

Collaborative Interest-aware Graph Learning for Group Identification

Rui Zhao^{1,2}, Beihong Jin ^{1,2}, Beibei Li³, and Yiyuan Zheng^{1,2}

¹ Institute of Software, Chinese Academy of Sciences, Beijing, China

² University of Chinese Academy of Sciences, Beijing, China

³ College of Computer Science, Chongqing University, Chongqing, China
Beihong@iscas.ac.cn

Abstract. With the popularity of social media, an increasing number of users are joining group activities on online social platforms. This elicits the requirement of group identification (GI), which is to recommend groups to users. We reveal that users are influenced by both group-level and item-level interests, and these dual-level interests have a collaborative evolution relationship: joining a group expands the user’s item interests, further prompting the user to join new groups. Ultimately, the two interests tend to align dynamically. However, existing GI methods fail to fully model this collaborative evolution relationship, ignoring the enhancement of group-level interests on item-level interests, and suffering from false-negative samples when aligning cross-level interests. In order to fully model the collaborative evolution relationship between dual-level user interests, we propose **CI4GI**, a Collaborative Interest-aware model for **Group Identification**. Specifically, we design an interest enhancement strategy that identifies additional interests of users from the items interacted with by the groups they have joined as a supplement to item-level interests. In addition, we adopt the distance between interest distributions of two users to optimize the identification of negative samples for a user, mitigating the interference of false-negative samples during cross-level interests alignment. The results of experiments on three real-world datasets demonstrate that CI4GI significantly outperforms state-of-the-art models.

Keywords: Recommender Systems · Group Recommendation · Graph Neural Networks · Contrastive Learning.

1 Introduction

With the proliferation of social media, joining online groups has become a vital way for users to share experiences, explore interests and expand social connections. For example, on the game platform *Steam*, players participate in multiplayer battles by joining game groups. Similarly, on the travel community *Mafengwo*, users can join groups of interest, find travel partners within the groups and plan group trips. For users, group participation serves not only as an effective channel for accessing vertical domain knowledge but also as a critical

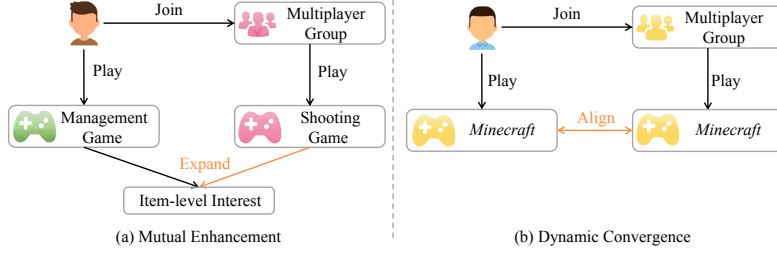


Fig. 1: Illustration of mutual enhancement and dynamic convergence.

hub for establishing social bonds. For the platform, users' interests in groups can enhance their engagement and retention. Therefore, the group identification (GI) task, i.e., recommending groups to users becomes a topic that needs to be explored. Compared with directly recommending items to users, recommending groups to users can establish an emotional link between the platform and users and maintain long-term user stickiness. This paper focuses on the GI task.

We note that the essence of the GI task is to understand users' interests and find groups that attract them. Specifically, users are jointly influenced by two levels of interests when joining groups: group-level interests and item-level interests, which can be learned from their historical participation in groups and historical interactions with items, respectively. Most importantly, these dual-level interests are deeply intertwined and exhibit a collaborative evolution mechanism: users initially select groups based on existing item-level interests, while their group interactions subsequently expand their item-level interests, because the items interacted with by the group might become potential item-level interests for users. This item-level interest evolution drives users to join more relevant groups, ultimately leading to a dynamic alignment between the dual-level interests.

The collaborative evolution process between the dual-level interests plays a crucial role in accurately modeling user interests for the GI task. It can be modeled from two aspects: i) **Mutual Enhancement**: On the one hand, joining groups expands users' item-level interests. For example, in Fig. 1(a), a user who typically plays single-player management games also joins a group to participate in multiplayer shooting games. In this case, the user's interest in shooting games within the group becomes a part of the item-level interest. On the other hand, item-level interests also influence users' decisions to join groups. Considering users' item-wise preferences in GI helps uncover groups that users are interested in. ii) **Dynamic Convergence**: There is an overlap between users' group-level and item-level interests. As shown in Fig. 1(b), a user plays *Minecraft* both in single-player mode and within a group in multiplayer mode. At this point, the user's item-level and group-level interests are highly aligned. This property justifies constructing cross-level preference alignment via self-supervised learning, thereby enhancing representation quality and recommendation accuracy.

Existing methods fail to effectively solve the GI task. Traditional recommendation models focus on binary recommendation tasks, such as user-item recommendation, which only model users’ item-level interests and are not suitable for group identification. Recently, several methods specifically designed for the GI task have been proposed, such as DiRec [15] and GTGS [17]. While these methods model both users’ group-level and item-level interests, they fail to fully capture the collaborative evolution relationship between these dual-level interests. Specifically, they model users’ group-level and item-level interests separately, overlooking the enhancement effect of group participation on users’ item-level interests. What is worse, although these methods employ contrastive learning to align cross-level interests within a user and push away this user’s interests from other users’ interests, they overlook interest overlaps among similar users, crudely pushing away the representations of similar users, which adversely affects the representation learning and recommendation effectiveness.

To fully model the collaborative evolution relationship between users’ dual-level interests, we propose a collaborative interest-aware graph learning model CI4GI. CI4GI leverages the hypergraph convolution network and graph attention network (GAT) to learn users’ group-level and item-level interests from their historical user-group and user-item interactions. To model the mutual enhancement between these dual-level interests, we design an interest enhancement strategy with two key innovations: i) Item Representation Enhancement: we augment the item representation in the user-item interaction graph based on group-item interaction information. ii) Contextual Enhancement: The group a user joins is considered as the context of that user. We identify additional interests from items interacted with by the groups the user has joined and use them as a supplement to item-level interests. Furthermore, to mitigate false-negative interference in contrastive learning caused by users with similar interests, we propose a dynamic false-negative sample optimized self-supervised learning loss to effectively align users’ group-level and item-level interests.

Our contributions are summarized as follows.

- We highlight that the key to the GI task lies in understanding users’ dual-level interests (i.e., group-level and item-level interests) and capturing their collaborative evolution relationship. To address this, we propose a collaborative interest-aware graph learning model CI4GI.
- We design an interest enhancement strategy that supplements users’ item-level interests through item representation enhancement and contextual enhancement.
- We propose a dynamic false-negative sample identification method based on the distance between interest distributions of two users to alleviate the problem of false-negative samples caused by overlapping interests of similar users.
- We conduct extensive experiments on three publicly available datasets, and the significant improvement of CI4GI on all datasets demonstrates its strength in completing the GI task.

The rest of the paper is organized as follows. Section 2 introduces the related work and Section 3 describes the model CI4GI in detail. Subsequently, Section 4 gives the experimental evaluation. Finally, the paper is concluded in Section 5.

2 Related Work

The group identification task is first proposed in CFAG [18], and there exist closely related tasks in the name of group recommendation. Therefore, we provide a brief introduction to these tasks and emphasize their differences.

The group recommendation aims to recommend items for a group of members [3,19,21]. Existing group recommendation models focus on aggregating the interests of different members to recognize their common interests. Multiple group recommendation models such as AGREE [1] and SoAGREE [2] propose different attention-based aggregation methods. Recently, ConsRec [16] models users and items as nodes, groups as hyperedges, and learns group representations through the hypergraph neural network. However, these methods mainly model the item-level interests of users, ignoring users’ group-level interests.

In addition to recommending items to groups, the term ‘group recommendation’ in the literature is also used to denote the recommendation of groups to their potential members. Traditional approaches typically use various algorithms to reconstruct user-group membership matrices by utilizing additional auxiliary information, such as the semantic content of group descriptions in CCF [4], visual information from photos in JTM [11], and user behaviors across different time periods in DMF [13]. CFAG [18] is the first to define the GI task, which learns the interaction relationships among users, groups, and items through tripartite graph convolution layers. DiRec [15] classifies the user’s intention of joining a group into social intention and personal interest intention, and then combines user and group representations under these two intention categories for recommendations. GTGS [17] models the user-group-item relationships using three hypergraphs: a group hypergraph from the user’s perspective, a user hypergraph from the item’s perspective, and a user hypergraph from the group’s perspective. However, all of these approaches model users’ group-level and item-level interests separately, not only ignoring the extension of group-level interests to item-level interests but also often improperly aligning two interests through contrastive learning.

On the other hand, our work is related to graph-based recommender systems (RSs), which can be broadly categorized into three classes from a modeling perspective: (i) graph convolutional network-based RSs [6,12,14]; (ii) graph attention network-based RSs [10,9]; and (iii) gated graph neural network-based RSs [5]. Most graph-based RSs focus only on user-item bipartite graphs, and directly applying these models to the group identification task is inappropriate due to the fact that they cannot capture the complex relationships among users, items, and groups. To realize the GI task, new graph-based models need to be developed.

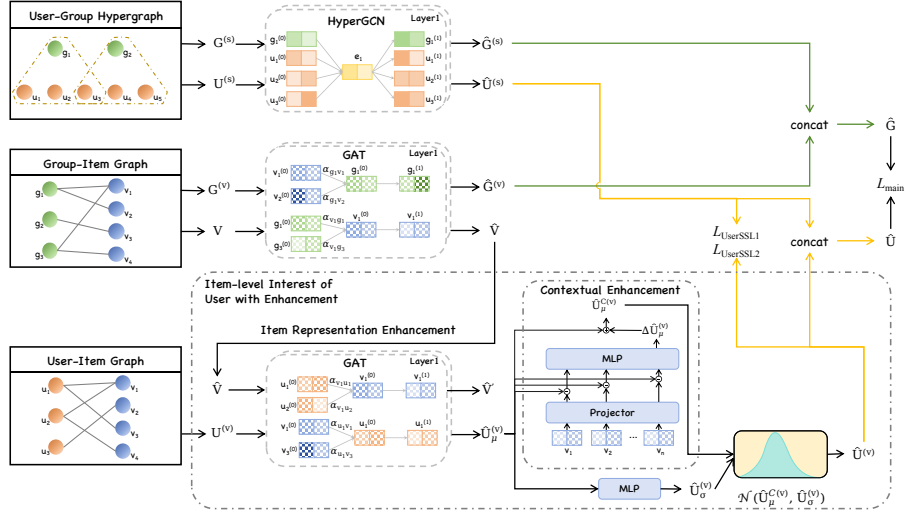


Fig. 2: Architecture of CI4GI.

3 Methodology

3.1 Model Overview

Let \mathcal{U} , \mathcal{V} and \mathcal{G} denote the user set, item set, and group set, respectively. There are three types of observed interactions among users, items, and groups, i.e., user-item interactions denoted as $\mathbf{X} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{V}|}$, group-item interactions represented as $\mathbf{Y} \in \mathbb{R}^{|\mathcal{G}| \times |\mathcal{V}|}$, and user-group affiliations represented as $\mathbf{Z} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{G}|}$. Given a user $u_i \in \mathcal{U}$, the goal of the group identification task is to predict the groups that user u_i has not yet joined but is highly likely to be interested in joining in the future.

For this task, we propose a collaborative interest-aware graph learning model CI4GI, whose architecture is shown in Fig. 2. CI4GI first employs a hypergraph to model user-group history interactions to learn group-level interests for users and groups. Then, CI4GI leverages two graph attention networks to model user-item and group-item interactions, capturing item-level interests for users and groups. To model the mutual enhancement between these dual-level interests, we design an interest enhancement strategy with two types of enhancements. First, item representation enhancement leverages group-item interaction information to augment item representations that have learned via the user-item interaction graphs. Second, contextual enhancement identifies additional interests of users from the items interacted with by the groups they have joined, serving as a supplement to users' item-level interests. Further, we propose a contrastive learning loss with dynamic false-negative sample optimization. It aligns users' group-level and item-level interests while alleviating the problem of false-negative samples caused by overlapping interests of similar users.

3.2 Embedding Layer

We maintain an embedding layer to initialize the learnable embeddings of users, items, and groups. For item set \mathcal{V} , the learnable item embedding matrix $\mathbf{V} \in \mathbb{R}^{|\mathcal{V}| \times d}$ is obtained after the embedding layer, where d is the embedding size. For the users, we represent group-level and item-level interests in two matrices, i.e., the group-level interest matrix $\mathbf{U}^{(s)} \in \mathbb{R}^{|\mathcal{U}| \times d}$ and the item-level interest matrix $\mathbf{U}^{(v)} \in \mathbb{R}^{|\mathcal{U}| \times d}$. Similarly, for groups, there are $\mathbf{G}^{(s)} \in \mathbb{R}^{|\mathcal{G}| \times d}$ and $\mathbf{G}^{(v)} \in \mathbb{R}^{|\mathcal{G}| \times d}$.

3.3 Group-level Interest Learning

We employ a hypergraph to model user-group affiliations. CI4GI defines a user-group affiliation hypergraph as $H^{UG} = (\mathcal{N}, \mathcal{E}^{UG})$, where $\mathcal{N} = \mathcal{U} \cup \mathcal{G}$ denotes the node set, i.e., a node of H^{UG} is a user in \mathcal{U} or a group in \mathcal{G} . The hyperedge set is denoted as \mathcal{E}^{UG} , where a hyperedge $e_k^{UG} \in \mathcal{E}^{UG}$, $k \in [1, |\mathcal{G}|]$ connects all user nodes in the group g_k along with the group node g_k . The association matrix $T^{UG} = [t_{ik}^{UG}] \in \mathbb{R}^{(|\mathcal{U}|+|\mathcal{G}|) \times |\mathcal{G}|}$ represents the connectivity between nodes and hyperedges in the hypergraph H^{UG} .

By modeling the hypergraph with a hyperedge connecting the group and all its members, we preserve the social relationships within the group. We then employ the classical hypergraph convolution for representation learning:

$$N^{(l)} = \sigma(D_{\mathcal{N}}^{-\frac{1}{2}} T^{UG} D_{\mathcal{E}}^{-1} T^{UGT} D_{\mathcal{N}}^{-\frac{1}{2}} N^{(l-1)} W^{UG}) \quad (1)$$

where $D_{\mathcal{N}} \in \mathbb{R}^{|\mathcal{N}| \times |\mathcal{N}|}$ and $D_{\mathcal{E}}^{|\mathcal{E}| \times |\mathcal{E}|}$ represent the degree matrices of the nodes and hyperedges, respectively. $N^{(l)} \in \mathbb{R}^{|\mathcal{N}| \times d}$ is the node representation of the l -th layer, $N^{(0)} = \text{Concat}(\mathbf{U}^{(s)}, \mathbf{G}^{(s)})$, $W^{UG} \in \mathbb{R}^{d \times d}$ is a learnable matrix, σ is the sigmoid function.

Finally, the group-level interests of users and groups are obtained after L layers hypergraph convolution: $\hat{\mathbf{U}}^{(s)}, \hat{\mathbf{G}}^{(s)} = N^{(L)}$.

3.4 Item-level Interest Learning

We learn the item-level interests of groups and users, respectively.

Item-level Interest of Group. We define the group-item interaction graph as $G^{GV} = (\mathcal{N}^{GV}, \mathcal{E}^{GV})$, where $\mathcal{N}^{GV} = \mathcal{G} \cup \mathcal{V}$ denotes the node set, an edge $e_{jk}^{GV} \in \mathcal{E}^{GV}$, $j \in [1, |\mathcal{V}|]$, $k \in [1, |\mathcal{G}|]$ denotes the group g_k interacting the items v_j . The association matrix $T^{GV} \in \mathbb{R}^{(|\mathcal{G}|+|\mathcal{V}|) \times (|\mathcal{G}|+|\mathcal{V}|)}$ represents the connectivity between nodes in G^{GV} . We use classical GAT [10] for representation learning:

$$\hat{\mathbf{G}}^{(v)}, \hat{\mathbf{V}} = \text{GAT}_{\Theta_{GV}}([\mathbf{G}^{(v)} || \mathbf{V}], T^{GV}) \quad (2)$$

where $\text{GAT}_{\Theta_{GV}}$ is a stack of L -layer graph attention networks, $\mathbf{G}^{(v)}$ and \mathbf{V} are the initial groups' item-level interest representations and item representations, $\hat{\mathbf{G}}^{(v)} \in \mathbb{R}^{|\mathcal{G}| \times d}$ and $\hat{\mathbf{V}} \in \mathbb{R}^{|\mathcal{V}| \times d}$ are those learned from GAT.

Item-level Interest of User with Enhancement. Similarly, CI4GI defines the user-item interaction graph as $G^{UV} = (\mathcal{N}^{UV}, \mathcal{E}^{UV})$. The association matrix

$T^{UV} \in \mathbb{R}(|\mathcal{U}|+|\mathcal{V}|) \times (|\mathcal{U}|+|\mathcal{V}|)$ denotes the connectivity between nodes in G^{UV} . We also apply a graph attention network to learn user-item interactions.

Further, we propose an interest enhancement strategy with two methods: item representation enhancement and contextual enhancement.

Item representation enhancement aims to enhance the item representation in user-item interaction graph with the group-item interaction information, thereby capturing the similarity between the item-level interest of the group and the user. This helps the model recognize the user's tendency to join a group based on their interests in the items. Specifically, we use the item representation $\hat{\mathbf{V}}$ learned from G^{GV} as the initial item representation for G^{UV} :

$$\hat{\mathbf{U}}_\mu^{(v)}, \hat{\mathbf{V}}' = \text{GAT}_{\theta_{UV}}([\mathbf{U}^{(v)} || \hat{\mathbf{V}}], T^{UV}) \quad (3)$$

where $\text{GAT}_{\theta_{UV}}$ is the graph attention network, $\mathbf{U}^{(v)}$ is the initial users' item-level interest representations and $\hat{\mathbf{V}}$ is the item representation learned from G^{GV} . $\hat{\mathbf{U}}_\mu^{(v)}$ and $\hat{\mathbf{V}}'$ are the user and item representations learned by the GAT.

In group identification scenarios, users' item-level interests are extended by joining groups, with items shared within these groups becoming potential interests for the user. Therefore, we propose a **contextual enhancement** that treats the groups the user has joined as their context information, identifying potential interests from the items that the user has historically interacted with by joining the group. Specifically, we first identify contextual items by the user-group affiliation matrix \mathbf{Z} and the group-item interaction matrix \mathbf{Y} . Then, we project the contextual item representation $\hat{\mathbf{V}}$ into the same vector space as the user's representation $\hat{\mathbf{U}}_\mu^{(v)}$ using a projection matrix \mathbf{W}^C . Finally, we compute the distance between the user representation and the contextual items embeddings to obtain the increment $\Delta\hat{\mathbf{U}}_\mu^{(v)}$ with the following formula:

$$\Delta\hat{\mathbf{U}}_\mu^{(v)} = \text{MLP}(\hat{\mathbf{U}}_\mu^{(v)} - D_{ZY}^{-1} \mathbf{Z} \mathbf{Y} \times \hat{\mathbf{V}} \mathbf{W}^C) \quad (4)$$

where $D_{ZY} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{U}|}$ is the row degree matrix of the matrix obtained after multiplying matrices \mathbf{Z} and \mathbf{Y} , which is used for normalization. $\mathbf{W}^C \in \mathbb{R}^{d \times d}$ is the projection matrix.

The context-enhanced user representation is then obtained by adding the increment $\Delta\hat{\mathbf{U}}_\mu^{(v)}$:

$$\hat{\mathbf{U}}_\mu^{C(v)} = \hat{\mathbf{U}}_\mu^{(v)} + \gamma \Delta\hat{\mathbf{U}}_\mu^{(v)} \quad (5)$$

where γ is a hyperparameter indicating the weight of the increment.

Further, we represent the users' item-level interests as independent Gaussian distributions instead of fixed embeddings, which improves the robustness and flexibility of users' interest representations. Specifically, based on the user representation $\hat{\mathbf{U}}_\mu^{(v)}$ learned from Eq. 3 (which is regarded as the mean of the user interest distribution), the variance of the user interest distribution is computed by a multilayer perceptron: $\hat{\mathbf{U}}_\sigma^{(v)} = \text{MLP}(\hat{\mathbf{U}}_\mu^{(v)})$.

We sample user representations $\hat{\mathbf{U}}^{(v)}$ from independent Gaussian distributions $\hat{\mathbf{U}}^{(v)} \sim \mathcal{N}(\hat{\mathbf{U}}_\mu^{(v)}, \hat{\mathbf{U}}_\sigma^{(v)})$. Since direct sampling prevents gradient backpropagation, we apply the reparameterization trick as follows:

$$\hat{\mathbf{U}}^{(v)} = \hat{\mathbf{U}}_\mu^{C(v)} + \hat{\mathbf{U}}_\sigma^{(v)} \epsilon \quad (6)$$

where ϵ is randomly sampled from the standard normal distribution $\mathcal{N}(0, 1)$.

3.5 Cross-level Interest Alignment

In this paper, we model users' group-level and item-level interests separately. Since these interests may overlap in real group identification scenarios, we propose using contrastive learning to align the cross-level interests. Additionally, similar users' interests can overlap, and vanilla contrastive learning may lead to false-negative samples. To address this, we introduce a dynamic false-negative sample optimized contrastive learning loss, using Wasserstein distance between user interest distributions to identify false-negative samples and improve the quality of cross-level interest alignment.

Vanilla Contrastive Learning. We align the cross-level interests of the same user by performing conventional contrastive learning on the user's group-level interest $\hat{\mathbf{U}}^{(s)}$ and item-level interest $\hat{\mathbf{U}}^{(v)}$ via the following InfoNCE [7] loss:

$$L_{\text{UserSSL1}} = - \sum_{u_i \in \mathcal{U}} \log \frac{\exp(\text{sim}(\hat{\mathbf{u}}_i^{(v)}, \hat{\mathbf{u}}_i^{(s)})/\tau)}{\exp(\text{sim}(\hat{\mathbf{u}}_i^{(v)}, \hat{\mathbf{u}}_i^{(s)})/\tau) + N_{V1}^U + N_{S1}^U} \quad (7)$$

$$N_{V1}^U = \sum_{u_{i'} \in \mathcal{U}_i^-} \exp(\text{sim}(\hat{\mathbf{u}}_{i'}^{(v)}, \hat{\mathbf{u}}_i^{(s)})/\tau), \quad N_{S1}^U = \sum_{u_{i'} \in \mathcal{U}_i^-} \exp(\text{sim}(\hat{\mathbf{u}}_i^{(v)}, \hat{\mathbf{u}}_{i'}^{(s)})/\tau) \quad (8)$$

where $\hat{\mathbf{u}}_i^{(s)}$ and $\hat{\mathbf{u}}_i^{(v)}$ form a pair of positive samples. \mathcal{U}_i^- is the set of negative samples w.r.t. the user u_i , which is composed of other users (i.e., $u_{i'} \neq u_i$) within the same batch. $\text{sim}(\cdot)$ is to calculate the similarity of a pair of vectors, which refers to the cosine similarity in this paper. τ is the temperature parameter.

Dynamic False-negative Sample Optimization. We dynamically identify false-negative samples by comparing users' item-level interest distributions. Specifically, for the item-level interest distribution $\hat{\mathbf{u}}_i^{(v)}$ of user u_i , we compute the Wasserstein distance to the distribution $\hat{\mathbf{u}}_j^{(v)}$ of any other user u_j . If the distance is less than or equal to a threshold μ , u_j is considered a false-negative sample of u_i . Only users with a Wasserstein distance greater than μ from u_i are included in the negative sample set.

$$\mathcal{Neg}_i = \{u_j | u_j \in \mathcal{U} \text{ and } d_{W2}(u_i, u_j) > \mu\} \quad (9)$$

where d_{W2} denotes the Wasserstein distance, the higher the similarity of two users' distributions, the smaller the Wasserstein distance, μ is the threshold, and \mathcal{Neg}_i is the set of negative samples w.r.t. the user u_i .

To ensure that the false-negative sample set remains adaptive, we dynamically update the set in each batch based on the latest interest distributions. This allows the contrastive learning process to continuously adjust to evolving user interests and prevents the model from being misled by static or outdated negative samples. Then, similar to conventional user contrastive learning, the following loss is used to align cross-level interest representations of the same user:

$$L_{\text{UserSSL2}} = - \sum_{u_i \in \mathcal{U}} \log \frac{\exp(\text{sim}(\hat{\mathbf{u}}_i^{(v)}, \hat{\mathbf{u}}_i^{(s)})/\tau)}{\exp(\text{sim}(\hat{\mathbf{u}}_i^{(v)}, \hat{\mathbf{u}}_i^{(s)})/\tau) + N_{V2}^U + N_{S2}^U} \quad (10)$$

$$N_{V2}^U = \sum_{u_{i'} \in \mathcal{U}_i^- \cap \text{Neg}_i} \exp(\text{sim}(\hat{\mathbf{u}}_{i'}^{(v)}, \hat{\mathbf{u}}_i^{(s)})/\tau), \quad N_{S2}^U = \sum_{u_{i'} \in \mathcal{U}_i^- \cap \text{Neg}_i} \exp(\text{sim}(\hat{\mathbf{u}}_i^{(v)}, \hat{\mathbf{u}}_{i'}^{(s)})/\tau) \quad (11)$$

3.6 Model Optimization

After obtaining the group-level and item-level interest representations of users and groups, we can calculate the main loss of CI4GI by three steps. First, the two types of interests are concatenated for both users and groups to obtain the final user representation $\hat{\mathbf{U}}$ and group representation $\hat{\mathbf{G}}$:

$$\hat{\mathbf{U}} = [\hat{\mathbf{U}}^{(s)} || \hat{\mathbf{U}}^{(v)}], \quad \hat{\mathbf{G}} = [\hat{\mathbf{G}}^{(s)} || \hat{\mathbf{G}}^{(v)}] \quad (12)$$

Then, the dot product similarity is used to compute the probability score s_{ik} of a user u_i joining the group g_k :

$$s_{ik} = \hat{\mathbf{u}}_i \cdot \hat{\mathbf{g}}_k \quad (13)$$

where $\hat{\mathbf{u}}_i = \hat{\mathbf{U}}(i, :)$ is the final representation of the user u_i and $\hat{\mathbf{g}}_k = \hat{\mathbf{G}}(k, :)$ is the final representation of the group g_k . Subsequently, we calculate the BPR loss as the main loss:

$$L_{\text{main}} = \frac{1}{|\mathcal{T}|} \sum_{(u_i, g_k, g_{k'}) \in \mathcal{T}} -\log \sigma(s_{ik} - s_{ik'}) \quad (14)$$

where $\mathcal{T} = \{(u_i, g_k, g_{k'}) \mid z_{ik} = 1 \text{ and } z_{ik'} = 0\}$ is the training set, $z \in \mathbf{Z}$ is the element in the user-group affiliation matrix. σ is the sigmoid function.

Finally, we jointly optimize the main loss and auxiliary losses:

$$L = L_{\text{main}} + \lambda_1(\beta L_{\text{UserSSL1}} + (1 - \beta)L_{\text{UserSSL2}}) + \lambda_2 \|\Theta\|_2^2 \quad (15)$$

where Θ is all trainable parameters in CI4GI, $\|\Theta\|_2^2$ is the regularization loss, λ_1 and λ_2 are hyperparameters, and $\beta = 1/(1 + \exp(-k(\text{epoch} - E)))$ is the annealing parameter that controls the weight of the two types of contrastive learning loss, where k and E are hyperparameters.

Table 1: Statistics of datasets.

Dataset	# Users	# Items	# Groups	# User-Group participation	# User-Item interactions	# Group-Item interactions
Mafengwo	1,269	999	972	5,574	8,676	2,540
Weeplaces	1,501	6,406	4,651	12,258	43,942	6,033
Douban	11,099	2,351	1,085	57,654	444,776	23,318

3.7 Complexity Analysis

Space Complexity. In CI4GI, the learnable parameters mainly come from item embeddings, two user interest embeddings and two group interest embeddings. In addition, in the hypergraph convolution, the number of parameters of the L layer is Ld^2 , and the number of parameters of the two L layers GAT is $2L(d^2 + 2d)$. The number of parameters in the MLP and the projection matrix is $O(d^2 + d)$. Therefore, the space complexity of CI4GI is $O(|\mathcal{V}|d + |\mathcal{U}|d + |\mathcal{G}|d + Ld^2 + Ld)$.

Time Complexity. The computational amount during the training of CI4GI is mainly concentrated on the hypergraph convolution and GAT. Assuming that $|T^{UG}|$ denotes the number of non-zero elements of the association matrix T^{UG} , the time complexity of hypergraph convolution is $O(L(|\mathcal{U}| + |\mathcal{G}|)d^2 + L|T^{UG}|d)$. Assuming that $|T^{GV}|$ and $|T^{UV}|$ denote the number of non-zero elements in the association matrices T^{GV} and T^{UV} , respectively, the time complexity of the two GATs is $O(L(|\mathcal{U}| + |\mathcal{G}| + |\mathcal{V}|)d^2 + L(|T^{GV}| + |T^{UV}|)d)$. Therefore, the time complexity of CI4GI is $O(Ld^2(|\mathcal{U}| + |\mathcal{V}| + |\mathcal{G}|) + Ld(|T^{UG}| + |T^{GV}| + |T^{UV}|))$.

4 Experiments

4.1 Experimental Setup

Datasets. We choose three public datasets to conduct experiments.

- **Mafengwo.** It records the travel history of users on the Mafengwo APP, where users can create or join groups, taking offline group trips. We use the dataset published by CFAG [18].
- **Weeplaces.** It records users' check-ins on location-based social networks in major cities of the U.S. We follow the same operations as in GroupIM [8] for constructing user-POI interactions and group-POI interactions.
- **Steam.** It records users' game preferences on the online gaming platform Steam, where users have their records of games they have played and can create or join a group. We use the dataset published by CFAG [18].

Table 1 lists the statistics of the three datasets. We randomly split the user participation in each dataset into training, validation, and test sets with a ratio of 7:1:2.

Baselines. The following baselines are chosen to compare with CI4GI:

Three recommendation models: **LightGCN**⁴, which is a classical graph-based recommendation model [6]; **SGL**⁵, which introduces contrastive learning into GNN-based recommendation by generating contrast views through node dropout, edge dropout, or random walk [14]; **SimGCL**⁶, which presents an embedding-based enhancement method to construct positive sample pairs in contrastive learning by adding uniform noise to the embedding [20]. As DiRec[15] does, we apply them on the GI task by treating each group as an item and thus only utilizing user-group affiliations.

Two group recommendation models: **AGREE**⁷, which is a classical group recommendation model using an attention mechanism for member aggregation [1]; **ConsRec**⁸, the state-of-the-art model for group recommendation, which proposes a hypergraph neural network to learn member-level aggregation and captures the group consensus on three views [16]. As DiRec[15] does, we adapt them by replacing their initial group-item BPR loss with user-group BPR loss.

Three GI models: **CFAG**⁹, a classical GI model, which constructs a group-user-item tripartite graph and designs a tripartite graph convolution layer [18]; **DiRec**¹⁰, which divides the user’s intention of joining a group into social intention and personal interest intention [15]; **GTGS**¹¹, which models the relationships between users, groups, and items by three hypergraphs, and proposes transition hypergraph convolution by using users’ preferences for items as a prior knowledge [17].

Implementation Details. We implement our model in PyTorch. In our model, L is set to 2, the temperature τ is set to 1, the hyperparameter k in β is set to 0.1, and the threshold E in β is set as follows: 20 for Mafengwo, 30 for Weeplaces and Steam, and contextual enhancement representation weight γ is set as follows: 1 for Mafengwo and Steam, and 1.25 for Weeplaces, and the Wasserstein distance threshold μ is set as follows: 1.5 for Mafengwo, 2.0 for Weeplaces and Steam, λ_1 and λ_2 are set to $1 \times e^{-4}$. For the sake of fairness, we set the size of all embeddings d to 256, the batch size to 1024 and the learning rate to 0.005 in all the experiments. For all baselines, the hyperparameters are set to values corresponding to best performance reported in their respective papers. Experiments are conducted on NVIDIA RTX3090 GPU with 24G memory. The implementation code has been released¹².

Metrics. To evaluate the performance of recommending groups to users, we adopt two metrics, i.e., Recall@K and NDCG@K (R@K and N@K for short), where Recall focuses on whether the user actually chooses the recommended

⁴ <https://github.com/kuandeng/LightGCN>

⁵ <https://github.com/wujcan/SGL-Torch>

⁶ <https://github.com/Coder-Yu/QRec>

⁷ <https://github.com/LianHaiMiao/Attentive-Group-Recommendation>

⁸ <https://github.com/FDUDSDE/WWW2023ConsRec>

⁹ <https://github.com/mdyfrank/CFAG>

¹⁰ <https://github.com/WxxShirley/CIKM2023DiRec>

¹¹ <https://github.com/mdyfrank/GTGS>

¹² <https://github.com/ZhaoRui-7/CI4GI>

Table 2: Overall performance. The values in bold and underlined are the best and second best results in each column.

Dataset	Mafengwo				Weeplaces				Steam			
Metric	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20
LightGCN	0.2925	0.3607	0.1865	0.2040	0.2490	0.3159	0.1465	0.1646	0.2411	0.3283	0.1333	0.1558
SGL	0.2957	0.3628	0.1937	0.2109	0.2511	0.3124	0.1456	0.1624	0.2327	0.3241	0.1289	0.1524
SimGCL	0.2943	0.3576	0.1890	0.2052	0.2486	0.3140	0.1466	0.1645	0.1872	0.2896	0.0930	0.1189
AGREE	0.1679	0.2302	0.1061	0.1222	0.2312	0.2957	0.1355	0.1529	0.1882	0.2977	0.0996	0.1275
ConsRec	0.3403	0.4312	0.2161	0.2382	0.3451	0.4379	0.2039	0.2288	0.2568	0.3608	0.1359	0.1626
CFAG	0.3007	0.4051	0.1698	0.1965	0.3848	0.4778	0.2251	0.2529	0.2328	0.3427	0.1224	0.1507
DiRec	0.3588	0.4636	0.2231	0.2500	<u>0.3904</u>	<u>0.4843</u>	<u>0.2341</u>	<u>0.2591</u>	<u>0.2738</u>	<u>0.3745</u>	<u>0.1436</u>	<u>0.1696</u>
GTGS	<u>0.3611</u>	<u>0.4672</u>	<u>0.2248</u>	<u>0.2520</u>	0.3813	0.4693	0.2308	0.2547	0.2225	0.3275	0.1110	0.1379
CI4GI	0.3937	0.5016	0.2557	0.2829	0.4179	0.4990	0.2664	0.2879	0.2912	0.3823	0.1605	0.1840
Inprov.(%)	9.02	7.36	13.74	12.26	7.04	3.03	13.79	11.11	6.35	2.08	11.76	8.49

group, NDCG focuses on the ranking of the recommended groups and K is set to either 10 or 20.

4.2 Performance Comparison

Overall Performance. Table 2 lists the experimental results on the three datasets. From Table 2, we have the following observations.

Traditional recommendation models perform poorly because they typically represent user-group affiliations as graphs, capturing only users’ group-level interests while overlooking their item-level interests. Similarly, group recommendation models are not well-suited for the GI task, as they lack the capability to model the complex user interests in GI scenarios.

In contrast, GI models achieve better performance as they are specifically designed for this task. However, while they capture both users’ group-level and item-level interests, they overlook the collaborative relationship between the dual-level interests, limiting their effectiveness.

Our CI4GI outperforms all baselines on three datasets. Taking NDCG@10 as an example, compared to the best baseline on each of the three datasets, CI4GI shows improvements of 11.76% - 13.79%, averaging at 13.09%.

Cold-start Performance. We evaluate the cold-start performance of CI4GI on Mafengwo and Weeplaces. Specifically, we randomly remove user-group interaction history in the training set, ensuring that each user has joined at most k groups, where $k = 1, 2, 3, 4$. We choose DiRec, GTGS, ConsRec and SGL, the best models from different types of baselines, for comparison.

As shown in Fig. 3, CI4GI outperforms the baselines in all cases. In particular, CI4GI achieves the best performance even when $k = 1$, i.e., the user has only one interacted group in history. This could be because CI4GI applies an interest enhancement strategy in the user’s item-level interest learning, which improves the representation of the user’s item-level interest and alleviates the deficiency of the user’s group-level interests in cold-start scenarios. As the threshold k increases

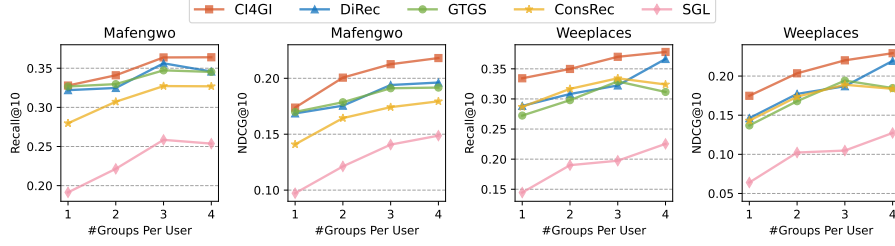


Fig. 3: Cold-start performance comparison.

Table 3: Ablation study.

Model	Mafengwo		Weeplaces		Steam	
	R@10	N@10	R@10	N@10	R@10	N@10
CI4GI	0.3973	0.2557	0.4179	0.2664	0.2912	0.1605
(A)w/o group-level interests	0.3351	0.2135	0.3828	0.2349	0.2699	0.1443
(B)w/o item-level interests	0.3203	0.2074	0.3097	0.1954	0.2836	0.1529
(C)w/o enhancement	0.3687	0.2365	0.3873	0.2436	0.2820	0.1592
(D)w/o item enhancement	0.3817	0.2547	0.4069	0.2508	0.2866	0.1602
(E)w/o contextual enhancement	0.3697	0.2426	0.3894	0.2460	0.2838	0.1541
(F)w/o CL	0.3730	0.2484	0.3994	0.2549	0.2765	0.1547
(G)w/o $L_{UserSSL1}$	0.3818	0.2503	0.4134	0.2649	0.2837	0.1564
(H)w/o $L_{UserSSL2}$	0.3772	0.2514	0.4115	0.2626	0.2845	0.1565
(I)w/o β	0.3797	0.2525	0.4109	0.2597	0.2826	0.1527

from 1 to 3, the performance of all models improves, indicating that more historical user-group interactions are beneficial to interest learning. However, when k increases from 3 to 4, some models experience a decline in performance (e.g., DiRec on Mafengwo, GTGS and ConsRec on Weeplaces), while CI4GI continues to improve its performance. This may be because a higher number of user-group interactions introduce more complex user’s interests, and CI4GI can capture these intricate interests by identifying the collaborative relationships between dual-level user interests, thus leading to superior performance.

4.3 Ablation Study

Effect of Dual-level Interest Learning. We design two variants to observe the impact of dual-level interest learning in CI4GI on performance. Variant A removes group-level interests in CI4GI, i.e., only item-level interests of users and groups are used to compute similarity. Variant B removes item-level interests in CI4GI.

The experimental results on three datasets are listed in the top part of Table 3. Removing either interest leads to significant performance degradation,

which illustrates the effectiveness of dual-level interest modeling in CI4GI. On Steam, the performance degradation of variant A is more significant, indicating that group-level interests play a more important role. In contrast, on Mafengwo and Weeplaces, the performance degradation of variant B is more significant, i.e., item-level interests play a more important role. This may be due to the fact that Weeplaces has a very low average number of users per group (fewer than 3), making it difficult to accurately characterize groups using only group-level interests, leading to a substantial performance drop.

Effect of Interest Enhancement Strategy. We build three variants to observe the effect of interest enhancement strategy during users’ item-level interest learning in CI4GI. Variant C deletes the interest enhancement strategy, i.e., directly uses the user representation obtained through the GAT as the item-level interest representation of the user. Variant D removes the item representation enhancement, replacing the item representation in user-item GAT learning with random initialization. Variant E deletes the contextual enhancement.

The experimental results of these three variants on three datasets are listed in the middle part of Table 3. The performance of variants D and E is lower than that of CI4GI, indicating that removing any of the interest enhancement methods results in a decrease in performance. Among the two variants, variant E shows the most significant performance degradation, highlighting the effectiveness of contextual enhancement in improving group identification. This is likely because it identifies the item that motivates the user to join the group as a potential interest, effectively enriching the user’s item-level preferences. Additionally, variant C exhibits a significant performance degradation, which suggests that the simultaneous deletion of the two interest enhancement methods has a significant negative impact on model performance.

Effect of Contrastive Learning. We build four variants to observe the effect of contrastive learning in CI4GI. Variant F deletes the entire contrastive learning (CL) module. Variant G removes the vanilla contrastive learning loss L_{UserSSL1} , variant H removes the contrastive learning loss with dynamic false-negative sample optimization L_{UserSSL2} , and variant I deletes the annealing parameter β used to balance the two contrastive learning losses, setting β to a constant value of 0.5.

The experimental results on three datasets are listed in the bottom part of Table 3. Both variants G and H underperform compared to CI4GI, indicating that removing either contrastive learning loss negatively impacts model performance. Variant I experiences a more significant performance degradation compared to variants G and H, as the absence of the annealing parameter prevents the model from properly balancing the two contrastive learning losses. This causes the model to rely on the Wasserstein distance for identifying false-negative samples before it has sufficiently learned user interest patterns and distributional representations. As a result, the identified false-negative samples are heavily influenced by random initialization and lack meaningful guidance. Additionally, variant F shows a substantial performance decline, highlighting the crucial role of the contrastive learning module in CI4GI.

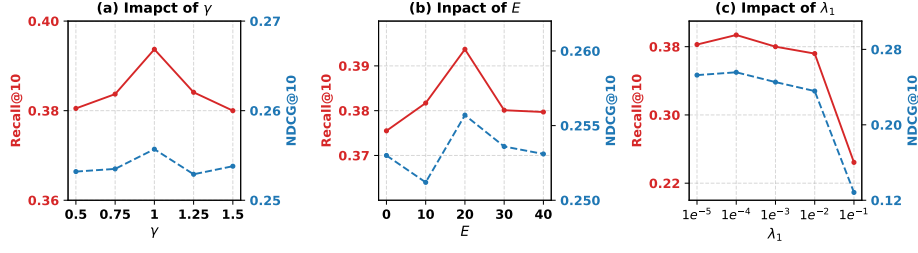


Fig. 4: Sensitivity analysis of hyperparameters γ , E and λ_1 on Mafengwo dataset.

4.4 Hyperparameter Sensitivity Analysis

We perform experiments on Mafengwo datasets to explore the sensitivity of the weight of contextual enhancement γ , the threshold E in annealing parameters β and the weight of contrastive learning loss λ_1 . We fix other hyper-parameters, and tune γ , E and λ_1 within $\{0.5, 0.75, 1, 1.25, 1.5\}$, $\{0, 10, 20, 30, 40\}$, $\{1e^{-5}, 1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}\}$, respectively. From the results in Fig. 4, we can observe that as γ increases from 0.5 to 1, the performance of CI4GI shows an increasing trend, and when γ exceeds 1, the model performance starts to decrease. This suggests that appropriate contextual enhancement representation weight helps to improve the quality of the user representation, which in turn improves the model performance. The performance of CI4GI first rises and then falls as E increases. When E is too small, the model is prematurely subjected to unguided false negative sample identification before it has learned the basic user interests. Conversely, when E is too large, the weight of contrastive learning with negative sample optimization is consistently small and fail to guide model optimization. CI4GI achieves optimal performance when $\lambda_1 = 1 \times e^{-4}$. If λ_1 is too small, the contrastive learning loss has little impact, while if λ_1 is too large, the weight of the contrastive learning loss is almost equal to the weight of the main loss, leading to a decrease in effectiveness.

5 Conclusion

Group identification is a challenging recommendation task, as a user’s decision on joining a group is jointly influenced by both group-level and item-level interests. Therefore, effectively capturing these two types of interests and their collaborative relationship is crucial for accurate group identification. We propose a model CI4GI, which simultaneously models users’ group-level and item-level interests, and designs an interest enhancement strategy to capture the mutual enhancement of the dual-level interests through item representation enhancement and contextual enhancement. Meanwhile, we design a contrastive learning strategy with dynamic false-negative sample optimization to improve the alignment of cross-level interests. Experimental results on public datasets show that CI4GI significantly improves the accuracy of group identification.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grant No. 62072450.

References

1. Cao, D., He, X., Miao, L., An, Y., Yang, C., Hong, R.: Attentive group recommendation. In: The 41st International ACM SIGIR conference on research & development in information retrieval. pp. 645–654 (2018)
2. Cao, D., He, X., Miao, L., Xiao, G., Chen, H., Xu, J.: Social-enhanced attentive group recommendation. *IEEE Transactions on Knowledge and Data Engineering* **33**(3), 1195–1209 (2019)
3. Chen, T., Yin, H., Long, J., Nguyen, Q.V.H., Wang, Y., Wang, M.: Thinking inside the box: learning hypercube representations for group recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1664–1673 (2022)
4. Chen, W.Y., Zhang, D., Chang, E.Y.: Combinational collaborative filtering for personalized community recommendation. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 115–123 (2008)
5. Cui, Z., Li, Z., Wu, S., Zhang, X.Y., Wang, L.: Dressing as a whole: Outfit compatibility learning based on node-wise graph neural networks. In: The world wide web conference. pp. 307–317 (2019)
6. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: 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 (2020)
7. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018)
8. Sankar, A., Wu, Y., Wu, Y., Zhang, W., Yang, H., Sundaram, H.: Groupim: A mutual information maximization framework for neural group recommendation. In: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. pp. 1279–1288 (2020)
9. Tao, Z., Wei, Y., Wang, X., He, X., Huang, X., Chua, T.S.: Mgat: Multimodal graph attention network for recommendation. *Information Processing & Management* **57**(5), 102277 (2020)
10. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017)
11. Wang, J., Zhao, Z., Zhou, J., Wang, H., Cui, B., Qi, G.: Recommending flickr groups with social topic model. *Information retrieval* **15**, 278–295 (2012)
12. Wang, X., He, X., Wang, M., Feng, F., Chua, T.S.: Neural graph collaborative filtering. In: Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. pp. 165–174 (2019)
13. Wang, X., Donaldson, R., Nell, C., Gorniak, P., Ester, M., Bu, J.: Recommending groups to users using user-group engagement and time-dependent matrix factorization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 30 (2016)

14. Wu, J., Wang, X., Feng, F., He, X., Chen, L., Lian, J., Xie, X.: 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 (2021)
15. Wu, X., Xiong, Y., Zhang, Y., Jiao, Y., Zhang, J.: Dual intents graph modeling for user-centric group discovery. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. pp. 2716–2725 (2023)
16. Wu, X., Xiong, Y., Zhang, Y., Jiao, Y., Zhang, J., Zhu, Y., Yu, P.S.: Consrec: Learning consensus behind interactions for group recommendation. In: Proceedings of the ACM Web Conference 2023. pp. 240–250 (2023)
17. Yang, M., Liu, Z., Yang, L., Liu, X., Wang, C., Peng, H., Yu, P.S.: Group identification via transitional hypergraph convolution with cross-view self-supervised learning. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. pp. 2969–2979 (2023)
18. Yang, M., Liu, Z., Yang, L., Liu, X., Wang, C., Peng, H., Yu, P.S.: Ranking-based group identification via factorized attention on social tripartite graph. In: Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining. pp. 769–777 (2023)
19. Yin, H., Wang, Q., Zheng, K., Li, Z., Yang, J., Zhou, X.: Social influence-based group representation learning for group recommendation. In: 2019 IEEE 35th International Conference on Data Engineering (ICDE). pp. 566–577. IEEE (2019)
20. Yu, J., Yin, H., Xia, X., Chen, T., Cui, L., Nguyen, Q.V.H.: Are graph augmentations necessary? simple graph contrastive learning for recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1294–1303 (2022)
21. Zhao, R., Jin, B., Lv, Y., Zheng, Y., Lai, W.: Multiple hypergraph learning for ephemeral group recommendation. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 89–105. Springer (2024)