2021 IEEE 37th International Conference on Data Engineering (ICDE) | 978-1-7281-9184-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICDE51399.2021.00136

Group-Buying Recommendation for Social E-Commerce

Jun Zhang, Chen Gao[§], Depeng Jin, Yong Li
Beijing National Research Center for Information Science and Technology
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
zhangjun990222@gmail.com, gc16@mails.tsinghua.edu.cn, {jindp, liyong07}@tsinghua.edu.cn

***Abstract*—Group buying, as an emerging form of purchase in social e-commerce websites, such as Pinduoduo[1], has recently achieved great success. In this new business model, users, *initiator*, can launch a group and share products to their social networks, and when there are enough friends, *participants*, join it, the deal is clinched. Group-buying recommendation for social e-commerce, which recommends an item list when users want to launch a group, plays an important role in the group success ratio and sales. However, designing a personalized recommendation model for group buying is an entirely new problem that is seldom explored. In this work, we take the first step to approach the problem of group-buying recommendation for social e-commerce and develop a GBGCN method (short for Group-Buying Graph Convolutional Network). Considering there are multiple types of behaviors (launch and join) and structured social network data, we first propose to construct directed heterogeneous graphs to represent behavioral data and social networks. We then develop a graph convolutional network model with multi-view embedding propagation, which can extract the complicated high-order graph structure to learn the embeddings. Last, since a failed group-buying implies rich preferences of the initiator and participants, we design a double-pairwise loss function to distill such preference signals. We collect a real-world dataset of group-buying and conduct experiments to evaluate the performance. Empirical results demonstrate that our proposed GBGCN can significantly outperform baseline methods by 2.69%-7.36%. The codes and the dataset are released at https://github.com/Sweetnow/group-buying-recommendation.**

***Index Terms*—Group-buying Recommendation; Graph Convolutional Network; Social Network**

I. Introduction

Group buying, as a new form of business model in social e-commerce websites, such as Pinduoduo, becomes increasingly popular. It was reported by Forbes[2] that group-buying has achieved great access in attracting new users, increasing user activities, improving platform profits, etc. Specifically, in this purchase model, a user can share the product, which he/she is interested in, broadcasting to his/her friends in social media. This stage is called *lanuching* a group and the user is called *initiator*. If one or several friend(s) who also want(s) to purchase the shared product, he/she (they) can take participate in the group. This stage is called *joining* a group and such friend(s) is (are) called *participant(s)*. When there are enough users in this group, group buying is clinched. This stage is called *clinching* a group. The whole process described above is illustrated in Figure 1. In short, a group buying can be understood that a group of users interact with a product, through the initiator's sharing and participants' following.

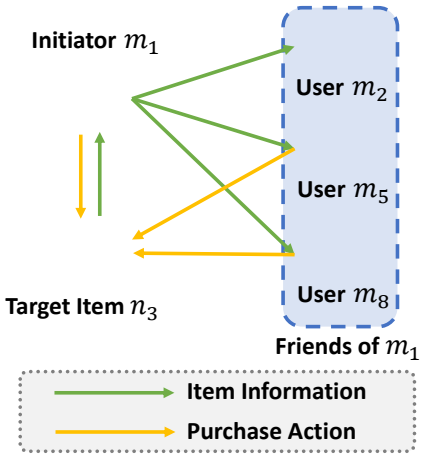

Fig. 1: The illustration of the process of one user launching a group buying, sharing the target item to friends and clinching the group buying.

The problem of **group-buying recommendation for social e-commerce** is defined to recommend product-candidates for the initiator to launch a group buying. With suitable recommended products customized for the initiator, the preferences of the initiator and the participants are more likely to be satisfied, and the group buying becomes more probable to clinch. Thus, it is essential and beneficial for the social e-commerce website to design powerful models for the group-buying recommendation. However, this is a completely new problem that is rarely explored by existing works. Whether a group succeeds to clinch or not is determined by multiple factors, including the initiator's interest, participants' interest, initiator's influence on other users, etc.

In fact, there are two kinds of recommendation tasks related to the problem of group-buying recommendation in social e-commence: social recommendation [1]–[12] and group recommendation [13]–[21]. Social recommendation is defined as utilizing social-relational data to enhance recommendation. A commonly accepted paradigm is to make use of friends' preference to help model users, assuming friends tend to have similar tastes. However, in the problem of group-buying recommenda-

[§]Chen Gao is the Corresponding Author.
[1]https://www.pinduoduo.com/
[2]https://www.forbes.com/sites/eladnatanson/2019/12/04/the-miraculous-rise-of-pinduoduo-and-its-lessons

978-1-7281-9184-3/21/$31.00 ©2021 IEEE
DOI 10.1109/ICDE51399.2021.00136

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

tion, the form of users interacting with products is completely different, where the launching and joining behaviors cannot be handled. On the other hand, group recommendation is defined as to recommend one product to a group. For example, in location-based services, the platform can recommend a travel destination to a family. However, in group recommendation, the group is always fixed (a family, for example). While in the group buying, the group is dynamic and a sub-sample from the entire social network, and how to determine the potential group of users is important. Therefore, existing solutions of both social recommendation and group recommendation cannot serve the task of group-buying recommendation.

Generally, the problem of group-buying recommendation for social e-commerce is faced with the following three major challenges:

- **Multiple roles of users in group buying.** In our problem, users can both launch a group or join a friend's group. In other words, users can play as different roles, initiators or participants, with different interests and motivations, making it challenging to capture accurately.
- **Complicated social influence.** Despite user interests, the initiator's influence on the social network is another significant factor determining whether the friend joins. It is challenging to model social influence and user interests jointly.
- **Complicated feedback of group-buying interaction data.** In our problem, the predictive signal is revealed by interaction data, since it is related to many users' preferences. Thus, it is challenging to learn from such complicated feedback.

To overcome the above mentioned challenges in our problem, we propose a solution named GBGCN (short for **G**roup-**B**uying **G**raph **C**onvolutional **N**etworks). Our GBGCN approaches the problem via graph convolutional network-based representation learning on the graph-structured group-buying data. Precisely, to capture the multiple roles of users in our problem, we develop in-view user-item embedding propagation to extract the preferences of different roles. We then employ cross-view embedding propagation for learning social influence and generate comprehensive representation. Last, we design a double-pairwise loss function to distill predictive signals that are related to many users in a group.

To summarize, the main contributions of this work are as follows,

- To the best of our knowledge, we are the first to approach the problem of personalized recommendation for group buying in social e-commerce, which is significant but not explored yet.
- We develop a graph convolutional network (GCN) based method with embedding propagations to model multiple user roles and social influence and a double-pairwise loss function to utilize the complex feedback in our problem.
- We collect a precious dataset from a large-scale social e-commerce website and conduct extensive experiments to evaluate the performance of GBGCN. Empirical results demonstrate that our GBGCN can help improve the recommendation performance significantly.

The remainder of the paper is as follows. We first formalize the problem in Section II and present our proposed method in Section III. We then conduct experiments in Section IV and review related work in Section V. Last, we conclude the paper in Section VI.

TABLE I: A list of commonly used notations.

Notat.	Description
$\mathcal{M}$	The set of users.
$\mathcal{N}$	The set of items.
P	The number of users.
Q	The number of items.
m	User's ID. (The subscript $i(p)$ if existed means the user is in initiator view (participant view).)
n	Item's ID. (The subscript is the same as m.)
M_p	Participant set.
t_n	The threshold of group size for item n.
b	One group-buying behavior.
B	The group-buying behavior set.
$\mathbf{S}$	The social network.
$\sigma(\cdot)$	The activation function.
$\mu(\cdot)$	Sigmoid function.
α	The role coefficient.
β	The loss coefficient.
W	The transformation matrix. (The first subscript denotes source and the second denotes target.)
b	The transformation bias. (The subscripts are the same as W.)
$\mathcal{L}$	The fine-grained loss function.
$\mathcal{L}_{-(+)}$	The loss function for failed (successful) behaviors.
$\mathcal{G}$	The set of the directed heterogeneous graphs.
$G_{i(p,s)}$	The directed initiator-item (participant-item, initiator-participant) interaction graph.
m_{p_j}	The j-th participant.
$\mathcal{N}_s^{I(O)}(\cdot)$	A function that mapping an input vertex to its incoming (outgoing) neighborhood in G_s.
$\mathcal{N}_{i(p)}(\cdot)$	A function that mapping an input vertex to its neighborhood in $G_{i(p)}$.
d	The embedding size.
u_m	The embedding of user m. (The second subscript if existed means which view it belongs to and the superscipt if existed means which layer generates it.)
v_n	The embedding of item n. (The subscript and superscipt are the same as u_m.)
L	The number of layers.
l	The l-th layer.
$\cdot\|\|\cdot$	Concatenation of two vectors.
y_{mn}	The prediction score of user m with item n.
$B_{-(+)}$	The failed (successful) part of B.

II. Problem Formulation

In social e-commerce, users can share items in her/his social networks, and group buying becomes a remarkable business model. In this model, users can launch a group buying and share it to online social networks. When there are friends taking participants in the group, the group-buying is clinched, or in other words, *succeeded*. If there are no or not enough friends to join, the group buying will be called a *failed* one.

Thus, users have two kinds of roles in group buying, *initiators*, who launch the group buying and sharing products to social networks and *participants* who join the group buying launched by his/her friends. The product, which is the target of the group buying, can be called *target item*. The e-commerce platform needs to provide recommendations to help users choose the proper target item to launch the group buying as an initiator. Let $\mathcal{M}$ and $\mathcal{N}$ denote the sets of users and items of which the sizes are P and Q, respectively.

The behavioral records of group buying are denoted as B consisting of triads of which one record is denoted as $b = \langle m_i, n, M_p \rangle$, where m_i denote initiator user, n denotes the

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

target item, and the M_p denotes the set of participants. The platform sets a threshold t_n for each item when the item is launched, which is always much less than the number of user's friends, and the group buying can be called a failed one if $|M_p|$ is less than t_n. Note that users in M_p are from the online social network of the initiator user m_i. Here we use $\mathbf{S}$, a $P \times P$ symmetric binary matrix to denote the social network. If user m and m' are friends in online social network, then $S_{mm'} = 1$ else $S_{mm'} = 0$.

With the above background and notations, we can formulate the problem of group-buying recommendation for social e-commerce as follows:

Input: Group-buying behaviors B and online social relations $\mathbf{S}$.

Output: A model that calculates a score for representing the unnormalized probability of user m launching a *successful* group to buy item n.

After obtaining the model, for a given user m_i, we can utilize the model to score all items and select the top-ranked items as the recommendation results for m_i.

III. Methodology

To solve the problem of group-buying recommendation for social e-commerce and overcome the three major challenges mentioned above, we organize multiple roles and complex interactions into directed heterogeneous graphs and introduce multi-graph convolution networks to obtain precise embedding representation from the graphs. We name our proposed solution GBGCN. Figure 2 illustrates the architecture of our GBGCN model, which is made up of the four parts as follows.

- **Raw Embedding Layer.** This layer initializes embeddings for user and item as raw representation regardless of roles.
- **In-View Propagation.** To distinguish user interests and item properties, we divide the group-buying behaviors into initiator-item interactions and participant-item interactions. After this, we refine better representation, considering the two roles by propagating information on the two user-item interaction bipartite graphs.
- **Cross-View Propagation.** In this layer, we use social interactions, sharing behaviors and joining behaviors, to guide the information propagation between different roles and model user influence. User-item interactions are utilized again for helping generate exhaustive representation. Fully connected layers (FC) are applied to transform information from source subspaces to target subspaces and extract the meaning parts.
- **Prediction.** Finally, we adopt the inner product to capture user's interests for the given item according to his/her role. And then, the layer outputs the combination of the initiator's preference and his/her friends' preference as the score of launching a successful group buying.

A. Constructing Directed Heterogeneous Graphs

In order to utilize all categories of behaviors, directed heterogeneous graphs are built to represent the two roles and multiplex interactions in our problem. We firstly introduce the concept of *views*. In our problem, both launching and joining can be regarded as user-item interactions. However, users may have potentially different interests when launching and joining. That is, an initiator should consider his/her friends' interests before launching and a participant should consider not only his/her own interests but also whether the relationship is close or not before joining. Therefore, we treat the whole user-item interactions as two parts, initiator-item interactions and participant-item interactions. Meanwhile, the graphs for organizing all behaviors consist of two views, *initiator view* that contains initiator-item interactions and *participant view* that contains participant-item interactions. Besides, we consider initiator-participant interactions, sharing, as cross-view connections. These settings are illustrated in Figure 3.

Following the settings above, we define the heterogeneous graphs as $\mathcal{G} = \{G_i, G_p, G_s\}$, where G_i, G_p, G_s denote the initiator view, the participant view and the sharing relations, respectively. Vertices in G_i and G_p both consist of user nodes $m \in \mathcal{M}$ and item nodes $n \in \mathcal{N}$. Vertices in G_s consist of user nodes $m \in \mathcal{M}$. For directed edges in graphs, considering a group-buying behavior $b = \langle m_i, n, M_p \rangle$, where $M_p = \{m_{p_1}, m_{p_2}, \cdots, m_{p_{|M_p|}}\}$, G_i contains a bidirectional edge (m_i, n), G_p contains $|M_p|$ bidirectional edges $(m_{p_j}, n), j = 1, 2, \cdots, |M_p|$ and G_s contains $|M_p|$ directed edges $(m_i, m_{p_j}), j = 1, 2, \cdots, |M_p|$ from initiator to participants. It is noted that we will use incoming neighborhood $\mathcal{N}_s^I$ and outgoing neighborhood $\mathcal{N}_s^O$ of G_s to distinguish shared and sharing behaviors in cross-view propagation below.

B. Multi Graph Convolution Networks

1) Raw Embedding Layer: Similar to the existing graph-based recommendation works [22]–[24], we represent each user/item ID with a vector representation as known as embedding to characterize the latent features. Formally, we represent user m and item n with embedding $u_m \in \mathbb{R}^d$ and $v_n \in \mathbb{R}^d$, where d is the embedding size.

2) In-View Propagation: It is found by existing researches on graph-based recommeders [25], [26] that message propagation on the user-item bipartite graph is able to explicitly utilize collaborative signals and distill useful information efficiently from user-item interactions. Intuitively, user's first-order neighbors on the user-item interaction graph, the purchased items represent the user's preference directly, and the second-order neighbors are often considered to have similar interests. In our problem, user-item interaction graphs are divided into two parts, initiator view and participant view. Each view carries the complete interaction records under the corresponding user role. Inspired by these, we devise graph convolution layers without FC layers following [26] to perform the embedding process on the two views and generate the role-specific embeddings from the raw embeddings.

The process in initiator view can be fomulated as follows:

$$
\begin{aligned}
u_{m,i}^{(l)} &= \frac{1}{|\mathcal{N}_i(m)|} \sum_{n' \in \mathcal{N}_i(m)} v_{n',i}^{(l-1)}, \\
v_{n,i}^{(l)} &= \frac{1}{|\mathcal{N}_i(n)|} \sum_{m' \in \mathcal{N}_i(n)} u_{m',i}^{(l-1)}, \\
where \quad & u_{m,i}^{(0)} = u_m, \quad v_{n,i}^{(0)} = v_n,
\end{aligned}
\tag{1}
$$

where l denotes the number of layers and $\mathcal{N}_i(\cdot)$ maps one vertex to its neighborhood in G_i. It should be noted that the second subscript i in $u_{m,i}$ and $v_{n,i}$ means that the embeddings are under initiator view.

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

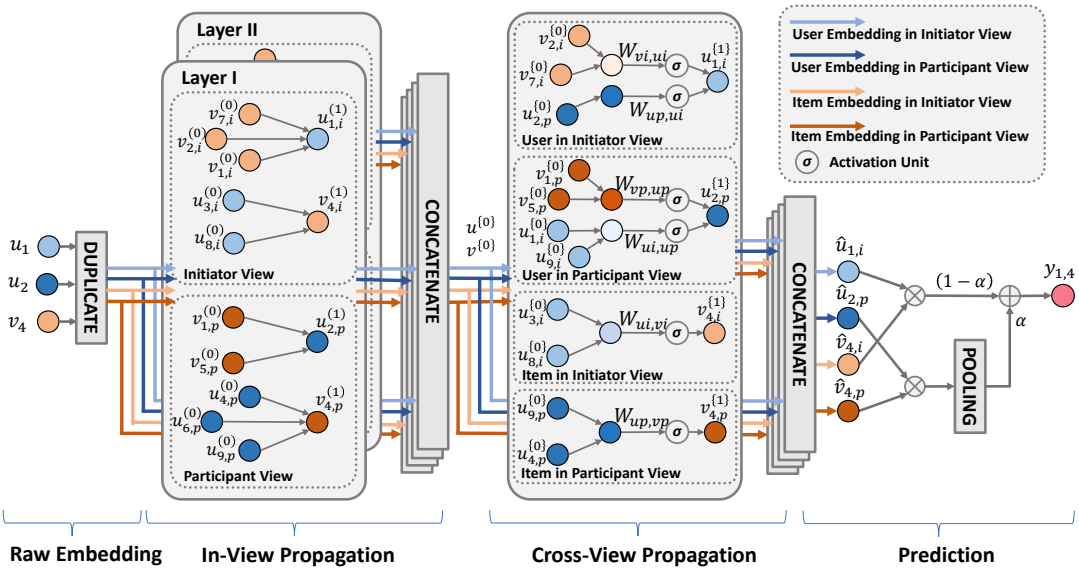

Fig. 2: The architecture of our proposed GBGCN model which is formed by cascading raw embedding layer, in-view propagation, cross-view propagation, and prediction. Here we take the prediction of item 4 and user 1 with only one friend user 2 as an example; as an initiator, user 1 has interacted with item 1, 2, 7; as a participant, user 2 has interacted with item 1, 5; item 4 has been interacted with user 3, 8 as initiators and with user 4, 6, 9 as participants. Some details, such as bias in the cross-view propagation part, are ignored due to space limitations. (Best view in color)

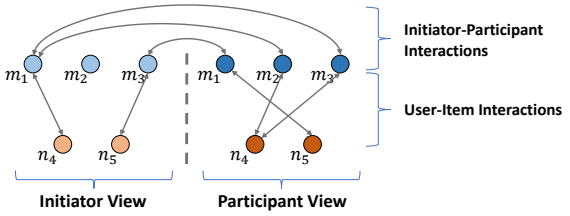

Fig. 3: The illustration of the heterogeneous graphs with two views and multiplex interactions to describe group buying.

It is almost consistent that the process in participant is fomulated as follows:

$$
\begin{aligned}
u_{m,p}^{(l)} &= \frac{1}{|\mathcal{N}_p(m)|} \sum_{n' \in \mathcal{N}_p(m)} v_{n',p}^{(l-1)}, \\
v_{n,p}^{(l)} &= \frac{1}{|\mathcal{N}_p(n)|} \sum_{m' \in \mathcal{N}_p(n)} u_{m',p}^{(l-1)}, \\
where \quad & u_{m,p}^{(0)} = u_m, \quad v_{n,p}^{(0)} = v_n,
\end{aligned}
\tag{2}
$$

where $\mathcal{N}_p(\cdot)$ maps one vertex to its neighborhood in G_p and the second subscript p means that those embeddings are under participant view.

After propagating with L layers, we generate multiple embeddings with information from different order neighborhood. To leverage them effectively and avoid information loss, it is intuitive that concatenate all levels' embeddings is a suitable manner:

$$
\begin{aligned}
u_{m,i}^{\{0\}} &= u_m || u_{m,i}^{(1)} || \cdots || u_{m,i}^{(L)}, \\
v_{n,i}^{\{0\}} &= v_n || v_{n,i}^{(1)} || \cdots || v_{n,i}^{(L)}, \\
u_{m,p}^{\{0\}} &= u_m || u_{m,p}^{(1)} || \cdots || u_{m,p}^{(L)}, \\
v_{n,p}^{\{0\}} &= v_n || v_{n,p}^{(1)} || \cdots || v_{n,p}^{(L)}.
\end{aligned}
\tag{3}
$$

In the in-view propagation layers, only one set of raw embeddings participate in the propagation process of both initiator view and participant view rather than two sets. We argue that, on the one hand, the design avoids inequality of model capacity; on the other hand, it requires the raw embedding to learn the essential features from two kinds of user-item interactions.

3) Cross-View Propagation: After in-view propagation layers, we obtain two sets of embeddings extracting user-item interaction information for initiator and participant view separately. Our key problem further turns to model the complicated social influence, which is also significant in determining whether a launched group clinches or not.

To solve this problem, we adopt cross-view information propagation with FC layers for extracting social influence and transform embeddings to corresponding subspaces. In particular, every user node in initiator view propagates self-information to users who used to join the user's group to simulate sharing behaviors and aggerates information in participant view from those users as the simulation of response in participants' joining behaviors at the same time. To introduce more

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

preference information as a supplement, the characteristics of interacted items are also considered in this layer. However, it is unreasonable to put the social influence related embeddings and preference related embeddings together. To overcome it, we adopt FC layers to distill meaningful parts and transform the embeddings from their original subspaces to the target subspace corresponding to the target view. For item nodes, we adopt the network structure consistent with the above to ensure the correspondence in the subspaces. We summarize the above design into the following formulas:

$$u_{m,i}^{\{1\}} = \sigma\left(\left(\frac{1}{|\mathcal{N}_i(m)|}\sum_{n'\in\mathcal{N}_i(m)} v_{n',i}^{\{0\}}\right)W_{vi,ui} + b_{vi,ui}\right) + \sigma\left(\left(\frac{1}{|\mathcal{N}_s^O(m)|}\sum_{m'\in\mathcal{N}_s^O(m)} u_{m',p}^{\{0\}}\right)W_{up,ui} + b_{up,ui}\right), \tag{4}$$

$$v_{n,i}^{\{1\}} = \sigma\left(\left(\frac{1}{|\mathcal{N}_i(n)|}\sum_{m'\in\mathcal{N}_i(n)} u_{m',i}^{\{0\}}\right)W_{ui,vi} + b_{ui,vi}\right), \tag{5}$$

$$u_{m,p}^{\{1\}} = \sigma\left(\left(\frac{1}{|\mathcal{N}_p(m)|}\sum_{n'\in\mathcal{N}_p(m)} v_{n',p}^{\{0\}}\right)W_{vp,up} + b_{vp,up}\right) + \sigma\left(\left(\frac{1}{|\mathcal{N}_s^I(m)|}\sum_{m'\in\mathcal{N}_s^I(m)} u_{m',i}^{\{0\}}\right)W_{ui,up} + b_{ui,up}\right), \tag{6}$$

$$v_{n,p}^{\{1\}} = \sigma\left(\left(\frac{1}{|\mathcal{N}_p(n)|}\sum_{m'\in\mathcal{N}_p(n)} u_{m',p}^{\{0\}}\right)W_{up,vp} + b_{up,vp}\right), \tag{7}$$

where $\sigma(\cdot)$ denotes the activation function, $\mathcal{N}_s^O(\cdot)$ and $\mathcal{N}_s^I(\cdot)$ map one vertex to its outgoing and incoming neighborhood in G_s, respectively. All $W \in \mathbb{R}^{(L+1)d\times(L+1)d}$ and $b \in \mathbb{R}^{(L+1)d}$ in Equation 4-7 are the trainable transformation matrices and biases, respectively. The subscripts in W and b mean the embeddings' source subspace and the target subspace. For example, the subscripts in $W_{up,ui}$ mean that the matrix transforms embeddings from **u**ser subspace in **p**articipant view to **u**ser subspace in **i**nitiator view.

Following the manner used in Equation 3, we concatenate the input embeddings with generated embeddings in this layer to obtain final vector representation as follows:

$$\begin{aligned}\hat{u}_{m,i} &= u_{m,i}^{\{0\}}||u_{m,i}^{\{1\}}, \quad \hat{v}_{n,i} = v_{n,i}^{\{0\}}||v_{n,i}^{\{1\}},\\ \hat{u}_{m,p} &= u_{m,p}^{\{0\}}||u_{m,p}^{\{1\}}, \quad \hat{v}_{n,p} = v_{n,p}^{\{0\}}||v_{n,p}^{\{1\}}.\end{aligned} \tag{8}$$

4) Prediction: After obtaining the final embeddings of both user and item in the two views, it is the next problem that how to predict the score of launching a successful group buying and return a ranking list given a user and item candidate set. We adopt a predictive function to estimate y_{mn} which is the score of user m launching a successful group buying with item n as follows:

$$y_{mn} = (1-\alpha)\hat{u}_{m,i}^T\hat{v}_{n,i} + \alpha\left(\frac{\sum_{m'=0}^{P-1} S_{mm'}\hat{u}_{m',p}^T\hat{v}_{n,p}}{\sum_{m'=0}^{P-1} S_{mm'}}\right), \tag{9}$$

where α is a hyper-parameter that should be tuned. In this predictive function, we calculate the similarity between the potential initiator m and the target item n in initiator view, and then calculate the average of the similarity between the initiator's friends and the target item in participant view, followed by weighted sum. The first step is to estimate the unnormalized probability of launching a group buying. The second step is to estimate whether there are enough participants for a successful deal according to the initiator's friends' interests. The role coefficient α indicates the importance of initiator and participant in recommeder system. For the weighted sum, we argue that the strong wish of either initiator or participant may lead to a successful deal. We leave the exploration of more complex predictive function for future works.

In conclusion, we address the first two major challenges of our problem by constructing directed heterogeneous graphs, multi-graph convolution networks and the prediction function. The solution to the last challenge will be introduced below.

C. Model Training

1) Fine-grained Loss Function: The last major challenge of our problem is the complicated feedback revealing fused preference signals of many users. More specifically, we can observe both successful and failed group-buying behaviors. A failed behavior shows that the initiator paid for the target item and launched a group buying, but there are not enough friends to join. Such behaviors imply that the initiators like the target items but their friends are not attracted to the target items to a certain degree, or the social influence between initiators and their friends is relatively weak. For those successful behaviors, it is clear that the initiators and the participants are all interested in the target items.

Based on these prior knowledge, we propose a fine-grained loss function based on *Bayesian Personalized Ranking* (BPR) [27] to mine the information in the behaviors better. For one failed behavior $b = \langle m_i, n, M_p\rangle$ where $|M_p|$ is less than threshold[3] , the loss function is as follows:

$$\mathcal{L}_-(b) = -\ln\mu\left(y_{m_in} - y_{m_in'}\right) - \beta\sum_{m'=0}^{P-1} S_{mm'}\ln\mu\left(y_{m'n'} - y_{m'n}\right), \tag{10}$$

where $\mu(\cdot)$ denotes sigmoid function, the loss coefficient β is a hyper-paramter to control the strength of belief that user m_i's friends dislike the target item n and item n' is an unobserved item (i.e. negative sample by randomly selection). For one successful behavior $b = \langle m_i, n, M_p\rangle$, the loss function is as follows:

$$\mathcal{L}_+(b) = -\ln\mu\left(y_{m_in} - y_{m_in'}\right) - \sum_{m'\in M_p}\ln\mu\left(y_{m'n} - y_{m'n'}\right). \tag{11}$$

[3]Threshold is a kind of value that highly depends on the dataset. In other words, it is determined by the service provider and cannot be directly modeled. Here our double-pairwise loss can leverage it indirectly.

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

Considering both failed behaviors B_- and successful behaviors B_+ split from the entire behaviors B, the fine-grained loss function is:

$$\mathcal{L} = \sum_{b \in B_+} \mathcal{L}_+(b) + \sum_{b \in B_-} \mathcal{L}_-(b). \tag{12}$$

Our training goal is to minimize the fine-grained loss function by optimize the model parameters. Besides, we conduct L_2 normalization to avoid over-fitting and social regularization term proposed in [1] for better learning.

2) Mini-batch Training: We perform mini-batch training for faster training speed. For constructing a mini-batch, we first sample a batch of group-buying behaviors $b = \langle m_i, n, M_p \rangle$, and then we adopt negative sampling technique, which is widely used in recommender systems [22], [27], [28] to handle implicit, *i.e.*, binary, feedback, to randomly select unobserved items $\{n'_1, n'_2, \cdots, n'_k\}$ for each observed behaviors with a sampling ratio of k. After sampling, we obtain k quadruples $\{\langle m_i, n, M_p, n'_1 \rangle, \cdots, \langle m_i, n, M_p, n'_k \rangle\}$ for each instance in the mini-batch. With the constructed mini-batch, an optimization step is taken to minimize our loss function.

3) Pre-training: Due to the data sparsity problem, it is difficult to train both embeddings and FC layers jointly. [28] also points out that neural networks are sensitive to initialization. To better train our GBGCN, we follow [19], [28] and pre-train its raw embeddings with an extremely simplified version of GBGCN that removes all propagation layers by Adam [29]. With the normalized pre-trained embeddings as an initialization, we adopt stochastic gradient descent (SGD) to fine-tune the pre-trained model.

D. Discussion

The three key components of our GCN-based method can be divided into three parts, which correspond to the three main challenges in the group-buying recommendation for social e-commerce. For the first challenge of user's multiple-role property, we design the in-view propagation layers for capturing the role-aware user interests; for the second challenge of complicated social influence, we design the cross-view propagation to modeling the social relations; for the third challenges of complex feedback, we design the double-pair loss function.

IV. Experiments

In this section, we first collect a precious real-world group-buying dataset and conduct extensive experiments to answer the following research questions.

- RQ1: Can our proposed GBGCN achieve the best recommendation performance compared with baselines?
- RQ2: How time efficient are our GBGCN and baselines?
- RQ3: How do the multi-view design in our GBGCN affect the performance?
- RQ4: How do the hyper-parameters affect the performance?
- RQ5: What insight can the learned embeddings provide?

In the next parts, we describe the experimental settings first, followed by the answers to the four research questions above.

TABLE II: Statistics of the dataset.

#Users		190,080	
#Items		30,782	
#Social Interactions		748,233	
#Group-buying Behaviors	932,896	#Successful	721,605
		#Failed	211,291

A. Experimental Settings

1) Data Collection and Preprocessing: Since there is no public group-buying dataset available, we collect the dataset from an e-commerce website Beibei[4], the largest e-commerce platform for maternal and infant products in China. To be more specifically, we obtained group-buying logs from Beibei's production database, simply filtered out users and items with few interactions as a widely-used manner mentioned in [27], [30], [31], and performed ID remapping to protect user privary. We summarize the statistics of the dataset in Table II and publish the dataset at https://github.com/Sweetnow/group-buying-recommendation.

In order to adapt the dataset to collaborative filtering methods and social recommendation methods, group-buying behavioral records should be converted into pure user-item interactions. There are two conversion methods: (1) only retain the initiator-item interactions or (2) treat both initiator-item and participant-item interactions as pure user-item interactions. Following experiments will show which method is better. For group recommendation methods, we consider each user and those who do group buying with him/her as a group; then each successful behavior is regarded as one activity of its initiator's group. However, we keep the parts for testing consistent in the original dataset and its variants above to make performance comparison equitable. The exception is that we replace each user with the group corresponding to the user as the input of group recommendation methods when testing.

2) Evaluation Protocols: Following existing works [28], [32]–[34], the widely used *leave-one-out* evaluation protocol is applied during the evaluation phase and the dataset is divided into the training set and the testing set. From the training set, we also randomly select one record for each user as the validation set to tune hyper-parameters. For the testing and validation set, we randomly sample 999 items which have not been interacted by the user given a user as initiator, and then all methods rank one test item and these sampled items. To report the ranking performance of all methods, we adopt two ranking metrics, *Recall* and *NDCG*, widely used in recommendation [35]–[37]:

- **Recall@K**: Recall measures whether the test item is present in the top-K item ranking list (1 if the answer is yes else 0).
- **NDCG@K**: *Normalized Discounted Cumulative Gain* (NDCG) complements Recall by evaluating the location where the test item appears rather than only considering whether it appears.

It should be noted that the reported performance is the averaged value for all users.

B. Overall Performance (RQ1)

1) Baselines: We compared our GBGCN with four categories of baseline methods, including collaborative filter-

[4]https://www.beibei.com

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

TABLE III: Overall performance comparison on our collected Beibei Dataset

Category	Method	Recall@3	Recall@5	Recall@10	Recall@20	NDCG@3	NDCG@5	NDCG@10	NDCG@20
CF	**MF**(oi)	0.0762	0.1055	0.1567	0.2219	0.0590	0.0710	0.0875	0.1039
	MF	0.1086	0.1456	0.2106	0.2886	0.0847	0.0999	0.1208	0.1405
	NCF	0.1231	0.1640	0.2327	0.3142	0.0961	0.1129	0.1351	0.1556
	NGCF	0.1171	0.1556	0.2190	0.2958	0.0922	0.1080	0.1284	0.1478
S	**SocialMF**	0.1135	0.1532	0.2202	0.3013	0.0889	0.1051	0.1268	0.1472
	DiffNet	0.1249	0.1664	0.2332	0.3153	0.0981	0.1151	0.1366	0.1573
G	**AGREE**	0.1036	0.1441	0.2097	0.2806	0.0798	0.0964	0.1175	0.1355
	SIGR	0.1038	0.1405	0.2034	0.2809	0.0806	0.0956	0.1159	0.1354
GB	**GBMF**	0.1262	0.1678	0.2350	0.3141	0.0991	0.1162	0.1379	0.1578
	GBGCN	**0.1341**	**0.1756**	**0.2444**	**0.3237**	**0.1064**	**0.1234**	**0.1456**	**0.1656**
	Improvement	6.29%	4.67%	4.04%	2.69%	7.36%	6.26%	5.64%	4.97%

ing methods, social recommendation methods, group recommendation methods and one group-buying recommendation method.

Collaborative filtering methods refer to those models that can only utilize user-item interaction data. The compared collaborative filtering methods are introduced as follows:

- **MF** [38]. *Matrix Factorization* (MF) is a widely used collaborative filtering model which exploits user implicit feedback to extract user preference and item feature. It adopts BPR as the loss function and samples a number of negative items with a certain ratio. Both positive and negative items are fed into the model for training in a certain proportion. We tune the learning rate and regularizer and then report the performance of the best model on the validation set.
- **NCF** [28]. *Neural network-based Collaborative Filtering* (NCF) is the state-of-the-art neural solution for recommendation tasks with implicit feedback. It ensembles Generalized Matrix Factorization (GMF) and Multi-Layer Perceptron (MLP) to model non-linearities in user-item interactions. We tune its hyper-parameters according to the related settings in the paper.
- **NGCF** [25]. *Neural Graph Collaborative Filtering* (NGCF) is a recent recommendation framework based on graph neural networks. It uses embedding propagation on the user-item bipartite graph to model high-order connectivity and extract collaborative signals. We optimize this model and tune it following the paper.

The compared social recommendation methods are as follows:

- **SocialMF** [1]. SocialMF is an MF-based model for recommendation with social networks, which force the preference of a user to be closer to the average preference of the user's social relations. BPR is also adopted as the loss function. We tune its learning rate and regularizer similarly with MF.
- **DiffNet** [11]. Diffnet is a state-of-the-art social recommendation model based on GCN. It leverages social relations and stimulates the recursive social influence propagation process on social networks for better embedding modeling. We tune the diffusion depth, learning rate, and coefficient of regularization term before reporting the best performance.

The compared group recommendation methods are as follows:

- **AGREE** [19]. *Attentive Group Recommendation* (AGREE) is based on NCF and the attention mechanism. It adopts an attention mechanism to obtain group representation and learns the interaction function between group and item by neural networks. We follow the pre-training pipeline in the paper to optimize the model on the variant of Beibei dataset for group recommendation.
- **SIGR** [21]. *Social Influence-based Group Recommender* (SIGR) is composed of a bipartite graph embedding model that leverages user-item graph and group-item graph and an attention mechanism used for learning user social influence in different groups. We follow the paper and try our best to tune it.

The compared group-buying recommendation method is as follows:

- **GBMF**. *Group-Buying Matrix Factorization* (GBMF) is a reasonable but intuitive solution for our problem compared with GBGCN. The baseline uses dot-based similarity function to calculate user's interest for the given item, computes the average score of the potential initiator's friends and then weighted sum is applied to estimate the final score and balance initiator and participants interest. BPR is adopted as the loss function. We tune the baseline like GBGCN.

We implement the baseline methods and our GBGCN model in PyTorch[5] with Deep Graph Library (DGL)[6]. We set the embedding size of all compared methods to be 32 to balance memory efficiency and model capacity, which is a common setting in existing works on graph representation learning based recommendation [22], [23]. Our main experiments also demonstrate that such embedding size has enough representation ability.

2) Parameter Settings: We tune all the hyper-parameters of our method on the validation set. We initialize parameters with the widely used Xavier initialization method [39]. When training, we set the negative sampling ratio as 1:1 following [40] and construct mini-batches with a size of 4096 to balance training efficiency and memory usage. To optimize the GBGCN model better, we follow the strategy described in [19], [28], employ Adam in the pre-training stage, and use the vanilla SGD in fine-tuning stage to avoid the problem of loss of momentum information. Besides, we search the Adam's learning rate within {1e-2, 1e-3, 1e-4, 1e-5} and search the SGD's in {10, 3, 1, 0.3}. In addition, we search the coefficients of regularization term in {1e-1, 3e-2, 1e-2, 3e-3, 1e-3, 3e-4, 1e-4, 3e-5, 1e-5}. α, the coefficient to balance the importance of initiator and participants, is searched at intervals of 0.1 within the range of 0.1 to 0.9. We also tune the loss coefficient β in {0.01, 0.02, 0.05, 0.1, 0.2,0.5}. The number of layers L is set to 2. For each model, we train it for 500 epochs, and we

[5]https://pytorch.org
[6]https://www.dgl.ai

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

save the model that has the best performance on the validation set for evaluation.

Firstly, we compare the overall performance of GBGCN and baseline methods on the top-K group-buying recommendation task. In order to fully demonstrate the ability of the methods in different segments, K is set to $\{3, 5, 10, 20\}$. It is another reason for relatively small K that the length of ranking list is 1000 according to the evaluation protocols described above. We tune all the hyper-parameters for each method carefully and report their best preformance on the real-world dataset in Table III. In the category column in Table III, CF, S, G, GB are the abbreviations of collaborative filtering, social recommendation, group recommendation and group-buying recommendation, respectively. Because two conversion methods are proposed above, we use *oi* to mark the model in Table III that adopts the first conversion method, which retains initiator-item interactions only and do not mark models using the second one to keep the table concise. From these results, we obtain several observations as follows:

- **Our GBGCN method outperforms all the baseline methods on the real-world dataset significantly.** Benefiting from the directed heterogeneous graphs and multi-view information propagation, GBGCN is capable of modeling the multi-role feature and complex interactions in group buying. The fine-grained loss function helps GBGCN fully mine information from the dataset. Compared to all baseline methods, GBGCN obtains the best performance in terms of Recall@K and NDCG@K and outperforms the best baseline by 2.69%-6.29% in Recall and 4.97%-7.36% in NDCG. The p-value of significance tests is less than 0.05, which demonstrates the improvement is stable and significant. It is noted that GBGCN performs better with smaller K, which is more meaningful for application scenarios. Such improvements on various metrics demonstrate GBGCN's effectiveness.
- **The performance of MF with only initiator-item interactions is much lower than that with both initiator-item and participant-item interactions.** As we can observe, MF with extra data achieves better prediction performance. This phenomenon indicates that collaborative filtering and social recommendation models should be fed as much data as possible to model user preference and item feature precisely. The same phenomenon also appears in experiments of other models so that we do not show their performance.
- **As collaborative filtering methods, NCF generally achieves better performance than NGCF.** According to [25], NGCF outperforms NCF on most item recommendation tasks. However, it is multiplex for the relations between user and item in our problem. NGCF is designed for exploring collaborative filtering signals in the user-item bipartite graph and might fails to fully model the multiplex relations while NCF is capable of describing them by MLP.
- **As group recommendation methods, the performance of AGREE and SIGR are both poor, even worse than MF.** In group recommendation problem, the recommender is required to recommend items for a fixed group, and users have no role difference. To measure user's influence in a group, the two methods both introduce attention mechanism to aggregate user embeddings as group embeddings while attention mechanisms do not work due to the data sparsity problem of the real-world dataset. More importantly, users have dynamic roles in group buying, which is hard for group recommendation methods to handle. The selection of loss function may be another reason. AGREE adopts a regression-based pairwise loss, which forces the margin of the prediction of observed interaction and unobserved one to be 1. SIGR adopts *logloss* to classify positive and negative items. Maybe it is relatively weaker for the two loss funcions to obtain information from negative sampling than BPR.
- **The performance of GBMF is the best among all the baseline methods in terms of most metrics.** As an intuitive but strictly suitable baseline for our problem, the performance of GBMF indicates that targeted design must be made for new problems and it is unsuitable to adapt methods which are built for different problems. Meanwhile, the performance comparison between GBGCN and GBMF indicates the benefits of the directed heterogeneous graphs, multi-view information propagation and the fine-grained loss function proposed in GBGCN.

To summarize, the extensive comparisons on the real-world dataset verify that our GBGCN method is effective and suitable for our problem.

TABLE IV: Time efficiency comparison.

Methods	**Training Time** (sec/epoch)	**Testing Time** (sec/epoch)
MF(oi)	2.99	4.74
MF	3.65	4.75
NCF	3.83	4.47
NGCF	5.68	4.87
SocialMF	5.27	4.83
DiffNet	4.77	4.55
AGREE	17.25	15.25
SIGR	58.29	8.56
GBMF	31.68	54.34
GBGCN	56.28	88.36

C. Time Efficiency Comparison

In this section, we conduct further experiments to study how time efficient our GBGCN and baselines are. All these experiments are run on the same single Linux server with one NVIDIA TITAN Xp and DGL is adopted to speed up all GCN-based models. Table IV reports the time efficiency results (measured in wall-clock time).

From the results, we can observe that collaborative filtering and social recommendation baselines all achieve good training and testing time efficiency due to relatively simple input data format. However, group recommendation and group-buying recommendation models spend more time on training and testing. The reason is that these models all consider user's friends or group members in forwarding phase but the number of each user's friends or each group's members is different. The inconsistent number of elements greatly impairs the parallelism of these models, resulting in a slower running speed.

D. Impact of Multi-view Design (RQ3)

In our GBGCN, we generate two sets of embeddings for initiator view and participant view separately by in-view propagation and cross-view propagation to model user's and item's different characteristics in different views and overcome the first major challenge mentioned in the introduction. There are

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

two intuitive questions about whether the multi-view design helps in GBGCN, and how does it affect the performance?

To answer them, we conduct ablation experiments on three degenerative models of GBGCN. Without reducing the capacity of the model, we simply replace the two sets of embeddings at the output of each propagation layer with their average pooling. To be specific, if we remove the user roles in a 2-layer GBGCN, $u_{m,i}^{(l)}$ and $u_{m,p}^{(l)}$ will be replaced by $\frac{u_{m,i}^{(l)}+u_{m,p}^{(l)}}{2}$ for $l = 1, 2$ after each propagation, and the treatment of $u_{m,i}^{\{1\}}$ and $u_{m,p}^{\{1\}}$ is also consistent. The ablation manners for removing only item roles and removing both user and item roles are similar. We adopt the same evaluation methods with the above experiments and report the performance comparison in Table V. We can observe that the recommendation performance becomes poor after removing either item roles or user roles. Meanwhile, it is intuitive that removing both user and item roles makes the performance worse than removing only one part. To conclude, the experimental results of the ablation study demonstrate the necessity of the multi-view design.

E. Hyper-parameter Study (RQ4)

For our GBGCN method, there are two significant hyper-parameters that are closely related to performance. In this section, we study how much these hyper-parameters affect prediction performance and find out the best hyper-parameters. We investigate the role coefficient α mentioned in Equation 9, followed by the loss coefficient β in Equation 10.

1) Role Coefficient: The role coefficient α, which is used in the prediction function, balances the influence of initiator and participant candidates for a successful group buying. A larger α implies that users on e-commerce websites tend to consider their friends' interest when launching deals. On the contrary, a smaller α means that users prioritize their own needs, and it is relatively easy to find enough participants to some extent.

To study how the coefficient affects the prediction performance, we present the Recall@10 and NDCG@10 performance in Figure 4. We can observe that a biased value of α worsen the prediction performance, and simply averaging is also not the best solution. Our method GBGCN achieves its best performance on the real-world dataset with $\alpha = 0.6$.

2) Loss Coefficient: We make use of the loss coefficient β to adjust the strength of belief for strong-negative feedback, failed group-buying behaviors. Actually, the reasons why a group failed are complicated. In addition to the weak social influence and friends' interest, there are also possible reasons such as the user's wrong operations, not sharing after purchase, etc. Therefore, the strong-negative feedback is not completely reliable.

To study what value of β is most beneficial to GBGCN, we search β in {0, 0.01, 0.02, 0.05, 0.1, 0.2,0.5} and present the Recall@10 and NDCG@10 performance in Figure 4 except $\beta = 0$. We can observe that appropriate value of β really helps GBGCN on the dataset. The best β value is 0.05. When $\beta = 0$, the double-pairwise loss degenerates to the standard pairwise loss and Recall@10 is 0.2401 and NDCG@10 is 0.1421, which is worse than our method. This result indicates that our double-pairwise loss is better than the standard pairwise loss.

F. Embedding Analysis (RQ5)

In GBGCN, we introduce the multi-view design to model multiple roles of users in group buying. We conduct ablation studies about this design and then verify its necessity in Section IV-D. We argue that the two sets of embeddings generated by multi-graph convolution networks contain specific information on the corresponding view. To verify it and obtain some insight from the embeddings, we (1) compute the probability density function (PDF) for cosine similarity of in-view propagation generated embeddings and cross-view propagation generated embeddings separately between initiator view and participant view for each entry and (2) visualize the final embeddings of randomly sampled 1000 items and 1000 users by a widely used method t-SNE [41].

The PDF curves are shown in Figure 5. We can observe that for in-view propagation generated embeddings, the embeddings of items are basically consistent in different views, and the embeddings of users show a certain difference. This phenomenon indicates that item characteristics reflected in initiator-item interactions and participant-item interactions are close, while users' interests start to diverge. For cross-view propagation generated embeddings, embeddings in the two views diverge obviously, which means that cross-view propagation and FC layers succeed in capture view-specific information.

The visualization is shown in Figure 6 also supports the insight above. Ignoring the clusters in the lower right corner, which might be users with few interactions in the corresponding view, the nodes in the figure can be seen as two parts, embeddings in initiator view on the right side and embeddings in participant view on the left side. The visualization demonstrates the ability of GBGCN to model the difference between the two views at the embedding level.

In conclusion, extensive experiments on the real-world dataset verify the efficacy of our proposed GBGCN. Further studies demonstrate that GBGCN succeeds in generating proper representation for both initiator view and participant view by the multi-view design.

V. Related Work

A. Group Buying and Group Recommendation

Group buying [42] is generally defined as that a group of people purchase the same product together. Actually, there are two kinds of forms of group buying in today's online e-commerce services. At the earlier group-buying [43]–[45], platforms help to organize a team consisting hundreds or even thousands users—who do not know each other in most times—to purchase items together from sellers. Recently, a new kind of group-buying has become very popular, such as Pinduoduo in China. Specifically, users can serve as the group founder, and invite their friends to join the group. As a result, the users in group of the new kind of group buying are always connected in social networks, which make it completely different from the previous one.

Another close-related topic is group recommendation [46] is to recommend one item to a group of users. All users in the group interact with *one* item together, and groups are always pre-defined and constant. Typical applications of group recommendation consist of travel recommendation, restaurant

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

TABLE V: Impact of Multi-view Design

Methods	Recall@10	Improve.	Recall@20	Improve.	NDCG@10	Improve.	NDCG@20	Improve.
GBGCN	0.2444	-	0.3237	-	0.1456	-	0.1656	-
Without Item Roles	0.2422	-0.93%	0.3226	-0.36%	0.1439	-1.17%	0.1642	-0.87%
Without User Roles	0.2430	-0.59%	0.3218	-0.59%	0.1447	-0.66%	0.1646	-0.65%
Without Item and User Roles	0.2408	-1.49%	0.3189	-1.49%	0.1439	-1.18%	0.1636	-1.21%

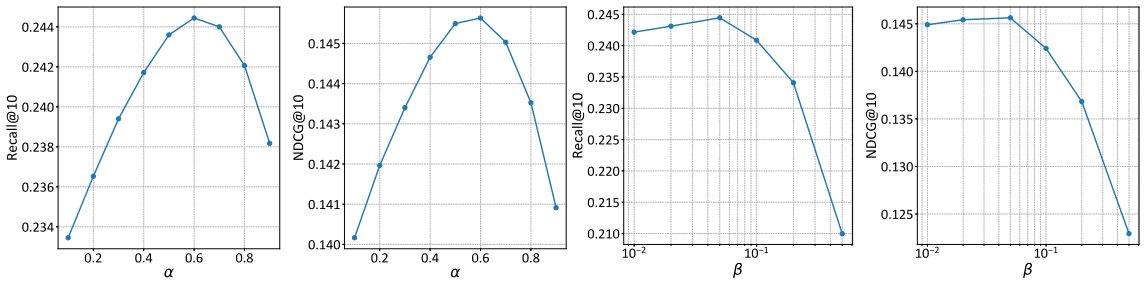

Fig. 4: Testing performance with different role coefficient α and different loss coefficient β.

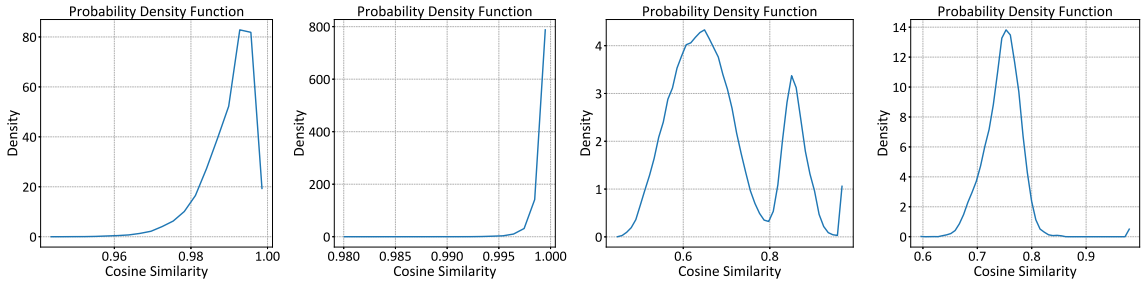

Fig. 5: Probability density function (PDF) of cosine similarity between embeddings in initiator view and participant view. (a) shows PDF of cosine similarity of the two sets of user's in-view propagation generated embeddings; (b) shows PDF of cosine similarity of the two sets of item's in-view propagation generated embeddings; (c) shows PDF of cosine similarity of the two sets of user's cross-view propagation generated embeddings; (d) shows PDF of cosine similarity of the two sets of item's cross-view propagation generated embeddings.

recommendation, etc. Early approaches [13]–[18] model the interest of each user belonging to one group and then design some strategies, such as average, least misery, maximum satisfaction, etc., to fuse the interests together. Cao *et al.* [19] propose to utilize neural collaborative filtering with attention mechanisms to model the user interest and group interest at the same time, which is further extended by the same authors with combining social network [20]. Yin *et al.* [21] consider the social relation in pre-defined groups, and introduce social-aware constraint into the preference learning in groups. In our problem of group-buying recommendation for social e-commerce, the group is formed dynamically from the social network, and there is no pre-defined group at all, which is very different compared with group recommendation tasks.

B. Social Influence and Social Recommendation

Social influence [47], [48] studies how friends affect users' behaviors or decisions and has wide applications in various areas [21], [49]–[51]. For recommender systems, social influence is introduced by the *social-trust* assumption that users trusts their friends' decisions and will have similar preferences with friends. The recommendation models that leverage social influence with social-network data is defined as social recommendation [52]. Based on the social-trust assumption, early works used regularization techniques to limit the distance of embedding vectors in latent space between a user and her friends [1]–[5]. Some works assume a user's decision should be a combination of her own characteristics and her trusted friends' recommendations [6]–[8]. Some other works [9], [10], [53] consider the recommendation and social relation prediction as two tasks and approach the problem with multi-task learning. Yu *et al.* [54] proposed to utilize generative adversarial networks to identifying reliable friends from observed and unobserved social networks. Wu *et al.* [11] proposed to use embedding propagation on the social network to capture the social-trust effect. Fan *et al.* [12] further combine embedding propagation with attention networks to model friends' various influence. Xu *et al.* [55] considered the recommendation scenario in social e-commerce when users serve as selling agents and leverage multi-hop graph convolutional networks to learn the social influence.

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

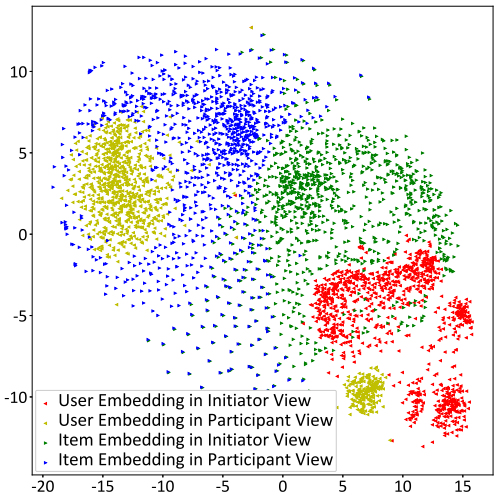

Fig. 6: The visualization of user and item embedding in initiator and participant view by t-SNE. (Best view in color)

Different from social recommendation, the aim of our problem is a group of users, rather than a single user. The only same point is that both two takes take use of social networks to help learn user preferences and users' social influence.

C. Graph Convolutional Networks for Recommendation

Graph convolutional network [56] has become a new state-of-the-art approach of graph representation learning. The core idea of GCN is to use embedding propagation to extract high-order structural knowledge and node attributes at the same time. [57] first applied GCN to collaborative filtering recommendation, which can be considered as extension to SVD++ [38]. Ying *et al.* [24] further extended GCMC to implicit recommender systems with a neighbor-sampling technique to reducing computation cost. Wang *et al.* [22] further adjusted the design of the propagation and prediction layer, which is a state-of-the-art GCN based collaborative filtering method. GCN also achieved big success on content-based recommendation, such as session-based recommendation [58], knowledge-based recommendation [23], price-aware recommendation [35], multi-behavior recommendation [59], bundle recommendation [60], etc. However, these existing GCN-based recommendation models are designed for collaborative filtering tasks or specific tasks with unique forms of data. Thus, they cannot be applied to our problem of group-buying recommendation.

VI. Conclusion and Future Work

In this work, we take the pioneer step to approach the problem of group-buying recommendation for social e-commerce. We propose to construct directed hetegenous graphs to represent group-buying behaviors, including launching and joining groups, and social network data. We develop a graph convolutional network method named GBGCN, which combines inner-view and cross-view embedding propagation with a double-pairwise loss function. We collect a real-world group-buying dataset and conduct extensive experiments. The results demonstrate that our GBGCN can achieve the best recommendation performance compared with baselines with significant improvements. As for future work, we plan to study the data sparsity issue and conduct A/B test to further evaluate our GBGCN model.

Acknowledgement

This work was supported in part by The National Key Research and Development Program of China under grant 2018YFB1800804, the National Nature Science Foundation of China under U1936217, 61971267, 61972223, 61941117, 61861136003, Beijing Natural Science Foundation under L182038, Beijing National Research Center for Information Science and Technology under 20031887521, and research fund of Tsinghua University - Tencent Joint Laboratory for Internet Innovation Technology.

References

[1] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proceedings of the fourth ACM conference on Recommender systems (RecSys)*, 2010, pp. 135–142.

[2] L. Guo, J. Ma, H.-R. Jiang, Z.-M. Chen, and C.-M. Xing, "Social trust aware item recommendation for implicit feedback," *Journal of Computer Science and Technology (JCST)*, vol. 30, no. 5, pp. 1039–1053, 2015.

[3] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *Proceedings of ACM International Conference on Web Search and Data Mining (WSDM)*, 2011, pp. 287–296.

[4] L. Du, X. Li, and Y.-D. Shen, "User graph regularized pairwise matrix factorization for item recommendation," in *International Conference on Advanced Data Mining and Applications (SDMA)*, 2011, pp. 372–385.

[5] X. Wang, X. He, L. Nie, and T.-S. Chua, "Item silk road: Recommending items from information domains to social users," in *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2017.

[6] H. Ma, I. King, and M. R. Lyu, "Learning to recommend with social trust ensemble," in *International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2009, pp. 203–210.

[7] J. Tang, H. Gao, and H. Liu, "mtrust: discerning multi-faceted trust in a connected world," in *International Conference on Web Search and Data Mining (WSDM)*, 2012, pp. 93–102.

[8] A. Davoudi, "Effects of user interactions on online social recommender systems," in *International Conference on Data Engineering (ICDE)*. IEEE, 2017, pp. 1444–1448.

[9] H. Ma, H. Yang, M. R. Lyu, and I. King, "Sorec: social recommendation using probabilistic matrix factorization," in *Proceedings of the 17th ACM conference on Information and knowledge management (CIKM)*, 2008, pp. 931–940.

[10] J. Tang, X. Hu, H. Gao, and H. Liu, "Exploiting local and global social context for recommendation." in *International Joint Conferences on Artificial Intelligence (IJCAI)*, vol. 13, 2013, pp. 2712–2718.

[11] L. Wu, P. Sun, Y. Fu, R. Hong, X. Wang, and M. Wang, "A neural influence diffusion model for social recommendation," in *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2019.

[12] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, "Graph neural networks for social recommendation," in *The World Wide Web Conference (WWW)*, 2019, pp. 417–426.

[13] L. Baltrunas, T. Makcinskas, and F. Ricci, "Group recommendations with rank aggregation and collaborative filtering," in *Proceedings of the fourth ACM conference on Recommender systems (RecSys)*, 2010, pp. 119–126.

[14] X. Liu, Y. Tian, M. Ye, and W.-C. Lee, "Exploring personal impact for group recommendation," in *Proceedings of the 21st ACM international conference on Information and knowledge management (CIKM)*, 2012, pp. 674–683.

[15] E. Quintarelli, E. Rabosio, and L. Tanca, "Recommending new items to ephemeral groups using contextual user influence," in *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys)*, 2016, pp. 285–292.

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.

[16] Q. Yuan, G. Cong, and C.-Y. Lin, "Com: a generative model for group recommendation," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, 2014, pp. 163–172.

[17] I. Christensen, S. Schiaffino, and M. Armentano, "Social group recommendation in the tourism domain," *Journal of intelligent information systems (JIIS)*, vol. 47, no. 2, pp. 209–231, 2016.

[18] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and W. Cao, "Deep modeling of group preferences for group-based recommendation," in *Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI)*, 2014.

[19] D. Cao, X. He, L. Miao, Y. An, C. Yang, and R. Hong, "Attentive group recommendation," in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR)*, 2018, pp. 645–654.

[20] D. Cao, X. He, L. Miao, G. Xiao, H. Chen, and J. Xu, "Social-enhanced attentive group recommendation," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2019.

[21] H. Yin, Q. Wang, K. Zheng, Z. Li, J. Yang, and X. Zhou, "Social influence-based group representation learning for group recommendation," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 566–577.

[22] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2019.

[23] X. Wang, X. He, Y. Cao, M. Liu, and T.-S. Chua, "Kgat: Knowledge graph attention network for recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. ACM, 2019.

[24] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*. ACM, 2018, pp. 974–983.

[25] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR)*, 2019, pp. 165–174.

[26] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "Lightgcn: Simplifying and powering graph convolution network for recommendation," in *Proceedings of the International ACM Conference on Research and Development in Information Retrieval (SIGIR)*, 2020.

[27] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," in *The Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009, pp. 452–461.

[28] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 27th International Conference on World Wide Web (WWW)*, 2017, pp. 173–182.

[29] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[30] X. He, H. Zhang, M.-Y. Kan, and T.-S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," in *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2016, pp. 549–558.

[31] X. He, X. Du, X. Wang, F. Tian, J. Tang, and T.-S. Chua, "Outer product-based neural collaborative filtering," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 2227–2233.

[32] J. He, J. Qi, and K. Ramamohanarao, "A joint context-aware embedding for trip recommendations," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 292–303.

[33] T. Chen, H. Yin, Q. V. H. Nguyen, W.-C. Peng, X. Li, and X. Zhou, "Sequence-aware factorization machines for temporal predictive analytics," in *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE, 2020, pp. 1405–1416.

[34] C. Gao, X. He, D. Gan, X. Chen, F. Feng, Y. Li, T.-S. Chua, and D. Jin, "Neural multi-task recommendation from multi-behavior data," in *International Conference on Data Engineering (ICDE)*, 2019.

[35] Y. Zheng, C. Gao, X. He, Y. Li, and D. Jin, "Price-aware recommendation with graph convolutional networks," in *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE, 2020, pp. 133–144.

[36] S. Madisetty, "Event recommendation using social media," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 2106–2110.

[37] F. Khawar and N. L. Zhang, "Modeling multidimensional user preferences for collaborative filtering," in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 1618–1621.

[38] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2008, pp. 426–434.

[39] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, Y. W. Teh and M. Titterington, Eds., vol. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256.

[40] Y. Zheng, C. Gao, X. He, Y. Li, and D. Jin, "Price-aware recommendation with graph convolutional networks," *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pp. 133–144, 2020.

[41] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research (JMLR)*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[42] K. S. Anand and R. Aron, "Group buying on the web: A comparison of price-discovery mechanisms," *Management Science*, vol. 49, no. 11, pp. 1546–1562, 2003.

[43] L. Bai, M. Hu, Y. Ma, and M. Liu, "A hybrid two-phase recommendation for group-buying e-commerce applications," *Applied Sciences*, vol. 9, no. 15, p. 3141, 2019.

[44] Y.-C. Hu, "A multicriteria collaborative filtering approach using the indifference relation and its application to initiator recommendation for group-buying," *Applied Artificial Intelligence*, vol. 28, no. 10, pp. 992–1008, 2014.

[45] Y.-F. Chen and H.-F. Lu, "We-commerce: Exploring factors influencing online group-buying intention in t aiwan from a conformity perspective," *Asian Journal of Social Psychology*, vol. 18, no. 1, pp. 62–75, 2015.

[46] S. Amer-Yahia, S. B. Roy, A. Chawlat, G. Das, and C. Yu, "Group recommendation: Semantics and efficiency," *Proceedings of the VLDB Endowment (PVLDB)*, vol. 2, no. 1, pp. 754–765, 2009.

[47] R. B. Cialdini and N. J. Goldstein, "Social influence: Compliance and conformity," *Annu. Rev. Psychol.*, vol. 55, pp. 591–621, 2004.

[48] M. O. Jackson, *Social and economic networks*. Princeton university press, 2010.

[49] S. Feng, G. Cong, A. Khan, X. Li, Y. Liu, and Y. M. Chee, "Inf2vec: Latent representation model for social influence embedding," in *2018 IEEE 34th International Conference on Data Engineering (ICDE)*. IEEE, 2018, pp. 941–952.

[50] X. Wang, Y. Zhang, W. Zhang, and X. Lin, "Distance-aware influence maximization in geo-social network." in *ICDE*, 2016, pp. 1–12.

[51] J. Kim, S.-K. Kim, and H. Yu, "Scalable and parallelizable processing of influence maximization for large-scale social networks?" in *2013 IEEE 29th international conference on data engineering (ICDE)*. IEEE, 2013, pp. 266–277.

[52] J. Tang, X. Hu, and H. Liu, "Social recommendation: a review," *Social Network Analysis and Mining (SNAM)*, vol. 3, no. 4, pp. 1113–1133, 2013.

[53] T.-H. Lin, C. Gao, and Y. Li, "Cross: Cross-platform recommendation for social e-commerce," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 515–524.

[54] J. Yu, M. Gao, H. Yin, J. Li, C. Gao, and Q. Wang, "Generating reliable friends via adversarial training to improve social recommendation," in *International Conference on Data Mining (ICDM)*, 2019.

[55] F. Xu, J. Lian, Z. Han, Y. Li, Y. Xu, and X. Xie, "Relation-aware graph convolutional networks for agent-initiated social e-commerce recommendation," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, 2019, pp. 529–538.

[56] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[57] R. v. d. Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," *arXiv preprint arXiv:1706.02263*, 2017.

[58] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, "Session-based recommendation with graph neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 33, 2019, pp. 346–353.

[59] B. Jin, C. Gao, X. He, D. Jin, and Y. Li, "Multi-behavior recommendation with graph convolutional networks," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 659–668.

[60] J. Chang, C. Gao, X. He, D. Jin, and Y. Li, "Bundle recommendation with graph convolutional networks," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 1673–1676.

Authorized licensed use limited to: Tsinghua University. Downloaded on July 31,2023 at 12:56:02 UTC from IEEE Xplore. Restrictions apply.