Neural Processing Letters (2024) 56:16
https://doi.org/10.1007/s11063-024-11534-y

A Graph Contrastive Learning Model Based on Structural and Semantic View for HIN Recommendation

Ruowang Yu[1] · Yu Xin[1] · Yihong Dong[1] · Jiangbo Qian[1]

Accepted: 8 January 2024 / Published online: 7 February 2024

Abstract
With the rapid growth of information in the Internet era, people are in great need of recommendation methods to filter information. At present, recommendation methods which based on heterogeneous information network (HIN) have attracted wide attention. Recently, HIN-based recommendation methods need to be modeled from two aspects: node structural association and semantic association. To this end, we propose a graph contrastive learning model based on structural and semantic view for HIN recommendation (GCL-SS). GCL-SS utilizes U-I interactive view to obtain node structural embeddings, and utilizes U-I semantic view to obtain node semantic embeddings. Based on these two kinds of embeddings, we establish a self-supervised contrastive learning mechanism to effectively integrate structural information and semantic information of user (item) nodes in HIN, and finally learn a more discriminative user (item) embedding. In addition, in order to strengthen the semantic association between nodes, we innovatively utilize time sequence encoder (TSE), such as LSTM, to encode semantic homogeneous network decomposed by HIN in U-I semantic view. At last, based on the user and item embeddings, we adopt bilinear decoder to model the potential association between user and item, so as to realize rating prediction of user to item. The experimental results on three real datasets confirm that our GCL-SS model performs better than state-of-the-art recommendation methods in rating prediction task. In addition, the results of four ablation experiments indicate that our GCL-SS model can effectively improve the performance of rating prediction in recommendation.

Keywords Heterogeneous information network (HIN) · Contrastive learning · Time sequence encoder · Rating prediction

Ruowang Yu, Yihong Dong and Jiangbo Qian have equally contributed to this work.

✉ Yu Xin
xinyu@nbu.edu.cn

Ruowang Yu
1464828311@qq.com

Yihong Dong
dongyihong@nbu.edu.cn

Jiangbo Qian
qianjiangbo@nbu.edu.cn

[1] Ningbo University, 818 Fenghua Road, Ningbo 315211, China

1 Introduction

With the rapid growth of the information in e-commerce and social media, it is difficult for people to find favorite products or content in a large amount of information in a short time. In order to greatly improve the information search rate and reduce labor cost, people need to filter information with the help of recommendation systems. Since heterogeneous information network (HIN) [1] is a complex network containing multiple types of nodes and relations, has abundant structural and semantic information, which can provide sufficient content for recommendation system. Therefore, the modeling of recommendation system is mainly based on HIN recently. HINs include movie recommendation network, bibliographic network [2], biomedical network [3], e-commerce network and so on. By modeling the HINs, the recommendation system can mine abundant potential information of all kinds of nodes and potential association between user and item in the HIN. Therefore, in recent years, more and more HIN-based recommendation methods have been proposed.

HIN-based recommendation methods, such as heterogeneous Graph Neural Network (HGNNs), usually need to obtain the embeddings of various types of nodes in HIN. Based on these node embeddings, such methods can achieve rating prediction of user to item by mining the potential association between user and item embeddings. At present, HIN-based recommendation methods can be divided into two categories by the type of user (item) embedding: graph-structure based HIN recommendation method [4–7], and meta-path based HIN recommendation method [8–15].

Graph-structure based HIN recommendation methods implement U-I recommendation for the U-I direct association (i.e., the interaction between user and item). This kind of methods obtain user and item embeddings by encoding graph structure in HIN. These embeddings can reflect the local structural information of user and item nodes in HIN. However, such methods only consider the structural information of user (item) nodes, but do not consider the semantic information of user (item) nodes.

Meta-path based HIN recommendation methods implement U-I recommendation for the UU (II) indirect association (i.e., the semantic associations between users or items). At present, this kind of methods mainly consider decomposing HIN into multiple unweighted homogeneous networks according to the type of meta-path. Then, these methods encode and fuse all unweighted homogeneous networks to obtain user (item) embeddings. Since the meta-path can reflect semantic association between various types of nodes in HIN, the user (item) embeddings obtained by these methods can represent the semantic association information of users (items). However, such methods only consider the semantic information of user (item) nodes, but do not consider the structural information of user (item) nodes and the degree of semantic association between user (item) nodes. If we ignore the degree of semantic association between nodes, a part of semantic information will be lost during message passing.

We establish U-I interactive view and U-I semantic view for node structural embedding and semantic embedding to realize the expressions of graph structure and node semantic association in HIN. Among them, U-I interactive view can extract node neighborhood environment (structural) features by encoding the neighborhood density and neighborhood location of user (item) nodes in U-I interactive network (bipartite graph), so as to obtain the structural embeddings of user (item) nodes. U-I semantic view considers the problems of multi-channel construction for user (item) semantic homogeneous networks, multi-channel feature fusion and semantic feature fusion. So, we firstly construct multiple weighted channels of user (item) semantic homogeneous network by meta-path decomposition and semantic homoge-

neous network weighting. Secondly, we encode and fuse the multi-channel features of user (item) semantic homogeneous network by TSE and self-attention mechanism to obtain the semantic features of user (item) nodes. Finally, we fuse the semantic features of the neighbor heterogeneous nodes through k-hop neighborhood aggregation to obtain the final semantic embeddings of user (item) nodes.

In summary, in order to integrate structural information and semantic information of user (item) nodes more effectively, we propose a graph contrastive learning model based on structural and semantic view for HIN recommendation (GCL-SS). By establishing a contrastive learning mechanism, GCL-SS can effectively combine the information extracted from U-I interactive view and U-I semantic view to improve the performance of recommendation. As a typical self-supervised method, contrastive learning [16] can maximize the similarity between the same node embeddings in U-I interactive view and U-I semantic view, while minimizing the similarity between the different node embeddings. Therefore, this method can integrate the information of U-I interactive view and U-I semantic view, while maintaining the uniqueness and specificality of each view information, so as to effectively retain the information of HIN.

The contributions of our work can be summarized as follows:

- In this paper, we propose the GCL-SS model. It establishes a self-supervised contrastive learning mechanism for U-I interactive view and U-I semantic view. This method can keep the encoding consistency between the same nodes in the two views while keeping the encoding difference between different nodes.
- In U-I semantic view, we innovatively utilize time sequence encoder, such as LSTM, to encode multiple semantic homogeneous networks. This method strengthens the semantic association between user (item) nodes by strengthening the semantic environment context of nodes in HIN.
- Based on GCL-SS, we propose an extended model, GCL-SS$_{AE}$. In U-I semantic view, we compare the structure of the reconstructed semantic homogeneous network with the original network. By strengthening the structural consistency of semantic homogeneous networks, GCL-SS$_{AE}$ can effectively preserve the semantic association information of nodes in HIN.

2 Related Works

In this section, we review some closely related studies, including HIN-based recommendation and graph contrastive learning.

2.1 HIN-Based Recommendation

Heterogeneous information network (HIN) is a complex network containing various types of nodes and rich connection relations, which can describe the relationship between complex information in the real world, such as e-commerce network, movie recommendation network and so on. To this end, by modeling the HINs, we can mine the potential information of various types of nodes and edges in the HIN, so as to obtain low-dimensional vector representations of nodes and edges. The obtained vector representations can be applied to many downstream tasks, such as link prediction, node classification, personalized recommendation [17, 18] and so on. Among them, the methods with personalized recommendation as the downstream task usually utilize user and item embedding to calculate the similarity between user and

item, so as to mine the potential association between user and item. At present, HIN-based recommendation methods can be divided into two categories by the type of user (item) embedding: graph-structure based HIN recommendation method and meta-path based HIN recommendation method.

Graph-structure based HIN recommendation methods encode graph structure mainly by the neighborhood aggregation of various types of nodes in HIN. For example, Wu et al. [4] proposed DiffNet which respectively obtains user and item embeddings by k-hop neighborhood aggregation of user social network and historical behavior. NGCF [5] and LightGCN [6] utilize multi-level message passing mechanism (e.g., k-hop neighborhood aggregation) to obtain user (item) embeddings. Zhang et al. [7] proposed HetGNN which aggregates neighbor nodes according to node types, and obtains node embeddings through two RNNs. The first RNN encodes the feature interaction of each node to obtain node context embeddings. Another RNN aggregates the context embeddings of neighbor nodes according to node types, and introduces the attention mechanism to measure the influence of various types of nodes, so as to obtain the final node embeddings.

Meta-path based HIN recommendation methods can be divided into meta-path based random walk methods and meta-path decomposition methods. (1) Meta-path based random walk methods firstly utilize random walk to obtain heterogeneous node sequences, and then they encode sampling sequence by skip-gram. At last, they utilize the nearest neighbor relationship to learn node embeddings. For example, metapath2vec [8] is an extension and improvement of the Deep Walk [9] algorithm in heterogeneous domain. By strengthening the context association of node sequences which are sampled by random walk, metapath2vec can take full advantage of the rich semantic information of HIN. HINE [10] obtains node embeddings by the meta-path based random walk to model the similarity between nodes. Then it optimizes node embeddings by minimizing the KL divergence of the two similarities (node embedding similarity and meta-path similarity). Fu et al. [11] designs a neural network model (HIN2vec) to maximize the co-occurrence probability of nodes in a meta-path-based random walk path, so as to obtain the vector representation of each node. (2) Meta-path decomposition methods utilize meta-path to decompose heterogeneous information networks into multiple heterogeneous or homogeneous networks. By encoding these heterogeneous or homogeneous networks, the semantic information of each meta-path can be fully utilized. For example, MV-HetGNN [12] firstly obtains multiple heterogeneous networks (multiple views) according to the meta-path, then performs neighborhood aggregation on each view, and finally uses intra-view and inter-view auto-encoding to obtain node embeddings. According to whether the nodes at both ends of the meta-path are the same, HMSG [13] decomposes HIN into multiple homogeneous networks and heterogeneous networks. Then it obtains node embeddings by node-level aggregation and semantic-level aggregation. HERec [14] firstly uses meta-path based random walk to extract meaningful node sequences from the HIN, then utilizes the sampled node sequences to decompose the HIN into multiple homogeneous networks, and finally uses three fusion functions to fuse the node features encoded by multiple homogeneous networks. AMERec [15] weights the homogeneous network after meta-path decomposition to distinguish the importance of node pairs, then performs random walk to extract node features, and finally utilizes self-attention mechanism to fuse the semantic information of multiple meta-paths.

In conclusion, graph-structure based HIN recommendation methods only consider the structural information of user (item) nodes, and meta-path based HIN recommendation methods only consider the semantic information of user (item) nodes. Both node structural and semantic information will have an impact on the final learned node embedding, which will affect the model recommendation performance. Therefore, it is significant to effectively com-

bine the two types of node information, so as to capture more potential information of nodes in HIN.

2.2 Graph Contrastive Learning

At present, as one of the main methods in self-supervised representation learning, contrastive learning has been widely used in computer vision [19, 20] and natural language processing [21, 22]. Contrastive learning utilizes the principle of mutual information maximization to learn feature representations. Inspired by visual contrastive learning, contrastive learning has been applied to graph data mining field recently. According to the network type, graph contrastive learning methods can be divided into homogeneous network contrastive learning (homogeneous domain) and heterogeneous network contrastive learning (heterogeneous domain).

In homogeneous domain, DGI [23] builds local features (i.e., node features) and global features (i.e., graph features) as positive pairs, and utilizes Infomax [24] theory to contrast, so as to learn node embeddings and graph embeddings. MVGRL [25] utilizes node features and graph features obtained from different views, to establish a contrastive learning mechanism. This strategy can effectively improve the accuracy of node classification and graph classification. CSSL [26] takes the augmentation graphs obtained by same graph as positive pairs, and takes the augmentation graphs obtained by different graphs as negative pairs. Based on positive and negative pairs, CSSL utilizes contrastive learning to improve the accuracy of graph classification. GCA [27] respectively utilizes topology-level augmentation and attribute-level augmentation to obtain topology and attribute features. Through the contrastive learning of the two views, the final node embedding can integrate graph structure and node attribute information.

In heterogeneous domain, HeCo [28] establishes network schema view and meta-path view to learn node embeddings, so as to capture both of local and global structure simultaneously. In addition, HeCo proposes a cross-view contrastive learning mechanism to extract positive and negative samples from the two views, so that the two views can supervise each other to learn the final node embedding. However, the disadvantage of HeCo is that node embedding is easily affected by view interaction noise. Therefore, HeCo cannot maintain the embedding consistency of each node in different views, and the embedding difference of with other nodes. SGL [29] establishes two views of U-I bipartite graph by various augmentation functions to achieve cross-view contrastive learning. However, the disadvantage of SGL is that it only models the structure of the bipartite graph and lacks semantic information.

3 Preliminary

In this section, we introduce important definitions related to HIN recommendations, as follows:

Definition 1 Heterogeneous information network (HIN). HIN is defined as a network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \phi, \varphi)$, where $\mathcal{V}$ and $\mathcal{E}$ denote sets of nodes and edges. HIN contains a node type mapping function $\phi: \mathcal{V} \rightarrow \mathcal{Q}$ and an edge type mapping function $\varphi: \mathcal{E} \rightarrow \mathcal{O}$, where $\mathcal{Q}(|\mathcal{Q}| > 1)$ represents the set of node types and $\mathcal{O}(|\mathcal{O}| > 1)$ denotes the set of edge types.

Figure 1a illustrates an example of HIN (Movielens). There are four types of nodes (“user”, “item”, “occupation” and “genre”), among which “item” can be regarded as “movie” in

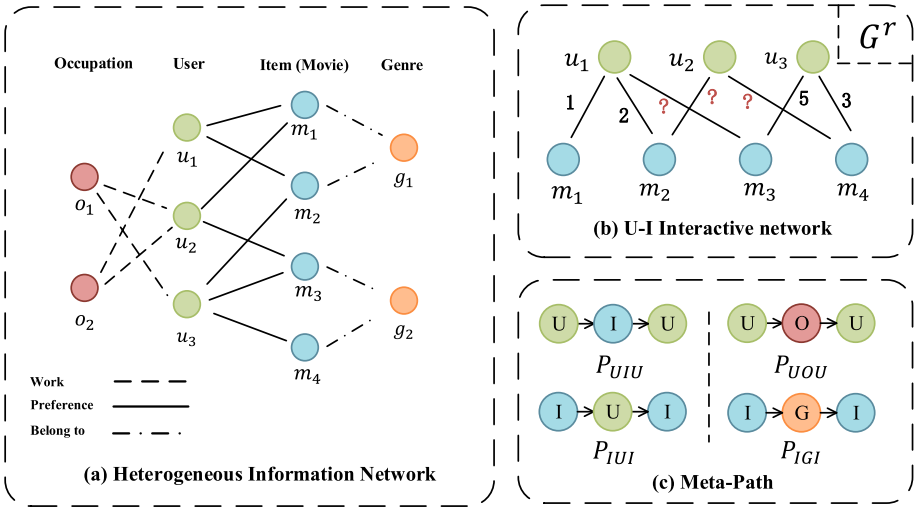

Fig. 1 A recommendation example of HIN (Movielens) and U-I interactive network (bipartite graph) and the illustrations of meta-path

Movielens dataset. Meanwhile, there are three types of relations ("work", "preference" and "belong to").

Definition 2 U-I interactive network (bipartite graph). The bipartite graph is a network which only contains user and item nodes taken from the HIN, noted as $G^r = (V, E, X, R, M)$. The node set $V = (U, I) \in \mathcal{V}$ of G^r contains user and item nodes, where $U = \{u_1, u_2, ..., u_{n_1}\}$ and $I = \{m_1, m_2, ..., m_{n_2}\}$; $E \in \mathcal{E}$ denotes set of edges, and e_{ij} denotes the edges between user u_i and item m_j; $X = (X_U, X_I)$ represents the learnable initial node features of users and items, where $X_U \in \mathbb{R}^{n_1 \times d}$ and $X_I \in \mathbb{R}^{n_2 \times d}$; $R \in \mathbb{R}^{n_1 \times n_2}$ is the adjacency matrix of G^r, where $R_{ij} = 1$ if $e_{ij} \in E$, otherwise $R_{ij} = 0$; $M \in \mathbb{R}^{n_1 \times n_2}$ is the rating matrix of G^r. A nonzero element M_{ij} represents a real rating $r \in \{1, ..., \mathcal{R}\}$ of user u_i to item m_j.

Figure 1b shows the interaction relationship (i.e., rating types) between user and item in the U-I interactive network. We utilize the known rating and user (item) embeddings obtained from model training to predict unknown rating between user and item.

Definition 3 Meta-path. A meta-path P is represented as $Q_1 \xrightarrow{O_1} Q_2 \xrightarrow{O_2} ... \xrightarrow{O_i} Q_{i+1}$, which describes a composite relation $O = O_1 \circ O_2 \circ ... \circ O_i$ between node types Q_1 and Q_{i+1}, where $\circ$ denotes the composition operator on relations.

Figure 1c shows four meta-paths extracted from HIN in Fig. 1a, which can describe different semantics information. For example, the meta-path P_{UIU} represents the relationship of the co-watching movies between two users; The meta-path P_{IUI} indicates that two movies are favored by the same user; The meta-path P_{UOU} represents two users with same occupation; The meta-path P_{IGI} represents two movies with the same genre.

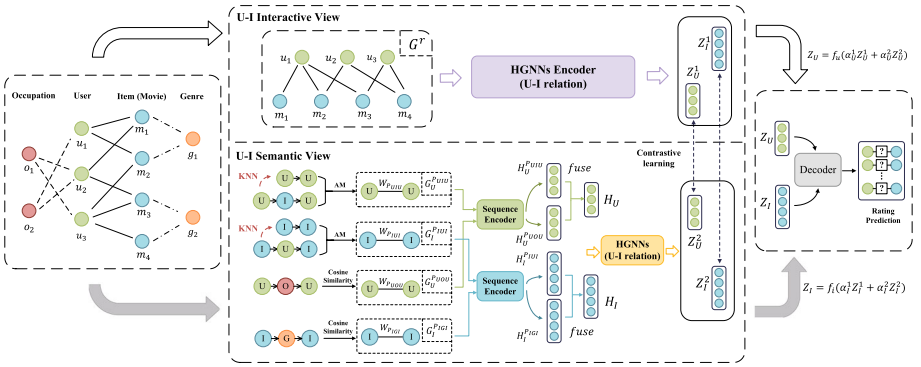

Fig. 2 The framework of the proposed GCL-SS model

4 Methodology

Since there are U-I direct association and UU (II) indirect association in HIN, and U-I direct association can express the interaction between user and item, while UU (II) indirect association can express the semantic information between users (items). Therefore, it is necessary to perform HIN-based recommendation from two aspects: U-I interaction association and U-I semantic association.

To this end, we propose the GCL-SS model. GCL-SS utilizes the mutual supervision of U-I interactive view and U-I semantic view to establish a cross-view contrastive learning mechanism. Among them, U-I interactive view can obtain node structural information, and U-I semantic view can obtain node semantic information. This contrastive learning mechanism can integrate the information of U-I interactive view and U-I semantic view, and also maintain the uniqueness and specificality of each view information. Figure 2 shows the overall framework of GCL-SS. GCL-SS includes U-I interactive view encoder, U-I semantic view encoder and bilinear decoder [30]. Firstly, GCL-SS encodes the U-I interaction network G^r in U-I interactive view to obtain user (item) structural embedding, and simultaneously encodes the HIN in U-I semantic view to obtain user (item) semantic embedding. Secondly, we adopt self-attention mechanism to fuse the user (item) structural embedding with semantic embedding to obtain the final user(item) embedding. At last, bilinear decoder utilizes user embeddings and item embeddings to reconstruct the interaction relationship (i.e., rating types) between user and item, so as to realize the rating prediction of user to item.

4.1 U-I Interactive View

In the U-I interaction network G^r, since the interaction between user and item can directly express user's preference for item, and the information such as neighborhood density and neighborhood location of user (item) nodes can represent its local structural information. Therefore, by encoding the local structural information of user (item) nodes in G^r, we can extract the node neighborhood environment (structural) features. In addition, since the U-I interaction network is a heterogeneous network with two types of nodes (user and item), we adopt heterogeneous GCN (or other HGNNs) as the basic module. This module captures the local structural information of user (item) nodes by k-hop neighborhood aggregation in G^r to obtain user (item) structural embeddings. Among them, the expression of user and item

structural embeddings Z^1 (Z_U^1 and Z_I^1) in U-I interactive view is as follows:

$$Z^1 = HGNNs(X_U, X_I, R) \tag{1}$$

where $X_U \in \mathbb{R}^{n_1 \times d}$ is a learnable initial user feature, $X_I \in \mathbb{R}^{n_2 \times d}$ is a learnable initial item feature; $R \in \mathbb{R}^{n_1 \times n_2}$ is the adjacency matrix of G^r.

4.2 U-I Semantic View

In HIN, there are various semantic associations between users (items). For example, if two users are connected by "occupation", it means that two users have the same work; if two users are connected by "item", it means that two users have the same preference. Therefore, the semantic association between users (items) can be regarded as UU (II) indirect association.

Since HIN is composed of multiple types of nodes and relations, and meta-path is an alternate category sequence of various node types and relations, which can reflect the semantic context association of nodes at both ends of the meta-path. Therefore, the semantic association information between user (item) nodes can be represented by meta-path. As shown in Fig. 1c, the meta-paths about user such as P_{UIU} and P_{UOU} can respectively represent two kinds of user semantic association information, so we can construct two channels (such as $G_U^{P_{UIU}}$ and $G_U^{P_{UOU}}$) of user semantic homogeneous network.

Since our model needs to encode each user (item) channel separately and obtain node semantic embedding, we mainly consider three problems in U-I semantic view: multi-channel construction for user (item) semantic homogeneous networks, multi-channel feature fusion for homogenous nodes, and semantic feature fusion for heterogenous nodes.

In terms of multi-channel construction, in order to construct user (item) channels, we use the meta-paths which start and end with the user or item node, such as P_{UOU} or P_{IGI}, as the basic meta-paths for our model. Firstly, we construct multiple channels of the user (item) semantic homogeneous network by meta-path decomposition of HIN. Secondly, we utilize augmentation module (AM) or cosine similarity to weight each channel of the user (item) semantic homogeneous network, so as to obtain multiple user-weighted channels $\{G_U^{P_{UIU}}, G_U^{P_{UOU}}, ...\}$ and item-weighted channels $\{G_I^{P_{IUI}}, G_I^{P_{IGI}}, ...\}$, where $G_U^{P_{UIU}}$ represents user-weighted semantic homogeneous network based on meta-path P_{UIU}.

In terms of multi-channel feature fusion, we first encode each user-weighted channel and item-weighted channel by TSE. Each user-weighted channel can obtain a user feature and each item-weighted channel can obtain an item feature. The TSE can strengthen the semantic environment context of nodes in each channel to strengthen the semantic association between users (items). Then, we adopt attention mechanism to fuse multiple channels of user (item) semantic homogeneous network to obtain user (item) semantic features.

In terms of semantic feature fusion, we utilize HGNNs to encode the fused node semantic features. By aggregating the semantic information of neighborhood heterogeneous nodes, we can obtain the final user (item) semantic embedding in U-I semantic view.

4.2.1 Multi-channel Construction Strategy

In this paper, we choose four basic meta-paths as shown in Table 1:

In U-I semantic view, when the meta-paths are P_{UIU} and P_{IUI}, we use augmentation module (AM) to weight the channels $G_U^{P_{UIU}}$ and $G_I^{P_{IUI}}$ (as shown in Fig. 2), so as to strengthen the preference association between users and the similarity association between items.

Table 1 Four basic meta-paths

Meta Path	Definition
P_{UIU}	Is a direct semantic association between users (through item association), which means that two users have common preferences.
P_{IUI}	Is a direct semantic association between items (through user association), which means that two items are liked by the same user.
P_{UOU}	Is an indirect semantic association between users (through occupation association), which means that two users have the same occupation.
P_{IGI}	Is an indirect semantic association between items (through genre association), which means that two items have the same genre.

Figure 3 shows the detailed process of the AM, we use KNN algorithm to obtain user social network G^s and item network G^c, which respectively represent user social relationship and item similarity relationship. In terms of user data augmentation, we utilize user social network G^s and U-I interactive network G^r to strengthen user preference association. If user u_i and user u_j both have the same preference and the social relationship, u_i and u_j are more similar. The similarity between u_i and u_j can be reflected by the number of same preferences between u_i and u_j. It can also be used as the weight between u_i and u_j in channel $G_U^{P_{UIU}}$. In addition, the process of item data augmentation is the same as user data augmentation. To this end, the adjacency matrices of channels $G_U^{P_{UIU}}$ and $G_I^{P_{IUI}}$ are respectively expressed as follows:

$$\begin{aligned} A_s &= (RR^T) \odot S \\ A_c &= (R^T R) \odot C \end{aligned} \tag{2}$$

where $R \in \mathbb{R}^{n_1 \times n_2}$ is the adjacency matrix of G^r; $S \in \mathbb{R}^{n_1 \times n_1}$ and $C \in \mathbb{R}^{n_2 \times n_2}$ respectively represent the adjacency matrices of G^s and G^c; $\odot$ represents Hadamard product. The edge weights W of $G_U^{P_{UIU}}$ and $G_I^{P_{IUI}}$ are respectively expressed as follows:

$$\begin{aligned} W^s_{u_i,u_j} &= \frac{A^s_{u_i,u_j}}{\sum A^s_{u_i,:}} \\ W^c_{m_i,m_j} &= \frac{A^c_{m_i,m_j}}{\sum A^c_{m_i,:}} \end{aligned} \tag{3}$$

where $A^s_{u_i,u_j}$ is the unnormalized weight between users in $G_U^{P_{UIU}}$ and $A^c_{m_i,m_j}$ is the unnormalized weight between items in $G_I^{P_{IUI}}$. According to $W^s_{u_i,u_j}$ and $W^c_{m_i,m_j}$, we can construct user-weighted channel $G_U^{P_{UIU}}$ and item-weighted channel $G_I^{P_{IUI}}$.

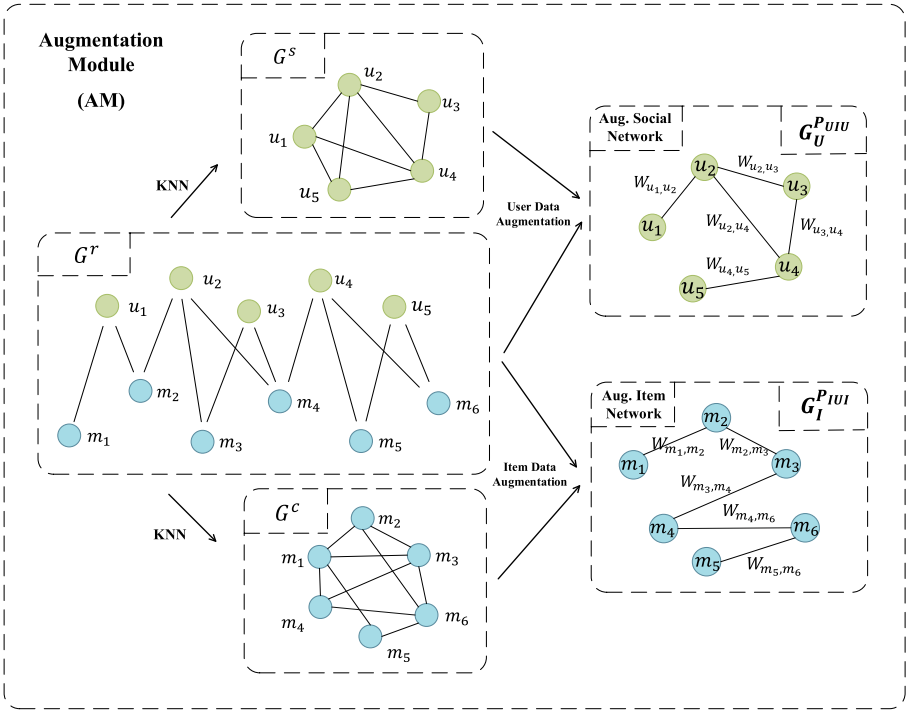

Fig. 3 Augmentation module (AM). It contains user data augmentation and item data augmentation

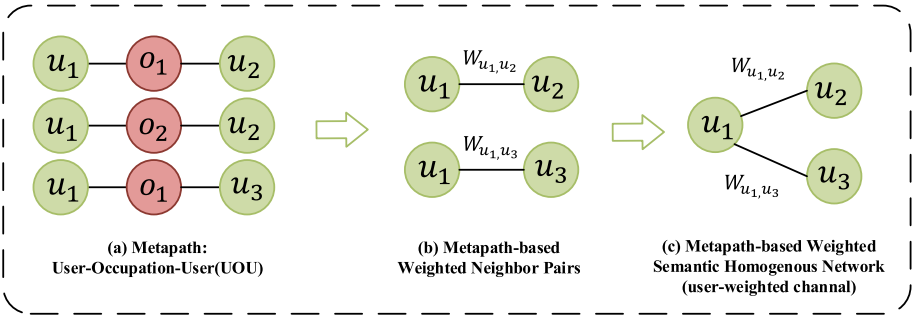

Fig. 4 Weighted process of semantic homogeneous network (Cosine Similarity)

When the meta-paths are P_{UOU} and P_{IGI}, we use cosine similarity to weight channels $G_U^{P_{UOU}}$ and $G_I^{P_{IGI}}$. As show in Fig. 4, for the meta-path P_{UOU}, Fig. 4a represents the meta-path instance of user u_1; Fig. 4b represents meta-path based weighted neighbor pairs; Fig. 4c represents meta-path based user-weighted channel. We use cosine similarity to calculate the similarity between user u_i and user u_j, and take this similarity as the weight W_{u_i,u_j} of node pairs (u_i and u_j):

$$W_{u_i,u_j} = \frac{|p_{u_i,u_j}^{P_{UOU}}|}{\sqrt{|N_{u_i}^{P_{UOU}}| \times |N_{u_j}^{P_{UOU}}|}} \tag{4}$$

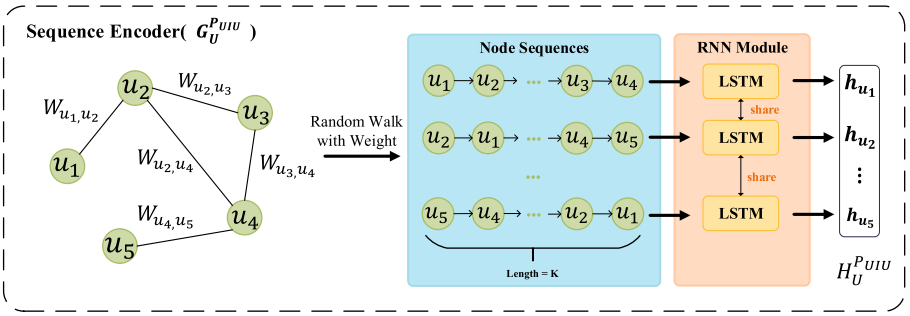

Fig. 5 Time sequence encoder (TSE). It contains Sample module and RNN

where $|p_{u_i,u_j}^{P_{UOU}}|$ represents the number of meta-path instances between u_i and u_j on meta-path P_{UOU}; $|N_{u_i}^{P_{UOU}}|$ and $|N_{u_j}^{P_{UOU}}|$ respectively represent the number of neighbors of u_i and u_j on meta-path P_{UOU}. According to W_{u_i,u_j}, we can construct user-weighted channel $G_U^{P_{UOU}}$. In addition, item-weighted channel $G_I^{P_{IGI}}$ can also be obtained.

4.2.2 Multi-channel Feature Fusion Strategy

Figure 5 is an example of time sequence encoding (TSE) for user-weighted channel $G_U^{P_{UIU}}$. Firstly, each user node performs random walk sampling with edge weight W to obtain a node sequence (length=K). Next, we utilize an RNN module (such as LSTM and Bi-LSTM) to encode the sampled node sequences to capture the environment context of all user nodes in $G_U^{P_{UIU}}$. For example, if a node sequence is $[u_1, u_i, ..., u_k]$, the corresponding node feature sequence is $[x_{u_1}, x_{u_i}, ..., x_{u_k}]$. We take this node feature sequence as the input of RNN, and the output h_{u_1} as the feature of user u_1. The expression of h_{u_1} is as follows:

$$h_{u_1} = LSTM([x_{u_1}, x_{u_i}, ..., x_{u_k}]) \tag{5}$$

According to Eq. (5), we can obtain the output $H_U^{P_{UIU}}$ of RNN for the node sequences of all users, where $H_U^{P_{UIU}} = \{h_{u_1}, ..., h_{u_5}\}$ are the features of node sequences. The $H_U^{P_{UIU}}$ are also treated as the features of head nodes in the node sequences. In addition, other weighted channels ($G_I^{P_{IUI}}$, $G_U^{P_{UOU}}$ and $G_I^{P_{IGI}}$) can respectively obtain node features ($H_I^{P_{IUI}}$, $H_U^{P_{UOU}}$ and $H_I^{P_{IGI}}$) by the TSE. In order to ensure the encoding consistency of homogenous node, we share the TSE parameters of multiple channels in user (item) semantic homogenous network.

Finally, we utilize attention mechanism to fuse the multi-channel node features of the user semantic homogenous network G_U or the item semantic homogenous network G_I. This attention mechanism can learn different semantic weights according to different meta-paths. In summary, we can obtain the node features H_U and H_I after multi-channel feature fusion:

$$H_U = \sigma([H_U^{P_{UIU}} \| H_U^{P_{UOU}}] \cdot W_U) \tag{6}$$

$$H_I = \sigma([H_I^{P_{IUI}} \| H_I^{P_{IGI}}] \cdot W_I) \tag{7}$$

where σ is a sigmoid function; $W_U \in \mathbb{R}^{2d \times d}$ and $W_I \in \mathbb{R}^{2d \times d}$ are learnable weights.

4.2.3 Semantic Feature Fusion Strategy

Based on the multi-channel node feature fusion strategy, we respectively take H_U and H_I as new user and item features on the U-I interaction network G^r. Then, we encode the G^r by heterogeneous GCN to obtain user (item) semantic embedding. This method makes the obtained node semantic embedding contain both self-semantic information and neighborhood semantic information. The expression of user and item semantic embeddings Z^2 (Z_U^2 and Z_I^2) in U-I semantic view is as follows:

$$Z^2 = HGNNs(H_U, H_I, R) \tag{8}$$

where $H_U \in \mathbb{R}^{n_1 \times d}$ and $H_I \in \mathbb{R}^{n_2 \times d}$ respectively represents node features of G_U and G_I after multi-channel feature fusion; $R \in \mathbb{R}^{n_1 \times n_2}$ is the adjacency matrix of G^r.

4.3 Self-supervised Contrastive Learning

In this section, we utilize contrastive learning mechanism to supervise the U-I interaction view and the U-I semantic view for each other. Such supervision mechanism can effectively preserve the information of the HIN to improve recommendation performance. Figure 6 shows the comparison scheme of user nodes between U-I interactive view and U-I semantic view, which can be divided into intra-view contrast and inter-view contrast. Among them, the same node in other view is defined as positive sample, and different nodes in the two views are defined as negative samples. The objective of contrastive learning is to maximize the similarity between positive samples and minimize the similarity between negative samples. This objective can integrate the information of U-I interactive view and U-I semantic view and reduce the influence of view interaction noise. According to this objective, the contrastive loss of node u_i is defined as follows:

$$\ell(u_i^1, u_i^2) = -log \frac{e^{\theta(u_i^1, u_i^2)/\tau}}{e^{\theta(u_i^1, u_i^2)/\tau} + \sum_{k \neq i} e^{\theta(u_i^1, u_k^1)/\tau} + \sum_{k \neq i} e^{\theta(u_i^1, u_k^2)/\tau}} \tag{9}$$

where τ is a control parameter. We define $\theta(u_i^1, u_i^2) = s(z_{u_i}^1, z_{u_i}^2)$, where $s(\cdot, \cdot)$ is the cosine similarity. Since the contrastive loss of node u_i in the two views are symmetric, the loss for another view can be defined as $\ell(u_i^2, u_i^1)$. Therefore, the user contrastive loss can be defined as the average over all user nodes:

$$\mathcal{L}_U = \frac{1}{2N_1} \sum_{i=1}^{N_1} [\ell(u_i^1, u_i^2) + \ell(u_i^2, u_i^1)] \tag{10}$$

where N_1 is the number of users. Similarly, the item contrastive loss can be expressed as follows:

$$\mathcal{L}_I = \frac{1}{2N_2} \sum_{i=1}^{N_2} [\ell(m_i^1, m_i^2) + \ell(m_i^2, m_i^1)] \tag{11}$$

where N_2 is the number of items.

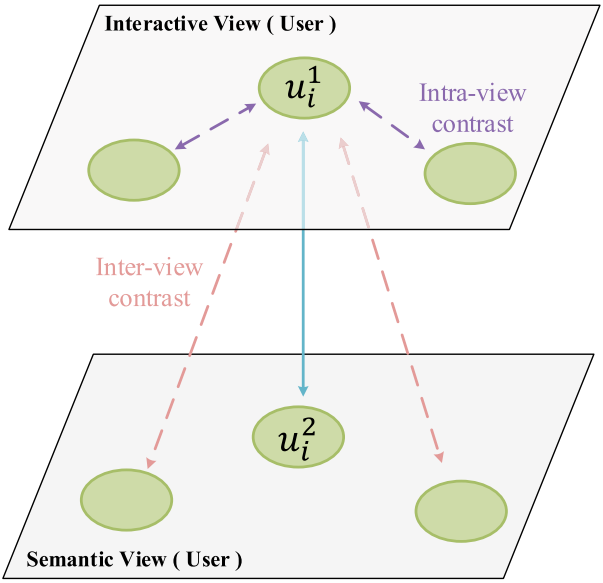

Fig. 6 Contrastive learning of two views (user)

4.4 Bilinear Decoder

The purpose of bilinear decoder is to reconstruct the interaction relationship (i.e., edge types) in the U-I interaction network G^r by user and item embeddings, and achieve rating prediction by the reconstructed edge types. Figure 7 shows the internal structure of the bilinear decoder, Z_U (Z_I) is the final node embedding of integrating user (item) structural embedding and semantic embedding through self-attention mechanism. The integration process is as follows: firstly, we respectively obtain the attention weights of the user (item) structural and semantic embedding in the two views:

$$\begin{aligned} \alpha_U^1 &= \frac{exp(Z_U^1 \cdot W_1)}{exp(Z_U^1 \cdot W_1 + Z_U^2 \cdot W_2)} \\ \alpha_U^2 &= \frac{exp(Z_U^2 \cdot W_2)}{exp(Z_U^1 \cdot W_1 + Z_U^2 \cdot W_2)} \end{aligned} \tag{12}$$

$$\begin{aligned} \alpha_I^1 &= \frac{exp(Z_I^1 \cdot W_3)}{exp(Z_I^1 \cdot W_3 + Z_I^2 \cdot W_4)} \\ \alpha_I^2 &= \frac{exp(Z_I^2 \cdot W_4)}{exp(Z_I^1 \cdot W_3 + Z_I^2 \cdot W_4)} \end{aligned} \tag{13}$$

where W_1, W_2, W_3 and W_4 are learnable weights. Secondly, we integrate user (item) structural and semantic embedding by these weights:

$$Z_U = f_u(\alpha_U^1 Z_U^1 + \alpha_U^2 Z_U^2) \tag{14}$$

$$Z_I = f_i(\alpha_I^1 Z_I^1 + \alpha_I^2 Z_I^2) \tag{15}$$

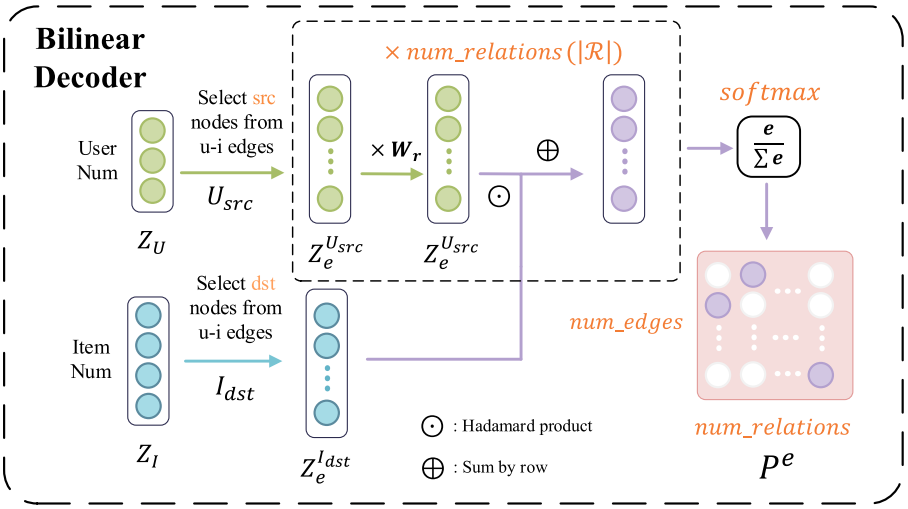

Fig. 7 Bilinear decoder. It obtains the probabilities of all ratings for all edge

where f_u or f_i is a projection head, which is an MLP with two hidden layers.

In addition, we respectively define the source and destination nodes of all U-I edges as U_{src} and I_{dst}, and the corresponding node embeddings are $Z_e^{U_{src}}$ and $Z_e^{I_{dst}}$. We reconstruct the interaction relationship (edge types) by using bilinear operations (i.e., the corresponding process of source and destination nodes in Fig. 7). Then, we obtain the normalized probabilities P^e of all ratings for each edge by SoftMax function. The expression of P^e is as follows:

$$P_r^e = \frac{Sum((Z_e^{U_{src}} \cdot W_r) \odot Z_e^{I_{dst}})}{\sum_{r=1}^{\mathcal{R}} Sum((Z_e^{U_{src}} \cdot W_r) \odot Z_e^{I_{dst}})} \tag{16}$$

$$P^e = Softmax(Concat(P_1^e, P_2^e, ..., P_{\mathcal{R}}^e)) \tag{17}$$

where W_r is a learnable weight; $\odot$ represents Hadamard product; Sum ($\oplus$) represents sum by row; P_r^e represents the probabilities of all observable edges when the rating type is r.

4.5 Model Training

In this paper, we take rating prediction task in recommendation as target, and utilize contrastive loss and rating prediction loss for end-to-end training. Among them, the overall contrastive loss $\mathcal{L}_C$ includes user contrastive loss and item contrastive loss. Therefore, the expression of $\mathcal{L}_C$ is as follows:

$$\mathcal{L}_C = \mathcal{L}_U + \mathcal{L}_I \tag{18}$$

In addition, we take cross-entropy classification loss as rating prediction loss. We can obtain accurate rating type of user to item by minimizing classification loss:

$$\mathcal{L}_{Rec} = -\frac{1}{|E_{ij}|} \sum_{(i,j) \in E_{ij}} \sum_{r=1}^{\mathcal{R}} Y[M_{ij} = r] \, log \; p(P_{ij}^e = r) \tag{19}$$

where $|E_{ij}|$ represents the number of observable edges in G^r; $Y[M_{ij} = r]$ represents whether the real rating of e_{ij} belong to rating type r, if it belongs, $Y = 1$, otherwise $Y = 0$; P_{ij}^e represents the probability of edge e_{ij} belonging to each rating type; $p(P_{ij}^e = r)$ represents the probability of edge e_{ij} belonging to rating type r.

In summary, the overall objective function can be defined as: $\mathcal{L} = \mathcal{L}_{Rec} + \beta\mathcal{L}_C$, where β is a balance factor, and represents the weight of contrastive loss.

5 Model Extension

In U-I semantic view, whether the rich semantic information in HIN can be effectively preserved will directly affect the recommendation performance. Therefore, in order to improve the recommendation performance, we propose an extended model, GCL-SS$_{AE}$. This model not only considers self-supervised contrastive learning between node structural embedding and semantic embedding, but also considers the structural consistency of semantic homogeneous networks in U-I semantic view.

Figure 8 is an improved scheme in U-I Semantic view based on GCL-SS, namely GCL-SS$_{AE}$. In U-I Semantic view, we reconstruct the user (item) feature H_U (H_I) after multi-channel fusion to obtain a new adjacency matrix $\hat{A}_U^{l_1}$ ($\hat{A}_I^{l_2}$). The expressions of $\hat{A}_U^{l_1}$ and $\hat{A}_I^{l_2}$ are as follows:

$$\begin{aligned} \hat{A}_U^{l_1} &= \sigma(H_U \cdot W_U^{l_1} \cdot H_U^T) \\ \hat{A}_I^{l_2} &= \sigma(H_I \cdot W_I^{l_2} \cdot H_I^T) \end{aligned} \tag{20}$$

where σ is a sigmoid function; $l_1 \in [1, L_1]$ represents meta-path type of user, such as P_{UIU}, P_{UOU}, and so on; $l_2 \in [1, L_2]$ represents meta-path type of item, such as P_{IUI}, P_{IGI}, and so on; $W_U^{l_1}$ and $W_I^{l_2}$ are learnable weights. Then, we adopt auto-encode mechanism to compute the reconstruction loss $\mathcal{L}_{AE}$ between the new adjacency matrix and the original adjacency matrix for each channel of semantic homogeneous network as part of the objective function $\mathcal{L}$. The expression of $\mathcal{L}_{AE}$ is as follows:

$$\mathcal{L}_{AE} = \sum_{l_1=1}^{L_1} dis(A_U^{l_1}, \hat{A}_U^{l_1}) + \sum_{l_2=1}^{L_2} dis(A_I^{l_2}, \hat{A}_I^{l_2}) \tag{21}$$

where $dis(\cdot, \cdot)$ is the binary norm distance of two matrices.

Through this auto-encode mechanism, the reconstructed network structure can be more similar to the original network, so as to preserve semantic information in HIN effectively.

6 Experiments

6.1 Experiments Setup

1) Dataset

In our experiments, we confirm the feasibility of the GCL-SS model using three real-world datasets, namely, Movielens dataset, Douban Book dataset, and Yelp dataset. They respectively cover the domains of movies, books, and business. The Movielens dataset contains 943 users, 1682 movies, and 100,000 movie ratings ranging from 1 to 5. The Douban Book dataset comprises 13,024 users, 22,347 books, and 792,062 ratings ranging from 1 to 5. The

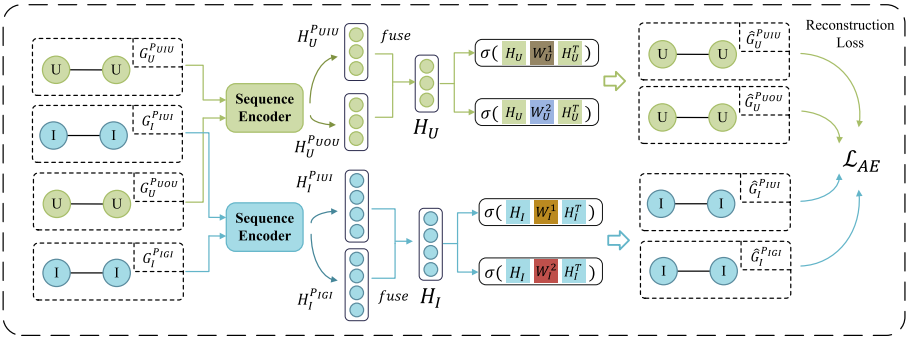

Fig. 8 An improved scheme in U-I Semantic view based on GCL-SS (GCL-SS$_{AE}$)

Table 2 Summary of three datasets from different domains

Dataset	Density	Node type	# Nodes	Edge type	# Edges	Meta path
Movielens	6.30%	**User (U)**	943	U–M	100,000	UMU
		Movie (M)	1682	U–U	47,150	MUM
		Occupation (O)	21	U–O	943	UOU
		Genre (G)	18	U–A	943	MGM
		Age (A)	8	M–M	82,798	–
				M–G	2861	
Douban Book	0.27%	**User (U)**	13,024	U–B	792,062	UBU
		Book (B)	22,347	U–U	169,150	BUB
		Location (L)	453	U–L	10,592	ULU
		Author (A)	10,805	B–A	21,905	BAB
		Publisher (P)	1,815	B–P	21,773	–
		Year (Y)	64	B–Y	21,192	
Yelp	0.08%	**User (U)**	16,239	U–B	198,397	UBU
		Business (B)	14,284	U–U	158,590	BUB
		Compliment (Co)	11	U–Co	76,875	UCoU
		City (Ci)	47	B–Ci	14,267	BCaB
		Category (Ca)	511	B–Ca	40,009	–

Bold represents the two types of nodes in the U-I interactive view

Yelp dataset comprises 16,239 users, 14,282 businesses, and 198,397 ratings ranging from 1 to 5. In addition, these datasets also contain other types of node information. We present the detailed information of the three datasets in Table 2, including the rating densities, the node types, the number of nodes, the edge types, the number of edges and the meta-paths we used.

6.1.1 Evaluation metrics

In order to evaluate the performance of different recommendation models and our model, we utilize the popular root mean square error (RMSE) and mean absolute error (MAE) as the evaluation metrics to evaluate the performance of the rating prediction. If the RMSE and

MAE values are lower, the performance of the model is better. The expression of RMSE and MAE are as follows:

$$RMSE = \sqrt{\frac{1}{|D_{test}|} \sum_{(i,j) \in D_{test}} (r_{ij} - \widehat{r_{ij}})^2} \tag{22}$$

$$MAE = \frac{1}{|D_{test}|} \sum_{(i,j) \in D_{test}} |r_{ij} - \widehat{r_{ij}}| \tag{23}$$

where $|D_{test}|$ represents the number of ratings for the test set; r_{ij} is a real rating of user u_i to item m_j, and $\widehat{r_{ij}}$ is a predicted rating of user u_i to item m_j.

6.1.2 Baselines

To evaluate the rating prediction performance of our model, we compare GCL-SS with nine state-of-the-art recommendation baselines to confirm the feasibility of the GCL-SS.

PMF [31]: This model is a traditional probabilistic matrix factorization model, which uses the product of two low-rank matrices (collaborative filtering) to obtain rating of user to item.

HeteMF [32]: This model is an MF based recommendation model, which uses meta-paths to calculate entities (such as user and item) similarity in HINs.

SoMF [33]: This model utilizes social network to improve the performance of recommender system. It proposes "social regularization" to extract information about social relationships.

SemRec [34]: This model incorporates weighted HIN into the collaborative filtering method to achieve rating prediction.

HERec [14]: This model utilizes Deep Walk to encode homogeneous networks obtained by the HIN decomposition, and proposes three fusion methods to optimize the MF model for recommendation.

NCF [35]: This model is a typical graph neural network recommendation method, which uses MLP to model user-item interaction.

MCRec [36]: This model utilizes CNN to encode meta-path and learns complex relationship from HIN to improve recommendation performance.

NGCF [5]: This model is one of the state-of-the-art graph neural network methods for recommendation, which uses a graph convolution network to model user-item interaction.

AMERec [15]: This model is an attention-aware meta-path based network embedding for HIN based recommendation. It decomposes the HIN according to the meta-path type to obtain homogeneous network, and utilizes self-attention mechanism to assign a personalized weight for each meta-path based weighted homogenous network, so as to obtain user and item embedding.

6.1.3 Parameter and Environment Settings

We divide each dataset into training and testing sets. We set 80% training rates for Movielens dataset and Douban Book dataset, and 90% training rates for more sparse Yelp dataset. We set the random walk length as 10 and the contrastive loss weight as 0.2 for all datasets. In addition, we randomly initialize the learnable user and item features, and set the initial feature dimension of user and item to 256, the dimension of hidden layers to 64, and the activation function to 'tanh'. Besides, we use cross entropy loss as loss function for rating prediction

Table 3 Rating prediction performance for GCL-SS, baselines and variants

Methods	Dataset					
	Movielens		Douban book		Yelp	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
PMF	1.0407	0.8324	0.7414	0.5774	1.4268	1.0412
SoMF	0.9802	0.7716	0.7302	0.5756	1.3392	1.0095
HeteMF	0.9503	0.7400	0.7360	0.5740	1.2549	0.9487
SemRec	0.9432	0.7312	0.7283	0.5675	1.1637	0.9043
HERec	0.9297	0.7338	**0.6811**	0.5502	1.0907	0.8395
NCF	0.9303	0.7354	0.7287	0.5737	1.0556	0.8254
MCRec	0.9287	0.7321	0.7012	0.5745	1.0823	0.8378
NGCF	0.9214	0.7220	0.7123	0.5769	1.0619	0.8694
AMERec	0.9199	0.7184	0.6912	0.5496	1.0506	0.8067
GCL-SS(k=10)	**0.8988**	**0.6996**	0.6899	**0.5357**	**1.0476**	**0.8000**
GCL-SS$_{NC}$(k=10)	0.9041	0.7069	0.6949	0.5420	1.0652	0.8130
GCL-SS$_{GCN}$(k=10)	0.9024	0.7043	0.6897	0.5371	1.0564	0.8091
GCL-SS$_{NW}$(k=10)	0.9027	0.7050	0.6910	0.5385	1.0543	0.8036
GCL-SS$_{UI}$(k=10)	0.9033	0.7050	0.6910	0.5366	1.0508	0.8023

Bold represents the optimal results of the model

and Adam as optimizer, and set the learning rate to 0.005. Our model is implemented by PYG framework and all experiments are performed using a computer with GPU (NVIDIA GeForce RTX 3090) and CPU (Intel i7-10700K).

6.2 Comparison of Related Methods

In this section, we conduct rating prediction experiments on three datasets. Table 3 shows the results (RMSE and MAE) of GCL-SS and baselines on three different datasets. In addition, Table 3 also shows the results of ablation experiments on GCL-SS and GCL-SS variants. Among them, GCL-SS$_{NC}$ is a variant of GCL-SS, which does not consider contrastive learning; GCL-SS$_{GCN}$ replaces TSE with GCN in U-I semantic view; GCL-SS$_{NW}$ does not consider the weights in semantic homogeneous network; GCL-SS$_{UI}$ uses only meta-paths P_{UIU} and P_{IUI}.

From the results in Table 3, we can draw the following conclusions:

(1) Compared with the traditional matrix factorization methods (i.e., PMF and SoMF), the HIN based methods (i.e., HeteMF, SemRec, HERec, MCRec and AMERec) and the collaborative filtering methods (i.e., NCF and NGCF) have significant performance improvements. In addition, our GCL-SS model generally performs better than the above nine baseline methods on two evaluation metrics (i.e., RMSE and MAE). However, the performance of our GCL-SS model is worse than that of HERec on the Douban Book dataset. The reason is that the excessive sampling by our sampler leads to poor model effect. We verified this explanation through experiments in Sect. 6.3. When the dataset is sparse, we only need a small sampling length to fully extract neighborhood information.

(2) Figure 9 shows the visualization results of the ablation experiments in Table 3 We can observe that the performance of GCL-SS is significantly better than the other four variants. The comparison results of GCL-SS and GCL-SS$_{NC}$ show that the contrastive learning

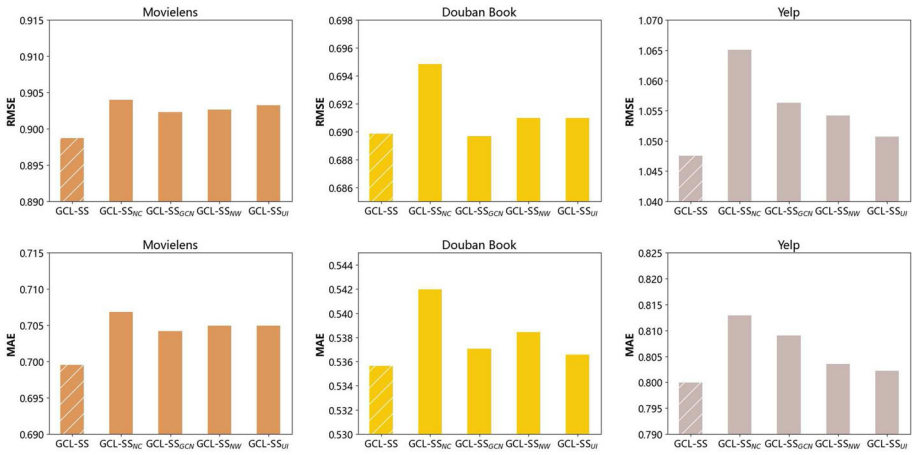

Fig. 9 The ablation experiment results of GCL-SS on three datasets

mechanism we established can effectively preserve the information of HIN. The comparison results of GCL-SS and GCL-SS$_{GCN}$ show that the TSE can strengthen the semantic association between users (item) by strengthening the semantic environment context of nodes in HIN. Therefore, TSE can further improve the model performance of rating prediction. The comparison results of GCL-SS and GCL-SS$_{NW}$ show that weighting the semantic homogeneous networks can reduce the loss of semantic information. The comparison results of GCL-SS and GCL-SS$_{UI}$ show that using more meta-paths to decompose HIN can effectively improve the performance of rating prediction.

6.3 Model Analysis

6.3.1 Impact of the Different Sampling Lengths

Since different sampling lengths in the TSE may affect the performance of rating prediction, we further analyze the impact of different sampling lengths through experiments. Under different sampling lengths, we set all the contrastive loss weights as 0.2. As shown in Figs. 10 and 11, we can observe that the rating prediction performance of GCL-SS is related to the sampling length on different datasets. The reason is that the three datasets have different link densities (as shown in Table 2). When the dataset is dense (such as Movielens), GCL-SS needs a larger sampling length (k=10) to fully obtain the information of multi-hop nodes. When the dataset is sparse (such as Yelp), only a small sampling length (k=7) is required to fully extract neighborhood information.

6.3.2 Impact of the Different Contrastive Loss Weights

Since different contrastive loss weights may also affect the performance of rating prediction, we further analyze the impact of different contrastive loss weights through experiments. Under different contrastive loss weights, we set all the sampling lengths as 10. As shown in Fig. 12, we can observe that GCL-SS have different requirements for contrastive loss weight on different datasets. The reason is that the three datasets have different sparsity. When the

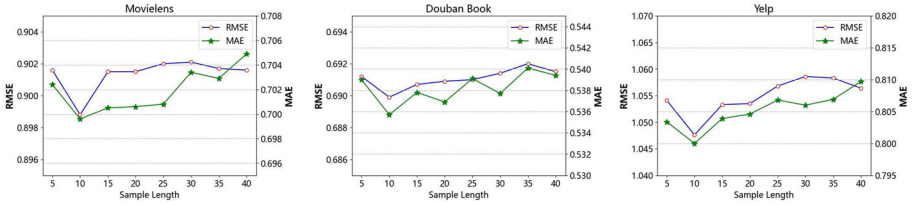

Fig. 10 The impact of different sampling lengths on three datasets (The step length is 5)

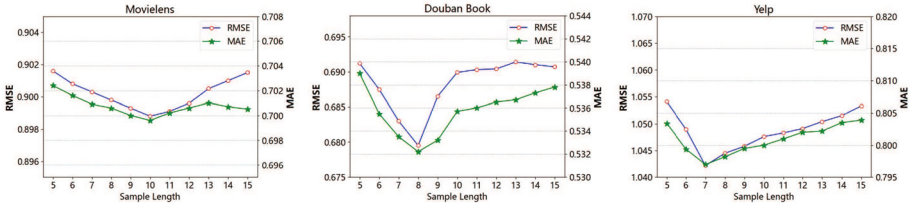

Fig. 11 The impact of different sampling lengths on three datasets (The step length is 1)

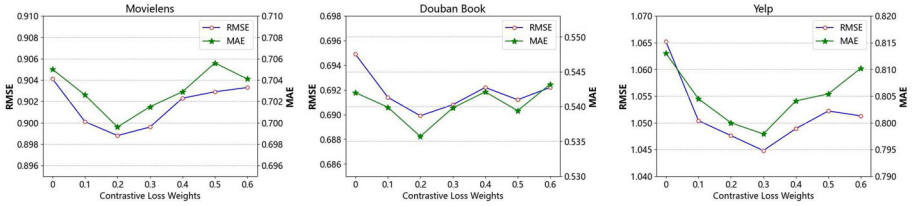

Fig. 12 The impact of different contrastive loss weights on three datasets

dataset is dense (such as Movielens), only a small contrastive loss weight ($\beta = 0.2$) is required to preserve the effective information of HIN. When the dataset is sparse (such as Yelp), a larger contrastive loss weight ($\beta = 0.3$) is required to preserve the effective information of HIN.

In addition, when we decrease or increase the contrastive loss weight, the model performance will decrease. The main reason is that when the contrastive loss weight is too small, it will lead to insufficient integration of node structural embedding and semantic embedding. When the contrastive loss weight is too large, GCL-SS will focus more on the integration of node structural and semantic embedding, rather than getting more suitable node structural and semantic embedding.

6.4 Model Extension Analysis

In this section, we respectively validate the impact of the extended model GCL-SS$_{AE}$ on rating prediction performance on three datasets. As is shown in Table 3, since AMERec is a rather competitive method on three datasets, we compare our extended model with base model (GCL-SS) and AMERec in rating prediction task. The results are shown in Table 4.

From Table 4, we can observe that the proposed extended model GCL-SS$_{AE}$ performs better than AMERec, and performs better than GCL-SS on the Movielens dataset. However, GCL-SS$_{AE}$ can only achieve slightly worse or comparable results than GCL-SS on the

Table 4 Evaluating the rating prediction of GCL-SS$_{AE}$ on three datasets (compared with AMERec and GCL-SS)

Dataset	Metrics	AMERec	GCL-SS	GCL-SS$_{AE}$
Movielens	RMSE	0.9199	0.8988	**0.8904**
	MAE	0.7184	0.6996	**0.6904**
Douban book	RMSE	0.6912	**0.6899**	0.6903
	MAE	0.5496	**0.5357**	0.5374
Yelp	RMSE	1.0506	1.0476	**1.0447**
	MAE	0.8067	0.8000	**0.7991**

Bold represents the optimal results of the model

Douban Book and Yelp datasets. The reason is that the link density of Douban Book and Yelp datasets is too small, the distribution of edges around nodes is sparse, and the semantic information preserved by the auto-encode mechanism is less. Therefore, the auto-encode mechanism does not significantly improve the model performance on the Douban Book and Yelp datasets.

7 Conclusions and Future Work

In this paper, we propose a graph contrastive learning model based on structural and semantic view for HIN recommendation (GCL-SS). GCL-SS adopts U-I interactive view and U-I semantic view to respectively obtain the structural embedding and semantic embedding of user (item) nodes. Then we establish a self-supervised contrastive learning mechanism for the two views, so that the information of the two views can be integrated and the uniqueness and specificality of each view can be preserved. In U-I semantic view, we innovatively use TSE to encode multiple semantic homogeneous networks, to strengthen the semantic association between user (item) nodes and further improve the performance of GCL-SS. In addition, we design an extended model (GCL-SS$_{AE}$) based on GCL-SS. This model can reduce the loss of semantic information by strengthening the structural consistency of semantic homogeneous networks.

The experimental results on three real datasets confirm that the GCL-SS model performs better than state-of-the-art recommended methods. Experimental results of the extended model (GCL-SS$_{AE}$) prove that the semantic association information of nodes in HIN can be effectively preserved by strengthening the structural consistency of semantic homogeneous networks. This method is helpful to improve the performance of rating prediction.

In GCL-SS, U-I interactive view adopts normal heterogeneous GCN as the basic module, which may not sufficiently extract the neighborhood structural information of the nodes. In addition, the sampling strategy in TSE can be improved to fully reflect the semantic environmental context of the nodes in the future work.

Acknowledgements This work was supported in part by the Natural Science Foundation of Zhejiang Province (Grant No.LY22F020001), the 3315 Plan Foundation of Ningbo (Grant No.2019B-18-G).

Data Availability The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Sun Y, Han J (2013) Mining heterogeneous information networks: a structural analysis approach. Acm Sigkdd Explor Newslett 14(2):20–28
2. Hu W, Liu B, Gomes J, Zitnik M, Liang P, Pande V, Leskovec J (2019) Strategies for pre-training graph neural networks. arXiv preprint arXiv:1905.12265
3. Davis AP, Grondin CJ, Johnson RJ, Sciaky D, King BL, McMorran R, Wiegers J, Wiegers TC, Mattingly CJ (2017) The comparative toxicogenomics database: update 2017. Nucleic Acids Res 45(D1):972–978
4. Wu L, Sun P, Fu Y, Hong R, Wang X, Wang M (2019) A neural influence diffusion model for social recommendation. In: Proceedings of the 42nd International ACM SIGIR conference on research and development in information retrieval, pp. 235–244
5. Wang X, He X, Wang M, Feng F, Chua T-S (2019) Neural graph collaborative filtering. In: Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval, pp. 165–174
6. He X, Deng K, Wang X, Li Y, Zhang Y, Wang M (2020) Lightgcn: simplifying and powering graph convolution network for recommendation. In: Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval, pp. 639–648
7. Zhang C, Song D, Huang C, Swami A, Chawla NV (2019) Heterogeneous graph neural network. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, pp. 793–803
8. Dong Y, Chawla NV, Swami A (2017) metapath2vec: scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 135–144
9. Perozzi B, Al-Rfou R, Skiena S (2014) Deepwalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining, pp. 701–710
10. Huang Z, Mamoulis N (2017) Heterogeneous information network embedding for meta path based proximity. arXiv preprint arXiv:1701.05291
11. Fu T-y, Lee W-C, Lei Z (2017) Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning. In: Proceedings of the 2017 ACM on conference on information and knowledge management, pp. 1797–1806
12. Shao Z, Xu Y, Wei W, Wang F, Zhang Z, Zhu F(2021) Heterogeneous graph neural network with multi-view representation learning. arXiv preprint arXiv:2108.13650
13. Cai X, Shang J, Hao F, Liu D, Zheng L (2021) Hmsg: Heterogeneous graph neural network based on metapath subgraph learning. arXiv preprint arXiv:2109.02868
14. Shi C, Hu B, Zhao WX, Philip SY (2018) Heterogeneous information network embedding for recommendation. IEEE Trans Knowl Data Eng 31(2):357–370
15. Yan S, Wang H, Li Y, Zheng Y, Han L (2021) Attention-aware metapath-based network embedding for HIN based recommendation. Expert Syst Appl 174:114601
16. Tian Y, Krishnan D, Isola P (2020) Contrastive multiview coding. In: European conference on computer vision, pp. 776–794 . Springer
17. Shi C, Li Y, Zhang J, Sun Y, Philip SY (2016) A survey of heterogeneous information network analysis. IEEE Trans Knowl Data Eng 29(1):17–37
18. Yang C, Xiao Y, Zhang Y, Sun Y, Han J (2020) Heterogeneous network representation learning: a unified framework with survey and benchmark. IEEE Trans Knowl Data Eng 34:4854

19. Chen T, Kornblith S, Norouzi M, Hinton G (2020) A simple framework for contrastive learning of visual representations. In: International conference on machine learning, pp. 1597–1607 . PMLR
20. He K, Fan H, Wu Y, Xie S, Girshick R (2020) Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9729–9738
21. Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
22. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R (2019) Albert: a lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942
23. Velickovic P, Fedus W, Hamilton WL, Liò P, Bengio Y, Hjelm RD (2019) Deep graph infomax. ICLR (Poster) 2(3):4
24. Linsker R (1988) Self-organization in a perceptual network. Computer 21(3):105–117
25. Hassani K, Khasahmadi AH (2020) Contrastive multi-view representation learning on graphs. In: International conference on machine learning, pp. 4116–4126 . PMLR
26. Zeng J, Xie P (2021) Contrastive self-supervised learning for graph classification. In: Proceedings of the AAAI conference on artificial intelligence, 35, 10824–10832
27. Zhu Y, Xu Y, Yu F, Liu Q, Wu S, Wang L (2021) Graph contrastive learning with adaptive augmentation. In: Proceedings of the web conference 2021, pp. 2069–2080
28. Wang X, Liu N, Han H, Shi C (2021) Self-supervised heterogeneous graph neural network with co-contrastive learning. In: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining, pp. 1726–1736
29. Wu J, Wang X, Feng F, He X, Chen L, Lian J, Xie X (2021) Self-supervised graph learning for recommendation. In: Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval, pp. 726–735
30. Berg Rvd, Kipf TN, Welling M (2017) Graph convolutional matrix completion. arXiv preprint arXiv:1706.02263
31. Salakhutdinov R, Mnih A (2007) Probabilistic matrix factorization. NIPS, pp. 1257–1264
32. Adomavicius G, Mobasher B, Ricci F, Tuzhilin A (2011) Context-aware recommender systems. In: Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, Lausanne, Switzerland, October 23-25, 2008, 335–336
33. Ma H, Zhou D, Liu C, Lyu MR, King I (2011) Recommender systems with social regularization. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, pp. 287–296
34. Shi C, Zhang Z, Luo P, Yu PS, Yue Y, Wu B (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, pp. 453–462
35. He X, Liao L, Zhang H, Nie L, Hu X, Chua T-S (2017) Neural collaborative filtering. In: Proceedings of the 26th international conference on world wide web, pp. 173–182
36. Hu B, Shi C, Zhao WX, Yu PS (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, pp. 1531–1540