recommendation with contrastive learning techniques, aiming to improve the effectiveness of recommender systems by leveraging the contrastive signals derived from knowledge graph structures for more accurate and contextually rich recommendations.
- **KGRec** [20] introduces an innovative approach that integrates self-supervised learning with knowledge graphs in the domain of recommender systems, aiming to enhance the recommendation process by employing self-supervised rationalization techniques to better understand and utilize the complex relationships within knowledge graphs.

3) EVALUATION PROTOCOLS

For the sake of fair comparison, we employed the widely adopted all-ranking strategy [19], [20], [34] to assess the quality of the recommended results. More specifically, for each user node, we arrange the values resulting from the dot product of its embedding and the embeddings of all item nodes in descending order. After excluding the item nodes that have had interactions with the said user node, we consider the top-K entries in the arrangement to calculate the values of NDCG@K and Recall@K [1], [40]. By calculating the average of the metric values across all user nodes, we obtain the final numerical values for the NDCG and Recall evaluation metrics. In our experiments, all values of K are set to be 20.

4) TRAINING DETAILS

In contrast to the majority of prior studies, our model demonstrates consistent and commendable performance across different datasets by employing a uniform set of hyperparameters. Most of compared baselines are evaluated based on the unified recommendation library RecBole [41]. For those baseline models not included in the RecBole library, we train them using the officially recommended code and hyper-parameters. For our model, the learning rate is 0.0005, the batch size is 4098, and the num of graph neural network layers is 3. On both interaction graph and knowledge graph, we randomly initialize the initial node embedding with Kaiming initializer [42], and the dropout ratio is set to be 0.5. In the contrastive learning section, the temperature parameter τ is uniformly set to 0.2, while contrastive loss balance parameter λ is set to be 0.1. On the hardware front, we utilized Intel i7-13700K CPU, NVIDIA TITAN RTX GPU for all model training and inference.

B. OVERALL PERFORMANCE COMPARISON

In Table 2, we listed the performance of all models across three datasets. Through analysis, we can draw the following conclusions:

- Our HCVCL model consistently outperforms all baseline models. We conducted evaluations on three real-world datasets of varying scenarios and scales and using mainstream evaluation metrics. This demonstrates that our HCVCL model effectively utilizes knowledge graphs to enhance recommendation performance and adeptly handles task-irrelevant information in the data.

TABLE 2. The overall experimental results of our proposed model and compared baseline models on three real world experimented datasets.

Model	Amazon-book		Yelp2018		Movielens	
	Recall	NDCG	Recall	NDCG	Recall	NDCG
BPR	0.1241	0.0639	0.0551	0.0376	0.4052	0.2609
GC-MC	0.1052	0.0531	0.0683	0.0451	0.4322	0.2070
LightGCN	0.1383	0.0729	0.0679	0.0440	0.4466	0.2164
CKE	0.1375	0.0686	0.0687	0.0432	0.4106	0.2669
KGCN	0.1122	0.0571	0.0534	0.0339	0.4237	0.2753
KGAT	0.1383	0.0738	0.0678	0.0431	0.4532	0.3007
KGIN	0.1441	0.0751	0.0710	0.0463	0.4661	0.3120
CKAN	0.1378	0.0734	0.0687	0.0442	0.4314	0.2891
SGL	0.1442	0.0762	0.0715	0.0469	0.4390	0.2159
MCCLK	0.1314	0.0706	0.0654	0.0422	0.4042	0.2539
KGCL	0.1506	0.0796	0.0755	0.0491	0.4411	0.2982
KGRec	0.1553	0.0818	0.0661	0.0425	0.4561	0.3030
HCVCL	**0.1685**	**0.0899**	**0.0824**	**0.0537**	**0.4726**	**0.3263**

- Comparing models based on contrastive learning (such as SGL, MCCLK, KGCL, KGRec) with other models reveals that the former generally perform better than the latter. This indicates that the incorporation of additional self-supervised signals indeed effectively enhances model performance. When comparing models that integrate knowledge graph information (like KGRec, KGCL, KGIN, KGAT, etc.) with those that do not utilize knowledge graph data, the former often exhibit superior performance. This underscores that appropriately leveraging the rich information in knowledge graphs can indeed enhance recommendation effectiveness. However, there are instances where models without knowledge graph integration perform better, highlighting the importance of effectively handling task-irrelevant information in the data. Our HCVCL model, based on the contrastive learning paradigm, effectively utilizes task-relevant information from knowledge graphs and achieved the best performance across all three datasets.

C. ABLATION STUDY

In this section, we evaluate the impact of several key components in our model on recommendation accuracy. Specifically, we substitute conventional components for parts of our designed model architecture and then observe the changes in accuracy on three datasets. This approach allows us to quantitatively assess how our model design influences the effectiveness of recommendations. Based on modifications to the model structure, we generate three variants of our HCVCL model:

- w.o. VC (Without View-Crossing Contrastive Learning): This variant removes the view-crossing module. After eliminating view-crossing, while keeping the number of enhanced views constant, we adopt the traditional contrastive learning paradigm. That is, we construct the contrastive learning loss function using the node embeddings on the enhanced global graph and the original global graph.

TABLE 3. Ablation studies for different variants of HCVCL, in terms of Recall@20 and NDCG@20.

Recall	Amazon-book	yelp2018	Movielens
HCVCL	0.1685	0.0824	0.4726
w/o VC	0.1649	0.0812	0.4706
w/o KC	0.1619	0.0747	0.4641
w/o VC&KC	0.1609	0.0733	0.4343
NDCG	**Amazon-book**	**yelp2018**	**Movielens**
HCVCL	0.0899	0.0537	0.3263
w/o VC	0.0876	0.0526	0.3203
w/o KC	0.0868	0.0495	0.3176
w/o VC&KC	0.0847	0.0483	0.2965

- w.o.KC (Without Knowledge Coupling): This variant removes the knowledge coupling in message passing. The purpose of the knowledge coupling is to balance the information volume from the interaction graph and the knowledge graph for each item node during each layer of message passing. Without knowledge coupling, we use the aggregation function from LightGCN for message passing.
- w.o. VC & KC (Without View-Crossing Contrastive Learning and Knowledge Coupling): This variant completely removes both the view-crossing and knowledge coupling modules, reducing the model to the SGL [14] model on the global view.

The experimental results are presented in Table 3. By comparing these variants, we can effectively evaluate the individual contributions of the view-crossing contrastive learning and knowledge coupling message passing modules to the overall performance of our HCVCL model. In summary, the accuracy of the variants that individually removed the VC (View-crossing) and KC (hlKnowledge Coupling) modules was significantly lower than that of our HCVCL model. This indicates the effectiveness of our model design. The variant that simultaneously removed both VC and KC modules can be seen as a simple extension of the SGL model in the field of knowledge enhanced recommendation. The contrast in accuracy between this variant and our model further highlights the improvements our model offers over classic models. The variant without the KC module consistently showed a more significant drop in accuracy across all three datasets and two metrics, suggesting that the KC module, which directly modifies the core message passing mechanism in graph neural networks, contributes more to our HCVCL model compared to the VC module.

D. ROBUSTNESS TO NOISE

In this section, to validate the robustness of our model against noise interference, we added adversarial samples to the training data in varying proportions. We then trained the model on this expanded dataset and tested it on the same test set. Specifically, for each user node in the training set, we randomly selected a portion of item nodes with which the user had no prior interaction, in proportion to the number of positive user-item interactions the user had, and added

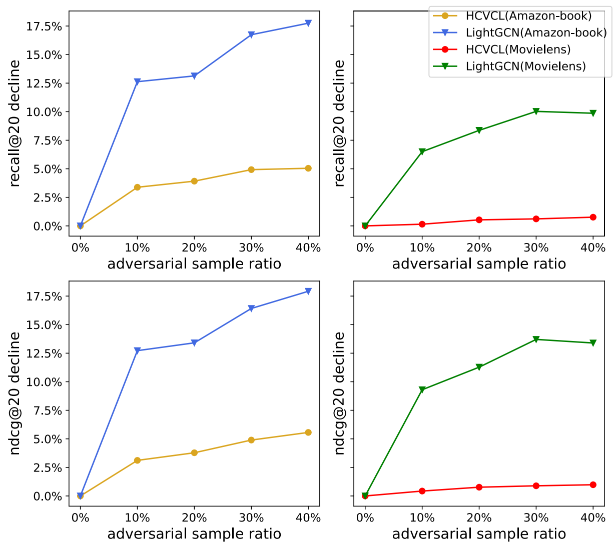


FIGURE 3. A comparison of the relative decline in Recall and NDCG metrics for the HCVCL and LightGCN models after injecting adversarial samples at different proportions.

these connections to the training set. Figure 3 shows that our HCVCL model exhibits good robustness, with only a minor decline in relevant metrics after adding adversarial samples of different proportions to the training data. Particularly when introducing 10% noise, the LightGCN model experiences a precipitous decline in performance, whereas our model demonstrates strong resistance to noise. This likely results from the unique design of our model. First, during the message passing phase, we control the amount of information from both interaction graph and knowledge graph entering the next layer, thus proportionally reducing the influence of any noise added to the interaction graph. Secondly, our view-crossing contrastive learning is more potent than conventional contrastive learning, with the additional two contrastive loss functions endowing the HCVCL model with better robustness.

E. HYPER-PARAMETERS SENSITIVITY STUDY

TABLE 4. Impact of λ_1 and τ on three datasets.

dataset	Recall				
Amazon-book	$\lambda_1, \tau =$	0.1	0.2	0.3	0.4
	0.01	0.1599	0.1610	0.1550	0.1493
	0.1	0.1359	**0.1685**	0.1572	0.1564
	1.0	0.0973	0.1461	0.1427	0.1361
Movielens	$\lambda_1, \tau =$	0.1	0.2	0.3	0.4
	0.01	0.4657	0.4683	0.4494	0.3562
	0.1	0.4367	**0.4726**	0.4336	0.3331
	1.0	0.1612	0.3010	0.3727	0.3736

In this section, we investigate the impact of three hyperparameters on the model performance: the temperature coefficient τ in the contrastive learning loss function and the weight λ_1 of the contrastive learning loss. As shown in Table 4, λ_1 and τ are searched from the range of (0.01, 0.1, 1.0) and (0.1, 0.2, 0.3, 0.4), respectively. Experimental data consistently shows that our model performs best under a specific set of parameters for λ_1 and τ. This indicates that our model possesses good scalability, being able to achieve similarly high performance across different datasets with the same parameter configuration.

V. CONCLUSION

In this work, we deeply analyze the issue of uneven distribution of task-relevant information across interaction graphs and knowledge graphs in knowledge-enhanced recommendation tasks and design a hierarchically coupled view-crossing contrastive learning model to mitigate this problem. In particular, to mitigate the issue caused by the uneven distribution of item node neighbors in the interaction graph and knowledge graph, we designed a hierarchical knowledge coupling graph representation learning method to controllably utilize knowledge graph information to enhance node representation. To mitigate the negative impact of pervasive noise in graph data on recommendation performance, we devised a view-crossing contrastive learning paradigm that more effectively addresses noise issues compared to traditional contrastive learning methods. Our experiments demonstrate the exceptional performance of the HCVCL model across various datasets. The ablation study highlights the effectiveness of the two critical components of our model, and the hyperparameter sensitivity analysis shows consistent superior performance under a unified set of hyperparameters, indicating good scalability of the model. Additionally, we injected adversarial samples into the training data at different ratios to observe changes in the model accuracy, and this experiment revealed the strong noise resistance capabilities of our model.

ACKNOWLEDGMENT

The authors would like to thank all the reviewer's comments to help improve this article.

REFERENCES

[1] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.
[2] J. Lian, X. Zhou, F. Zhang, Z. Chen, X. Xie, and G. Sun, "xDeepFM: Combining explicit and implicit feature interactions for recommender systems," in *Proc. 24th ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining*, 2018, pp. 1754–1763.
[3] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," in *Proc. World Wide Web Conf.*, 2018, pp. 689–698.
[4] S. Rendle, W. Krichene, L. Zhang, and J. Anderson, "Neural collaborative filtering vs. matrix factorization revisited," in *Proc. 14th ACM Conf. Recommender Syst.*, 2020, pp. 240–248.
[5] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.
[6] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lió, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.
[7] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2019, pp. 165–174.
[8] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2018, pp. 974–983.

[9] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "LightGCN: Simplifying and powering graph convolution network for recommendation," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2020, pp. 639–648.

[10] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, "Collaborative knowledge base embedding for recommender systems," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 353–362.

[11] H. Wang, M. Zhao, X. Xie, W. Li, and M. Guo, "Knowledge graph convolutional networks for recommender systems," in *Proc. World Wide Web Conf.*, 2019, pp. 3307–3313.

[12] X. Wang, T. Huang, D. Wang, Y. Yuan, Z. Liu, X. He, and T.-S. Chua, "Learning intents behind interactions with knowledge graph for recommendation," in *Proc. Web Conf.*, 2021, pp. 878–887.

[13] Z. Wang, G. Lin, H. Tan, Q. Chen, and X. Liu, "CKAN: Collaborative knowledge-aware attentive network for recommender systems," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2020, pp. 219–228.

[14] J. Wu, X. Wang, F. Feng, X. He, L. Chen, J. Lian, and X. Xie, "Self-supervised graph learning for recommendation," in *Proc. 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2021, pp. 726–735.

[15] C. Zhou, J. Ma, J. Zhang, J. Zhou, and H. Yang, "Contrastive learning for debiased candidate generation in large-scale recommender systems," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, 2021, pp. 3985–3995.

[16] X. Cai, C. Huang, L. Xia, and X. Ren, "Simple yet effective graph contrastive learning for recommendation," in *Proc. 11th Int. Conf. Learn. Represent.*, 2023. [Online]. Available: https://openreview.net/forum?id=FKXVK9dyMM

[17] H. Wang, Y. Xu, C. Yang, C. Shi, X. Li, N. Guo, and Z. Liu, "Knowledge-adaptive contrastive learning for recommendation," in *Proc. 16th ACM Int. Conf. Web Search Data Mining*, 2023, pp. 535–543.

[18] D. Zou, W. Wei, X.-L. Mao, Z. Wang, M. Qiu, F. Zhu, and X. Cao, "Multi-level cross-view contrastive learning for knowledge-aware recommender system," in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2022, pp. 1358–1368.

[19] Y. Yang, C. Huang, L. Xia, and C. Li, "Knowledge graph contrastive learning for recommendation," in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2022, pp. 1434–1443.

[20] Y. Yang, C. Huang, L. Xia, and C. Huang, "Knowledge graph self-supervised rationalization for recommendation," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, 2023, pp. 3046–3056.

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

[22] Y. Ye, X. Wang, J. Yao, K. Jia, J. Zhou, Y. Xiao, and H. Yang, "Bayes EMbedding (BEM) refining representation by integrating knowledge graphs and behavior-specific networks," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manag.*, 2019, pp. 679–688.

[23] D. Yang, Z. Guo, Z. Wang, J. Jiang, Y. Xiao, and W. Wang, "A knowledge-enhanced deep recommendation framework incorporating GAN-based models," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 1368–1373.

[24] H. Wang, F. Zhang, M. Hou, X. Xie, M. Guo, and Q. Liu, "Shine: Signed heterogeneous information network embedding for sentiment link prediction," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, 2018, pp. 592–600.

[25] H. Wang, F. Zhang, M. Zhao, W. Li, X. Xie, and M. Guo, "Multi-task feature learning for knowledge graph enhanced recommendation," in *Proc. World Wide Web Conf.*, 2019, pp. 2000–2010.

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

[27] X. Yu, X. Ren, Q. Gu, Y. Sun, and J. Han, "Collaborative filtering with entity similarity regularization in heterogeneous information networks," in *Proc. IJCAI HINA*, vol. 27, 2013.

[28] C. Shi, B. Hu, W. X. Zhao, and P. S. Yu, "Heterogeneous information network embedding for recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 2, pp. 357–370, Feb. 2019.

[29] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based context for top-N recommendation with a neural co-attention model," in *Proc. 24th ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining*, 2018, pp. 1531–1540.

[30] X. Wang, D. Wang, C. Xu, X. He, Y. Cao, and T.-S. Chua, "Explainable reasoning over knowledge graphs for recommendation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 5329–5336.

[31] X. Huang, Q. Fang, S. Qian, J. Sang, Y. Li, and C. Xu, "Explainable interaction-driven user modeling over knowledge graph for sequential recommendation," in *Proc. 27th ACM Int. Conf. Multimedia*, 2019, pp. 548–556.

[32] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, and M. Guo, "RippleNet: Propagating user preferences on the knowledge graph for recommender systems," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2018, pp. 417–426.

[33] X. Tang, T. Wang, H. Yang, and H. Song, "AKUPM: Attention-enhanced knowledge-aware user preference model for recommendation," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 1891–1899.

[34] X. Wang, X. He, Y. Cao, M. Liu, and T. Chua, "KGAT: Knowledge graph attention network for recommendation," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Aug. 2019, pp. 950–958.

[35] Y. Qu, T. Bai, W. Zhang, J. Nie, and J. Tang, "An end-to-end neighborhood-based interaction model for knowledge-enhanced recommendation," in *Proc. 1st Int. Workshop Deep Learn. Pract. High-Dimensional Sparse Data*, Aug. 2019, pp. 1–9.

[36] J. Zhao, Z. Zhou, Z. Guan, W. Zhao, W. Ning, G. Qiu, and X. He, "IntentGC: A scalable graph convolution framework fusing heterogeneous information for recommendation," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 2347–2357.

[37] M. Gutmann and A. Hyvarinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 297–304.

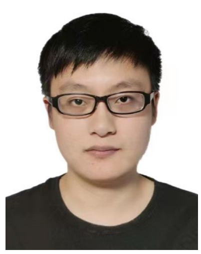
[38] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," 2012, *arXiv:1205.2618*.

[39] R. van den Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," 2017, *arXiv:1706.02263*.

[40] J.-H. Yang, C.-M. Chen, C.-J. Wang, and M.-F. Tsai, "HOP-Rec: High-order proximity for implicit recommendation," in *Proc. 12th ACM Conf. Recommender Syst.*, Sep. 2018, pp. 140–144.

[41] W. X. Zhao, S. Mu, Y. Hou, Z. Lin, Y. Chen, X. Pan, K. Li, Y. Lu, H. Wang, C. Tian, Y. Min, Z. Feng, X. Fan, X. Chen, P. Wang, W. Ji, Y. Li, X. Wang, and J.-R. Wen, "RecBole: Towards a unified, comprehensive and efficient framework for recommendation algorithms," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 4653–4664.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.



SHUAI CHEN received the bachelor's degree from the School of Mathematical Sciences, Nankai University, Tianjin, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer Science, Beihang University, Beijing, China. His research interests include graph neural networks, recommendation systems, and data mining.

ZHOUJUN LI received the bachelor's degree from the School of Computer Science, Wuhan University, in 1984, and the Ph.D. degree from the College of Computer Science and Technology, National University of Defense Technology, in 1999. He is currently a Professor with the School of Computer Science, Beihang University, Beijing, China. His research interests include data mining, artificial intelligence, and natural language processing.