Dual-Graph Convolutional Network and Dual-View Fusion for Group Recommendation

Chenyang Zhou[1], Guobing Zou[1](✉), Shengxiang Hu[1], Hehe Lv[1], Liangrui Wu[1], and Bofeng Zhang[2]

[1] School of Computer Engineering and Science, Shanghai University, Shanghai, China
{chenyangzhou,gbzou,shengxianghu,hhlv,lr_w}@shu.edu.cn
[2] School of Computer and Information Engineering, Shanghai Polytechnic University, Shanghai, China
bfzhang@sspu.edu.cn

Abstract. Group recommendation constitutes a burgeoning research focus in recommendation systems. Despite a multitude of approaches achieving satisfactory outcomes, they still fail to address two major challenges: 1) these methods confine themselves to capturing user preferences exclusively within groups, neglecting to consider user collaborative signals beyond groups, which reveal users' potential interests; 2) they do not sufficiently take into account the impact of multiple factors on group decision-making, such as individual expertise and influence, and the group's general preferences. To tackle these challenges, we propose a new model named DDGR (**D**ual-Graph Convolutional Network and **D**ual-View Fusion for **G**roup **R**ecommendation), designed to capture representations addressing two aspects: member preferences and group preferences. DDGR consists of two components: 1) a dual-graph convolutional network that combines the benefits of both hypergraphs and graphs to fully explore member potential interests and collaborative signals; 2) a dual-view fusion strategy that accurately simulates the group negotiation process to model the impact of multiple factors from member and group view, which can obtain semantically rich group representations. Thorough validation on two real-world datasets indicates that our model significantly surpasses state-of-the-art methods.

Keywords: Group Recommendation · Hypergraph Learning · Graph Convolution Networks · Attention Mechanism

1 Introduction

The popularity of social media has led to a rise in online group activities. Recommending the related item to a group is a critical task in the information system. Unlike user recommendation, group recommendation involves group decision-making which is a complex process. Each group member has their preferences, which will affect the final decision. The more complicated aspect is that the

D.-N. Yang et al. (Eds.): PAKDD 2024, LNAI 14649, pp. 231–243, 2024.
https://doi.org/10.1007/978-981-97-2262-4_19

influence of group members also changes dynamically when faced with different choices. In the group decision-making process, it is necessary to minimize conflicts among members and improve the common acceptance of members.

In group preference capture, the early methods mainly adopted predefined and fixed aggregation strategies, such as average, popularity [3], and PIT [10] etc. However, these methods cannot model the dynamic changes of the group in the face of different decisions.

Considering that different group members will have different influences on group preference, the models [1,8,14] are generated to solve it by assigning corresponding weights to each group member. However, these models tend to prioritize pairwise connections between users, overlooking high-order interactions both within and outside the group. Due to the success of hypergraphs in modeling high-order feature relationships, several group recommendation models [5,9,17] proposed hypergraph convolutional network to capture user and group-level group preferences.

Despite achieving impressive results, the aforementioned methods still have some limitations to be better explored. 1) When it comes to preference aggregation strategies, these methods only take into account collaborative signals between group members. They overlooked the optimization of individual preferences when users are outside of a group, which is problematic considering that groups are composed of individual users. This leads to inaccuracies in aggregating group preference information. 2) the final decision in group decision-making is often influenced by multiple factors, such as individual expertise, the influence of group members, and the general preferences of the group. Most models consider only one aspect without taking into account different perspectives. Oversimplification of such complex factors can lead to a biased understanding of the group's decision-making process, posing limitations to the accuracy of the recommendation system.

To solve the problems, we put forward the model named DDGR(**D**ual-Graph Convolutional Network and **D**ual-View Fusion for **G**roup **R**ecommendation). Firstly, one key consideration is that as users increasingly purchase identical products, it often signals a greater convergence in their preferences. So we model the interest-similarity graph according to the interaction data. To have a more comprehensive understanding of user preferences, we present a dual graph convolutional network that integrates graphs and hypergraphs to capture collaborative information from both within and outside user groups. Third, we design a dual-view attention mechanism fusion strategy that takes into account the impact of group members, their expertise, and the group's overall preferences on the final decision. Considering that each factor has different weights in different situations, we design an adaptive weight fusion strategy to indicate each factor's weight. We adopt a joint training strategy that combines group-item and user-item recommendations during training. The following are the key contributions of our work:

- We establish interest-similarity graphs for individual users and hypergraphs for groups based on interaction data. On this basis, we propose a dual-graph

convolution network that uses hypergraphs and graphs to extract users' collaborative information within and outside the group and capture latent interests.
- We design a dual-view attention mechanism fusion strategy to model multiple factors in the group decision-making process. Meanwhile, we design an adaptive weight fusion strategy to measure the weight of different factors and obtain semantically rich group representations.
- Our proposed method is subject to extensive experimentation, incorporating two real-world datasets. The results clearly demonstrate that the method significantly exceeds most methods for group recommendation tasks.

2 Problem Formulation

In this section, we begin by concentrating on the definition of the group recommendation task. Subsequently, we provide a definition of a hypergraph.

Definition 1 (Group Recommendation). We define the set of users as $\boldsymbol{\mathcal{U}} = \{u_1, u_2, \ldots, u_{|\mathcal{U}|}\}$, the set of items as $\boldsymbol{\mathcal{I}} = \{i_1, i_2, \ldots, i_{|\mathcal{I}|}\}$, and the set of groups as $\boldsymbol{\mathcal{G}} = \{g_1, g_2, \ldots, g_{|\mathcal{G}|}\}$. The t-th group $g_t \in \mathcal{G}$ is a collection of users $\mathcal{G}_t = \{u_1, u_2, \ldots, u_i \ldots, u_{|\mathcal{G}_t|}\}$, where $u_i \in \mathcal{U}$, $|\mathcal{G}_t|$ is the size of $\mathcal{G}_t$. Let $\mathbf{R} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$ denote the user-item interaction matrix, where $r_{ui} = 1$ if the user u interacted with item, otherwise $r_{ui} = 0$. Let $\mathbf{S} \in \mathbb{S}^{|\mathcal{G}| \times |\mathcal{I}|}$ denote the group-item interaction matrix, where $s_{gi} = 1$ if the group g interacted with item i, otherwise $s_{gi} = 0$. In group recommendation, the goal is to suggest a list of items that a target group is likely to be interested in. Formally, this involves developing a function, f_g, which assigns a real-valued score to each item, indicating the probability of the target group g_t interacting with that item: f_g: $\boldsymbol{\mathcal{I}} \rightarrow \mathbb{R}$.

In hypergraphs, each hyperedge can connect multiple nodes, allowing for the representation of various relationships, which can support modeling multidimensional relationships in groups.

Definition 2 (Hypergraph). The hypergraph G is formally defined as $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ exhibits a collection of M distinct vertices, and $\mathcal{E}$ represents the set of hyperedges containing N edges. And each hyperedge $\epsilon \in \mathcal{E}$ can contain multiple vertices. The incidence matrix $\mathbf{H} \in \mathbb{R}^{M \times N}$ can represent hypergraph, where $h_{v\epsilon} = 1$ if the hyperedge ϵ contains the vertex $v \in \mathcal{V}$, otherwise $h_{v\epsilon} = 0$. The diagonal matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$ is utilized to represent the weight of the hyperedges. The degree of each vertex is represented by the diagonal matrix $\mathbf{D}$, where the vertex degree D_v can be calculated as $D_v = \sum_{\epsilon=1}^{M} W_\epsilon H_{v\epsilon}$. Similarly, the diagonal matrix $\mathbf{B}$ is used to denote the degree of each hyperedge, where the hyperedge degree B_ϵ can be determined as $B_\epsilon = \sum_{\epsilon=1}^{N} H_{i\epsilon}$.

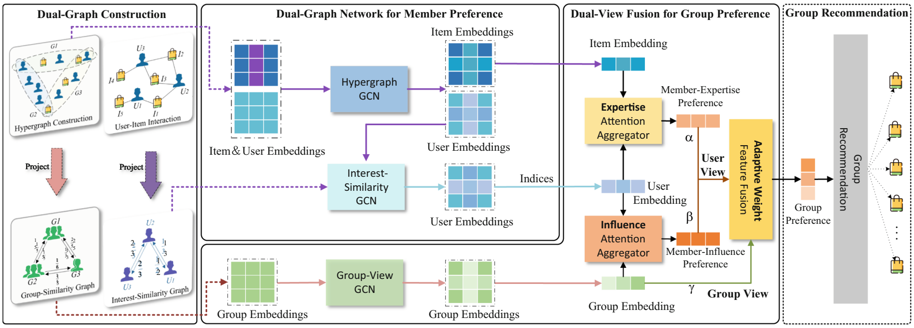

Fig. 1. The overview architecture of our proposed DDGR model, consisting of the member preference learning and group preference learning.

3 Approach

In this section, we introduce the proposed DDGR in detail, and it is composed of three vital parts: Dual-Graph Construction, Dual-graph Network for Member Preference, and Dual-view Fusion for Group Preference. Figure 1 illustrates the overall architecture of DDGR.

3.1 Dual-Graph Construction

Effective group recommendations require establishing suitable connections. We construct a hypergraph to model higher-order relationships between members and items and construct a member interest-similarity graph exploring latent interests when the user is outside the group.

Hypergraph Construction. The transformation from group interaction data to hypergraph $G_h = (\mathcal{V}_h, \mathcal{E}_h)$ is shown in Fig. 2. Group members and items form hyperedge $\mathcal{E}_H = \{u_1, u_2, \ldots, u_{|\mathcal{U}|}, i_1, \ldots, i_{|\mathcal{I}|}\}$. Unlike graphs, the members and items are explicitly connected by hyperedge in the hypergraph. It can extract many-to-many high-order relations from graphs.

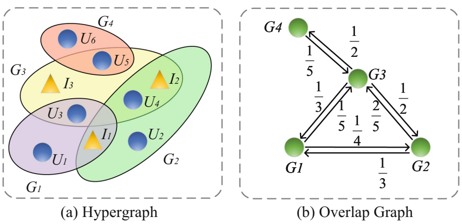

Fig. 2. Example of a hypergraph and overlap graph.

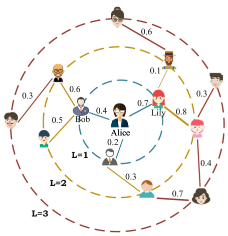

Fig. 3. Example of a member interest-similarity graph.

Member Interest-Similarity Graph Construction. The work [7] indicates that an increased level of interaction with shared products among users often signifies a higher probability of common interests, we design an interest-similarity graph $G_b = (\mathcal{V}_b, \mathcal{E}_b)$ to shows the similarity of interests between users which is shown in Fig. 3. The user i and user j are connected if they interact with common items, and we give edge weight to represent the similarity of user preferences:

$$S(I_i, I_j) = \frac{|I_i \cap I_j|}{|I_i|} \tag{1}$$

where $|I_i|$ represents the total number of items which user i interacted with, $|I_i \cap I_j|$ is the number of items which user i and j interacted together. The weight signifies the proximity between the two users, with an unequal influence on both sides.

3.2 Dual-Graph Network for Member Preference

The dual-graph network was proposed to learn collaborative high-order representation of users and items based on the hypegraph, and capture cross-user collaborative information of users based on the member interest-similarity graph.

We introduce the hypergraph convolution operation to exploit high-order interactions to learn the user's and item's dynamic representation. Let ${\mathbf{X_h}}^{(0)} = [\mathbf{U}; \mathbf{I}]$ be the input of hypergraph convolutional network, which is the concatenation of user embeddings $\mathbf{U} \in \mathbb{R}^{|U| \times d}$ and item embeddings $\mathbf{I} \in \mathbb{R}^{|I| \times d}$. Building upon the spectral hypergraph convolution proposed in [4], the work defines its hypergraph convolution as:

$${\mathbf{X_h}}^{(l+1)} = \mathbf{D}^{-1}\mathbf{H}\mathbf{W}\mathbf{B}^{-1}\mathbf{H}^T{\mathbf{X_h}}^{(l)}{\mathbf{\Theta_h}}^{(l)} \tag{2}$$

where $\mathbf{D}$ denotes the vertex degree matrix of the hypergraph, and $\mathbf{B}$ denotes the hyperedge degree matrix. $\mathbf{W}$ is initialized as an identity matrix, indicating the assignment of equal weights for all hyperedges. $\mathbf{H}$ is an incidence matrix to delineate the relationship between nodes and hyperedges. ${\mathbf{\Theta_h}}^{(l)}$ is the learnable weight matrix. After passing $\mathbf{X}_0$ through $\mathbf{L}$ hypergraph convolutional layers, we get the final embeddings $\mathbf{X}_h$ by averaging embeddings obtained at each layer, where $\mathbf{X}_h = \frac{1}{L+1}\sum_{l=0}^{L}\mathbf{X}_h^{(l)}$. By leveraging the node-edge-node transformation, the hypergraph convolutional network can efficiently extract high-order correlations on the hypergraph.

The hypergraph network is employed to capture users' high-dimensional relation within the group; however, it does not acquire collaborative information from users outside the group. The member Interest-similarity graph depicts the similarity of interest and contains cross-user information. Based on this, we aim to mine the mutual influence between users and discover users' potential interests. So we apply the graph convolution operation to capture communication features. Let user embeddings $\mathbf{X_b} \in \mathbb{R}^{|U| \times d}$, which is the output of the hypergraph convolution layer, be the input of the interest-similarity graph convolutional network. The graph convolution operation is defined as:

$${\mathbf{X_b}}^{(l+1)} = \mathbf{D_b^{-1/2}}\mathbf{A_b}\mathbf{D_b^{-1/2}}{\mathbf{X_b}}^{(l)}{\mathbf{\Theta_b}}^{(l)} \tag{3}$$

where $\mathbf{A_b} \in \mathbb{R}^{|U| \times |U|}$ defines An incidence matrix, $|U|$ is the number of users and $\mathbf{A_{p,q}} = \mathbf{W_{p,q}}$ according to definition of Eq. (1). $\mathbf{D_b} \in \mathbb{R}^{|U| \times |U|}$ is a diagonal degree matrix where $\mathbf{D}_{p,q} = \sum_{q=1}^{|U|} \mathbf{A}_{p,q}$.

3.3 Dual-View Fusion for Group Preference

Dual-graph network helps count for group member preference but has a relatively minor contribution toward group decision-making. Making decisions is a complex process for groups, it will be affected by many factors, such as individual expertise, the influence of group members, and the general preferences of the group. So we propose a dual-view fusion strategy to capture these factors from *member* and *group* views and get the most reasonable group preference possible.

From the members' view, we consider two factors: *Member-expertise Preference* and *Member-influence Preference*. For Member-expertise Preference, imagine a scenario where a group of friends is selecting products. If a member possesses expertise in assessing electronic products, his influence on the group's decision-making process is likely to be more significant. For Member-influence Preference, a company group discusses where to travel, and the leader may have greater decision-making power to make decisions on behalf of the group even though most members have different choices. So the varying identities and positions of group members can lead to fluctuations in the levels of influence they exert in group decision-making. We define $\alpha^E(i,t)$ denotes the weight of user u_i in the group's decision-making with respect to item i_t and $\alpha^P(i,l)$ represents the score of the user u_i's influence in the group g_l. User embedding $\mathbf{u}_i$, item embedding $\mathbf{i}_t$ and group embedding $\mathbf{g}_l$ is the input of neural attention network, which is defined as:

$$o(i,t) = \mathbf{h}^T \operatorname{ReLU}\left(\mathbf{P}_u \mathbf{u}_i + \mathbf{P}_i \mathbf{i}_t + \mathbf{b}\right), \quad \alpha^E(i,t) = \frac{\exp o(i,t)}{\sum_{i' \in \mathcal{G}_l} \exp o\left(i', t\right)} \tag{4}$$

$$p(i,l) = \mathbf{h}^T \operatorname{ReLU}\left(\mathbf{P}_u \mathbf{u}_i + \mathbf{P}_g \mathbf{g}_l + \mathbf{b}\right), \quad \alpha^P(i,l) = \frac{\exp p(i,l)}{\sum_{i' \in \mathcal{G}_l} \exp p\left(i', l\right)} \tag{5}$$

In the attention network, the activation function utilized for the hidden layer is ReLU, and we use a weight vector $\mathbf{h}$ to project the score $o(t,j)$. Finally, we calculate the group's member-expertise representation $\mathbf{g}_l^E$ and the group's member-influence representation $\mathbf{g}_l^P$ using a weighted sum operation:

$$\mathbf{g}_l^E = \sum_{\mathbf{u}_i \in \mathcal{G}_l} \alpha^E(i,t)\mathbf{u}_i, \quad \mathbf{g}_l^P = \sum_{\mathbf{u}_i \in \mathcal{G}_l} \alpha^P(i,l)\mathbf{u}_i. \tag{6}$$

By incorporating an attention mechanism, each member's contribution to group is learned from interaction data and varies dynamically based on different scenes.

In addition to aggregating the embeddings of group members, we employ dedicated group embeddings to represent groups from the group view. The intention is to take intrinsic group-level preferences into account and capture collaborative information between groups. Group decisions may not always align with individual preferences as they aim to satisfy the collective interests of the group. When a family group discusses about restaurant choices, individual preferences may differ and have their preferance. Finally, the group's choice of a restaurant aims to satisfy everyone's taste rather than selecting the one that's liked the most by everyone individually. Groups often have common members or shared purchases, leading to interaction between groups. The similarity between two groups will be higher if they share more common members or items. So we employ an overlap graph to map out inter-group connections and hidden interests, with Fig. 2 detailing the transformation and showcasing the interactions and proximity among groups. Group embeddings $\mathbf{G} \in \mathbb{R}^{|G|\times d}$ is the input of graph convolutional operation:

$$\mathbf{G}^{(l+1)} = \mathbf{D_g}^{-\mathbf{1/2}}\mathbf{A_g}\mathbf{D_g}^{-\mathbf{1/2}}\mathbf{G}^{(l)}\mathbf{\Theta_g}^{(l)} \tag{7}$$

where $\mathbf{D_g} \in \mathbb{R}^{|G|\times|G|}$ is a diagonal degree matrix and $\mathbf{D}_{p,q} = \sum_{q=1}^{|G|}\mathbf{A}_{p,q}$, $|G|$ is the number of groups. The incidence matrix is defined as $\mathbf{A_g} \in \mathbb{R}^{|G|\times|G|}$ and $\mathbf{A_{p,q}} = \mathbf{W_{p,q}}$. We set the matrix $\mathbf{A_g} = \mathbf{A_g} + \mathbf{I}$, where $\mathbf{I}$ is an identity matrix. Finally, we calculate the group-view representation $\mathbf{g}_l^G$.

The complexity of group decision-making lies not only in being influenced by multiple factors but also in considering the weight of each factor. So we propose an adaptive weight fusion strategy to capture the weight of each factor, resulting in a more reasonable group preference representation $\mathbf{g}_l$. We compute the influence score α of member-expertise factor:

$$\alpha = \mathbf{w}^T \operatorname{ReLU}\left(\mathbf{W}_f\left[\mathbf{g}_l^E \odot \mathbf{i}_h; \mathbf{g}_l^E; \mathbf{i}_h\right] + \mathbf{b}_f\right) \tag{8}$$

Similarly, for member-influence and group-level preference, we can get scores β and γ, respectively. By weighted addition, we get the group representation $\mathbf{g}_l$:

$$\mathbf{g}_l = \alpha\mathbf{g}_l^E + \beta\mathbf{g}_l^P + \gamma\mathbf{g}_l^G \tag{9}$$

We utilize the adaptive weight fusion network to extract key features from the three hidden group representations and blend them together seamlessly.

3.4 Group Recommendation and Model Training

After utilizing the Dual-graph Network and Dual-view Fusion, we can effectively capture cooperative signals between groups, discern the potential preferences of group members, identify group decision factors, and acquire a semantically rich group representation. Subsequently, we can proceed to match items of interest to the group.

Given that our objective is to rank items, we utilize the regression-based pairwise loss that has been motivated by the work [11].The group training set $\mathcal{O}_G$ is defined as a collection of triplets (i, j, j'), each triplet represents a scenario in which group g_i interacted with item i_j but has no prior interaction with i'_j:

$$\mathcal{L}_{\text{group}} = \sum_{(i,j,j')\in\mathcal{O}_G}\left(\hat{y}_{ij} - \hat{y}_{ij'} - 1\right)^2 \tag{10}$$

Considering the data sparsity of group interactions, we adopt a joint learning strategy that simultaneously combines the group-item interaction and user-item interaction data to enhance recommendation tasks. Similarly, we define $\mathcal{L}_{\text{user}}$ the same pairwise loss function as Eq. (10) to optimize user recommendation. As a consequence, the overall loss function is a combination of the losses incurred by both the group and user pairs:

$$\mathcal{L} = \lambda \mathcal{L}_{\text{group}} + (1 - \lambda)\mathcal{L}_{\text{user}} \tag{11}$$

where hyper-parameter λ plays a role in balancing the weights between the group and user losses. Notably, the primary focus of our task is group recommendation.

4 Experiments

This section covers the details of our experiments, including the datasets, baseline methods, and evaluation metrics. To investigate the following research questions, we carried out rigorous experiments on two openly accessible datasets.

4.1 Experimental Dataset and Setup

Datasets. To assess the performance of our proposed model, we evaluated it on two real-world datasets, namely MaFengWo and CAMRa2011. The MaFengWo dataset is a collection of user-generated travel experiences and group journeys. The CAMRa2011 dataset consists of real-world movie ratings from both individuals and groups. Detailed statistics for both datasets are presented in Table 1.

Table 1. The statistics of two datasets. U-I and G-I indicate user and item interactions, and group and item interactions, respectively.

Datasets	Users	Groups	Items	U-I	G-I	Avg. Group size
MaFengWo	5,275	995	1513	39,761	3,595	7.19
CAMRa2011	602	290	7,710	116,344	145,068	2.08

Competing Methods. To validate the performance of our group recommendation model, we conducted a comparative analysis with the following baseline: **Popularity** [3], **PIT** [10], and **COM** [16] models are mainstream probabilistic model. Baseline also includes deep learning models, which can be divided into the following: the neural network-based model(**NCF** [7]), the attention-based model(**AGREE** [1], **SIGR** [14], **GAME** [8], **GroupSA** [6], **SoAGREE** [2]), and the Hypergprah-based model(**HCR** [9], **ConsRec** [12]).

Evaluation Metrics. For the purpose of assessing group recommendation performance, we utilized commonly-used metrics called Hits Ratio (HR) and

Normalized Discounted Cumulative Gain (NDCG) at top-K recommendation list [15].

$$HR@K = \frac{\#hit@K}{|\mathcal{D}_{\text{test}}|}, \quad NDCG@K = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{i=1}^{|\mathcal{D}_{\text{test}}|} \frac{1}{\log_2(p_{i@K}+1)} \tag{12}$$

Consequently, we divided the data into two separate sets: training set($\mathcal{D}_{train}$) and testing set($\mathcal{D}_{test}$). We performed a random sampling of the missing data to generate negative instances for each positive instance in the testing set.

4.2 Experimental Results and Analysis

In this section, we present a comparison of the recommendation performance of our proposed model to that of several baseline models. The group and user performance for the CAMRa2011 and MaFengWo, as shown in Tables 2 and 3, respectively. From the result, we can observe that:

Table 2. Performance comparison of top-K **group** recommendation on datasets.

Datasets	MaFengWo				CAMRa2011			
Metric	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
Popularity	0.3115	0.2169	0.4251	0.2537	0.4324	0.2825	0.5793	0.3302
PIT	0.4159	0.2965	0.5012	0.3382	0.5632	0.3741	0.7523	0.4492
COM	0.4432	0.3325	0.5528	0.3812	0.5798	0.3785	0.7695	0.4385
NCF	0.4701	0.3657	0.6269	0.4141	0.5803	0.3896	0.7693	0.4448
AGREE	0.4729	0.3694	0.6321	0.4203	0.5879	0.3933	0.7789	0.4530
SIGR	0.5041	0.3955	0.6569	0.4573	0.6172	0.4473	0.8158	0.4870
GroupSA	0.4876	0.3871	0.6409	0.4351	0.5906	0.4163	0.7800	0.4667
GAME	0.4759	0.3956	0.6346	0.4482	0.5953	0.4356	0.7957	0.4713
SoAGREE	0.4898	0.3807	0.6481	0.4301	0.5883	0.3955	0.7807	0.4575
HCR	0.7759	0.6611	0.8503	0.6852	<u>0.6772</u>	<u>0.6115</u>	0.8193	<u>0.6576</u>
ConsRec	<u>0.8844</u>	<u>0.7692</u>	<u>0.9156</u>	<u>0.7794</u>	0.6407	0.4358	<u>0.8248</u>	0.4945
DDGR	**0.9126**	**0.8517**	**0.9317**	**0.8595**	**0.7648**	**0.7548**	**0.8386**	**0.7782**

Table 3. Performance comparison of top-K **user** recommendation on datasets.

Datasets	MaFengWo				CAMRa2011			
Metric	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
Popularity	0.4047	0.2876	0.4971	0.3172	0.4624	0.3104	0.6026	0.3560
NCF	0.6363	0.5432	0.7417	0.5733	0.6119	0.4018	0.7894	0.4535
AGREE	0.6357	0.5481	0.7403	0.5738	0.6196	0.4098	0.7897	0.4627
SoAGREE	0.6510	0.5612	0.7610	0.5775	0.6223	0.4118	0.7967	0.4687
HCR	<u>0.7571</u>	0.6703	0.8290	0.6937	0.6731	<u>0.4608</u>	**0.8595**	<u>0.5219</u>
ConsRec	0.7725	<u>0.6884</u>	<u>0.8404</u>	<u>0.7107</u>	<u>0.6774</u>	0.4568	<u>0.8412</u>	0.5104
DDGR	**0.7930**	**0.7285**	**0.8518**	**0.7445**	**0.7475**	**0.7255**	0.8375	**0.7540**

DDGR consistently outperforms other baseline models, achieving the highest performance on both group recommendation datasets. The great improvements in performance provide further evidence of the effectiveness of our model for capturing group representation. DDGR shows a more significant improvement over the second-best model at NDCG@K compared to HR@K. It highlights DDGR's ability to prioritize items that align with the interests of user groups, resulting in more precise recommendations. It also showcases DDGR's capacity to dynamically model the decision-making process within groups and accurately capture their preferences.

In the user recommendation task, DDGR achieved excellent results in all metrics, essentially reaching optimal precision. This observation highlights that our proposed method has the ability to capture users' latent interests, thereby optimizing their preference representations.

To further explore the importance of Dual-Graph Networks and Dual-View Fusion Strategy, we conducted several ablation studies. Figure 4 shows the results of DDGR and the two variants, DDGR-G denotes the ablated model "DDGR with Dual-Graph Network only" and DDGR-V denotes "DDGR with dual-view fusion only". In both benchmark datasets, it is consistently observed that DDGR outperforms DDGR-G and DDGR-V across all evaluation metrics, indicating that Dual-Graph Network and Dual-View Fusion Strategy can facilitate each other, and combining them enables a more comprehensive capture of group interests. DDGR-V performs better than DDGR-G on both datasets. It indicates that Dual-view fusion strategy has a larger impact than the Dual-Graph Network on group representation of learning.

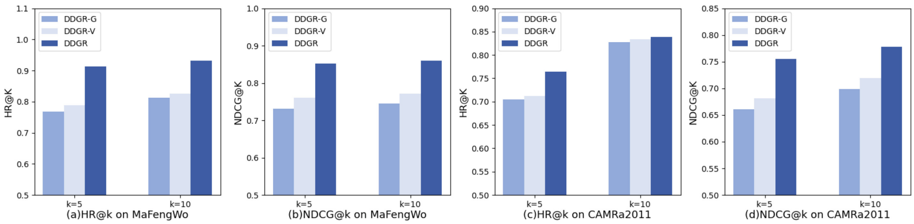

Fig. 4. The performance comparison between DDGR and its variants.

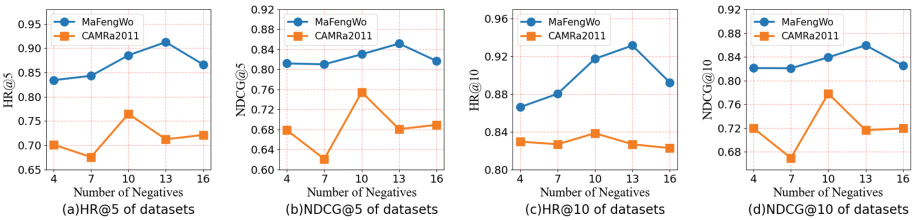

Fig. 5. The trend of HR and NDCG w.r.t. the number of negative samples.

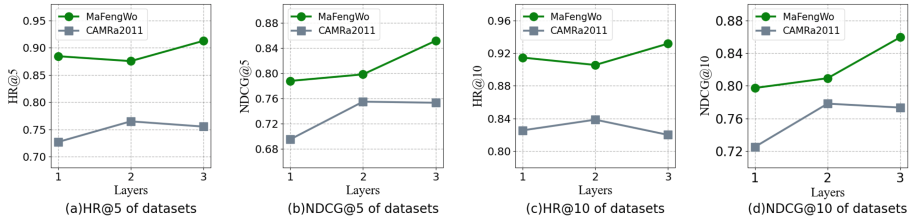

Fig. 6. The trend of HR and NDCG w.r.t. the number of convolutional layers.

4.3 Parameter Sensitivity

To investigate the impact of negative sampling and the number of convolutional layers on the performance of our model, we conducted a series of experiments. The impact of negative sampling on DDGR is presented in Fig. 5. We can observe that the performance of DDGR will not consistently increase as more negative samples are added. If there are too many negative samples in the sampling process, it's possible that the model will select items that the group would potentially be interested in, which leads to the deviation of group representation learning. So it is necessary to control them within a reasonable range. In Fig. 6, as the number of layers increases, DDGR does not consistently demonstrate improved performance on CAMRa2011. One potential explanation is that as the number of layers increases, nodes in higher levels may acquire the issue of over-smoothing. The increased complexity introduced by additional layers may not always align with the underlying data, causing difficulties in distinguishing groups and leading to adverse effects in performance gains.

5 Related Works

Group Recommendation. Early works on group recommendation used two approaches: *score aggregation* and *preference aggregation*. Score aggregation methods, such as average, least misery, and maximum satisfaction, employ a fixed strategy to calculate group representations. However, these methods couldn't capture the dynamic group decision process and neglected changes in group. Recently, with the successful development of the deep neural network, model-based approaches have achieved significant advances. AGREE [1], GroupSA [6] incorporate attention mechanisms to automatically learn and assign the user's corresponding weight to optimize the preference aggregation strategy. GAME [8] utilizes the heterogeneous information network to generate multi-view embeddings for nodes and members' weights. Although these methods offer a dynamic aggregation process of group preference, they all fail to capture complex and higher-order user interactions.

Hypergraph-Based Recommendation. In real-world scenarios, relationships among objects are more complicated than simple pairwise connections;

squeezing intricate relationships into paired relationships naively will inevitably lead to valuable information loss. Xia *et al.* [13] utilized a hypergraph neural network to enhance session-based recommendation tasks by modeling session-based data as a hypergraph. Similarly, Zhang *et al.* [17] proposed a hierarchical hypergraph neural network based on user and group-level hypergraphs. The HCR [9] uses a dual-channel hypergraph convolutional network that extracts collaborative information and group similarity to enhance performance. HyperGroup [5] proposes connecting groups as an overlapping set network to capture the similarity of groups and learn accurate group representations from group-item interactions.

6 Conclusion and Future Work

In this work, we introduced a novel model called DDGR to tackle the key challenges in group recommendation tasks: 1) how to obtain comprehensive group members' preferences from interaction data, 2) how to get a semantically rich group representation by emulating the decision-making processes. For group members' preferences, we construct the dual-graph network that combines the advantages of both hypergraphs and graphs to capture high-order and pairwise relationships between users. For group representation learning, we propose a dual-view fusion strategy that considers the impact of multiple factors in decision-making from two views, allowing us to accurately simulate the decision-making process. Comprehensive experiments on datasets demonstrate that DDGR surpasses other approaches in group recommendation tasks. In the future, we will combine group recommendations with large language models.

Acknowledgments. This work was supported by National Natural Science Foundation of China (No.62272290, 62172088) and Shanghai Natural Science Foundation(No.21ZR1400400).

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 and 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 Trans. Knowl. Data Eng. **33**(3), 1195–1209 (2019)
3. Cremonesi, P., Koren, Y., Turrin, R.: Performance of recommender algorithms on top-N recommendation tasks. In: ACM Conference on Recommender Systems, pp. 39–46 (2010)
4. Feng, Y., You, H., Zhang, Z., Ji, R., Gao, Y.: Hypergraph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 3558–3565 (2019)
5. Guo, L., Yin, H., Chen, T., Zhang, X., Zheng, K.: Hierarchical hyperedge embedding-based representation learning for group recommendation. ACM Trans. Inf. Syst. **40**(1), 1–27 (2021)

6. Guo, L., Yin, H., Wang, Q., Cui, B., Huang, Z., Cui, L.: Group recommendation with latent voting mechanism. In: IEEE International Conference on Data Engineering, pp. 121–132. IEEE (2020)
7. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: International Conference on World Wide Web, pp. 173–182 (2017)
8. He, Z., Chow, C.Y., Zhang, J.D.: GAME: learning graphical and attentive multi-view embeddings for occasional group recommendation. In: International ACM SIGIR Conference on Research and Development in Information Retrieval (2020)
9. Jia, R., Zhou, X., Dong, L., Pan, S.: Hypergraph convolutional network for group recommendation. In: IEEE International Conference on Data Mining, pp. 260–269. IEEE (2021)
10. Liu, X., Tian, Y., Ye, M., Lee, W.C.: Exploring personal impact for group recommendation. In: ACM International Conference on Information and Knowledge Management, pp. 674–683 (2012)
11. Wang, X., He, X., Nie, L., Chua, T.S.: Item silk road: recommending items from information domains to social users. In: International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 185–194 (2017)
12. Wu, X., et al.: ConsRec: learning consensus behind interactions for group recommendation. In: Proceedings of the ACM Web Conference 2023, pp. 240–250 (2023)
13. Xia, X., Yin, H., Yu, J., Wang, Q., Cui, L., Zhang, X.: Self-supervised hypergraph convolutional networks for session-based recommendation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 4503–4511 (2021)
14. Yin, H., Wang, Q., Zheng, K., Li, Z., Yang, J., Zhou, X.: Social influence-based group representation learning for group recommendation. In: International Conference on Data Engineering, pp. 566–577. IEEE (2019)
15. Yin, H., Wang, Q., Zheng, K., Li, Z., Zhou, X.: Overcoming data sparsity in group recommendation. IEEE Trans. Knowl. Data Eng. **34**, 3447–3460 (2020)
16. Yuan, Q., Cong, G., Lin, C.Y.: COM: a generative model for group recommendation. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 163–172 (2014)
17. Zhang, J., Gao, M., Yu, J., Guo, L., Li, J., Yin, H.: Double-scale self-supervised hypergraph learning for group recommendation. In: ACM International Conference on Information and Knowledge Management, pp. 2557–2567 (2021)