GRUIFI: A Group Recommendation Model Covering User Importance and Feature Interaction

Jingwei Zhang[1], Jing Cheng[1], Ya Zhou[1], Qing Yang[2]

[1] Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, China
[2] Guangxi Key Laboratory of Automatic Dectection Technology and Instrument, Guilin University of Electronic Technology, China
gtzjw@hotmail.com, 2605496187@qq.com, 1435691210@qq.com, gtyqing@hotmail.com

Abstract

Group recommendation derives from a phenomenon that a group with similar interests have formed various communities, which creates the requirements that a group of users in one community want to share personalized services. Different from traditional recommendations that focus on individuals, group recommendation needs to consider the differences in preference of group members. How to build a proper model for group members to aggregate different preferences is still a challenging problem: (1) the influence of group members is quite different; (2) a user decision is directly or indirectly influenced by other members in the same group. This paper proposed a **G**roup **R**ecommendation model covering **U**ser **I**mportance and automatic **F**eature **I**nteraction (GRUIFI), which can model interaction data of group member and learn group potential preference representation. Our model exploits an attention mechanism to obtain the weights of group members that represent user importance, and those dynamic user weights are integrated to learn a group representation. Then we design a neural network that combines the multi-head attention to automatically learn fine-grained interactions between groups and items, and further capture the interdependency between group members. Finally, **t**he experiments on the two real-world datasets show that GRUIFI performs significantly better than baseline methods.

Keywords: Group recommendation, Preference aggregation, High-order feature interactions, Group representation

1 Introduction

With the appearance and popularization of the intelligent applications, although getting all kinds of information is becoming convenient, it is still difficult for users to accurately obtain information resources that meet their own needs [1-2]. In order to provide each user with personalized information services and effectively solve the problem of information overload, recommendation system was proposed.

For the research of recommendation systems, various types of recommendation systems have emerged, such as mobile recommendation systems, context-aware recommendation systems, and social network recommendations [3]. There are many scenarios in the real world for group recommendation, which leads to the development of social networks and online communities [4]. Users have formed social network communities based on similar interests, where they participate in group activities and share their life. At this time, traditional personalized recommendations are often designed to simulate the activities of individual user, and cannot meet the recommendation needs when users participate in activities in the form of groups. As a result, a novel information service pattern called group recommendation system has received increasing attention [5].

In recent years, group recommendation systems have been widely used in many fields, such as tourism [6] and social activities [7-8]. In general, the group can be divided into two types: persistent groups and occasional groups [9]. A persistent group that is a group of fixed group members who not only have long-term similar interests, but also have a large amount of group-item interaction data [10]. An occasional group is the first time that users form a group at a specific time or in a specific environment, and group members have no clear similar interests [11]. Online applications make it easy to build various user groups. Group recommendation systems allow these user groups to share and obtain valuable information by discussing certain topics, and find users with similar interests to expand their social circle. Owing to the abundance of these information resources, it is helpful to provide accurate recommendation services to groups [12].

As the recommendation object is expanded from a user to a group, personalized recommendation systems are no longer applicable, which brings many challenges for the group recommendation systems. The

*Corresponding Author: Qing Yang; E-mail: gtyqing@hotmail.com
DOI: 10.53106/160792642021092205017

decision-making process of a group should take into full account not only the difference of preferences among group members, but also the interaction between group members, so that the recommendation results can meet the needs of all group members. Therefore, how to explore the preference aggregation strategy by learning from interaction data is a key issue in the research of group recommendation system. Most of the existing group preference aggregation strategies use predefined strategies to aggregate preferences of group members, such as average strategy [13], least misery strategy [14], and maximum satisfaction strategy [15]. However, this way of these static strategies defines user importance is not comprehensive enough. At the same time, these methods do not fully consider the influence of personal characteristics of group members on group decision-making. The key point is that these methods do not make good use of the relationships among group members, and individual user preferences may be highly influenced by other group members. Therefore, these predefined strategies are too simple to capture the complex process of group decision-making, resulting in suboptimal performance of group recommendation.

In this paper, we focus on designing a representation model to learn the importance of group members, and then learning deeply interaction between group members. To obtain a group representation, we aggregate the representation of group members by an embedding layer in a learnable way. Specifically, we introduce a latent variable to represent the importance of group members in group decision-making, and then adopt the attention network to learn the weight of group members, which is capable of assigning different weights for group members. Besides, to model the interactions among group members, we introduce a new neural architecture for group recommendation. Specifically, we consider user-item interaction data, and design the multi-head attention to automatically learn high-order feature interactions of from group-item interaction data. Subsequently, it further obtains the impact of group members interaction on recommendation. These measures provide more effective performance of group recommendation.

The main contributions of this paper are summarized as follows:

- We proposed GRUIFI, which is a deep learning architecture for group recommendation. Our model can dynamically obtain the weight of group members that represents user importance by using attention mechanism. Then the comprehensive group representation is accurately captured by modeling the representation of items, users and groups and aggregating the representation of users by an embedding layer.
- We designed a neural network that combines the multi-head attention, which as used to automatically learn high-order feature interaction and to obtain the interdependency between group members. Specifically, the usage of multi-head attention network provides an effective way to study the problem of explicitly learning high-order feature interactions for group members.
- We conducted extensive experiments on two real-world datasets, and the results show that our model outperforms the compared methods significantly.

2 Related Work

Preference aggregation strategy and preference aggregation method are the key technology for the group recommendation system. The traditional group recommendation system is divided into two stages: user preference acquisition and recommendation generation. Group recommendations can aggregate preferences before or after recommendation generation. According to the different stages of preference aggregation in the recommendation process, preference aggregation methods are divided into two categories, memory-based and model-based approaches. Among them, preference aggregation after recommendation is called memory-based approach, and preference aggregation before group recommendation is called model-based approach.

Memory-based approaches can be further subdivided into preference aggregation and score aggregation [16]. The preference aggregation firstly uses a recommendation algorithm to generate a recommendation list for each user in the group, and then generates a group recommendation list through aggregation strategies [17]. While the score aggregation firstly predicts the score of each user in the group on the candidates, and then uses the aggregation strategy to get the group's score on the candidates [13]. The two most popular strategies for score aggregation are the average (AVG) and the least misery (LM) strategies. The AVG strategy makes use of the average score of group members as the final recommendation score, so that the recommendation results are satisfied to all group members [17]. The LM strategy satisfies all group members by selecting the lowest score among all group members as the final score [13]. By contrast with the LM strategy, the MS strategy takes the largest score in the group as the group score [15]. Through the comparative analysis of these aggregation strategies, the above three aggregation methods have their respective limitations. The AVG strategy may return items that cause dissatisfaction among individual group members, while the LM strategy is difficult to grasp the overall preference of the group and is vulnerable to malicious ratings. The MS strategy is biased in using a single user to determine group preferences. Baltrunas et al. [13] pointed out that the performance of three strategies depends on group size and inner-group similarity. It is inevitable that group members have different preferences for each item.

In contrast, model-based approaches [18] assign different weights to group members in the recommendation process according to their characteristics or influence. Zhao et al. [19], according to the thought of Nash equilibrium, simulated the preferences of group members through game theory and combined with the matrix factorization to aggregate the preference of each group members. However, since the result of Nash equilibrium is a series of items, game theory may not be able to recommend a single specific item. Liu et al. [8] proposed a Personal Impact Topic (PIT) model for group recommendation, assuming that the most influential user represents the group and have a higher influence on group decision-making. However, the assumptions of this model ignore the fact that the decision of the user is representative of the group decision only if the user is an expert in the field. Yuan et al. [11] proposed a consensus model (COM) for group recommendation, which simulated the generation process of group activities through a probability model. In addition, it also assigns different weights to users according to the topic. COM believes that the same user has different weights in different groups, and the behaviors of group members are independent of each other without interference. However, it assumes that users in different groups have the same probability to follow group decisions.

Neural networks have been extensively applied in recommender systems because of high quality recommendations [20]. He et al. [21] proposed neural collaborative filtering (NCF), which can learn interaction functions from data. It pointed out that the common matrix factorization model is a special case of NCF [22]. Recently, attention mechanism [23] has been widely used in recommendation system, because it can effectively capture the non-linear and non-trivial relationship between users and items to make better recommendations. Cao et al. [10] proposed a AGREE model that exploited attention network and NCF for a group recommendation, which can learn the dynamic weight of users in the group. Moreover, it applies the attention mechanism to the aggregation of interactive items to obtain group representation for recommendation. SoAGREE [24] introduced the information of social attributes on the basis of the AGREE model. However, AGREE and SoAGREE cannot consider good performance heavily depends on the direct learning of group preference embedding from group members interaction. Based on the inspiration of existing research, our work falls into the category of model-based approaches. We consider dynamic learning aggregation strategy and modeling user interaction under the framework of deep learning. This paper proposes GRUIFI for learning the representations of users, items and groups. GRUIFI adopts attention network to model the importance of group members, learns the integration strategy of group members, and the usage of multi-head attention learns interaction function to make group decision recommendations after obtaining the group representation.

3 Methods

Generally speaking, recommending an item for the group requires two key factors: one is how to obtain the semantic representation of a group; and the other is how to model the interaction between the group and the item. In view of these two factors, our proposed GRUIFI model consists of two components, (1) group representation learning which use attention mechanism to learn user preference and user importance; and (2) interaction learning which recommends item for both groups and items with multi-head attention for high-order feature interactions of group members. We first present the notations and formulate the group recommendation problem to be solved. We then introduce the two key ingredients of our proposed model. Lastly, we discuss the optimization method.

3.1 Notations and Problem Formulation

Following the convention, we use bold capital letters (e.g., $\mathbf{X}$) and bold lowercase letters (e.g., $\mathbf{x}$) to represent matrices and vectors, respectively. We employ nonbold letters (e.g., x) to denote scalars, and squiggle letters (e.g., $\mathcal{X}$) to denote sets. If not clarified, all vectors are in column forms.

Based on the user-item interaction (e.g., rating, clicking or purchasing) data used in representative recommended scenarios, we use the set $\mathcal{U}=\{u_1,...,u_n\}$ for all users and the set $\mathcal{V}=\{v_1,...,v_m\}$ for all items. Let $\mathbf{R}=[r_{ij}]_{n\times m}$ denote the user-item interactions matrix. Suppose $\mathcal{G}=\{g_1,...,g_l\}$ is the set consisting of all groups, and all members come from user set $\mathcal{U}$, i.e., group members with user index $K_l=\{k_{l,1},k_{l,2},...,k_l,|g_l|\}$, and $|g_l|$ is the size of the group. We denote the group-item interactions matrix as $\mathbf{Y}=[y_{lj}]_{s\times m}$. The symbols used in this paper are summarized in Table 1.

Table 1. Descriptions of the symbols used in this paper

Symbols	Description
n,m,l	number of users, items and groups
$\mathcal{U},\mathcal{V},\mathcal{G}$	set of users, items, groups
K_l	set of users in group
u_i,v_j,g_l	i-th user, j-th item and l-th group
$\|g_l\|$	the size of group g_l
$\mathbf{Y},\mathbf{R}$	matrices for grou-item and user-item
y_{lj},r_{ij}	rating of g_l over v_j and u_i over v_j
$\hat{y}_{lj},\hat{r}_{ij}$	estimated rating of g_l over v_j and u_i over v_j
$\boldsymbol{u}_i,\boldsymbol{v}_j$	emdedding of u_i and v_j
$\boldsymbol{g}_l(j)$	emdedding of l-th g_l

Input: Users $\mathcal{U}$, groups $\mathcal{G}$, items $\mathcal{V}$, group-item interactions matrix $\mathbf{Y}$, user-item interactions matrix $\mathbf{R}$.
Ouput: given a group g_l, through the prediction function, we output real value score.

3.2 Model Architecture

Figure 1 presents the architecture of GRUIFI for group recommendation with two major parts. (1) Representation learning. This part consists of the four layers: the input layer, the embedding layer, the attention layer and representation layer. The embedding layer look up features of groups, users and items to the representation of users and items. In the attention layer, through attention mechanism, it can dynamic learn the weight of each user that represent users' importance. In the representation layer, the user embedding aggregation is combined with group preference embedding as the representations of group. (2) Interaction learning. This part contains the interaction layer and prediction layer. In the interaction layer, a neural network combines with multi-attention is designed to learn a non-linear interaction function with respect to groups (or users) and items. The prediction layer with a fully connected layer (FCL) takes the hidden interaction vector as the input and returns the final prediction score.

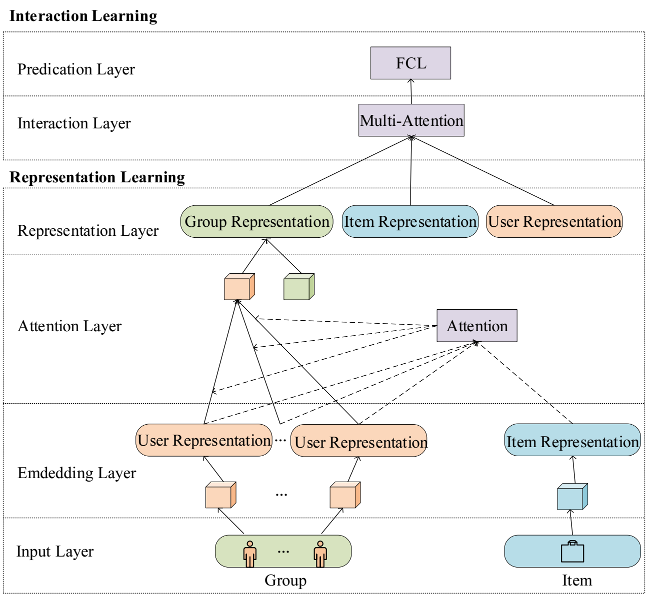


Figure 1. The architecture of consists of two major parts: representation learning and neural interaction learning

3.2.1 Representation Learning

We developed a novel model to predict the group's rating of items under the study of presentation learning. Representation learning represents each entity object as an embedding vector, and it learns the hidden attributes of the entity through data training. It uses feature vectors for calculation, and automatically learns low-dimensional representations of input entities from the training process. We take the item as an independent research object and set three entity objects: group, item and user.

To learn the representations for users and items, embedding extracted from the original data will be recognized as a fixed embedding. It reflects the inherent interests of a user and the inherent features of an item. For the group representation, we introduce a latent variable to represent the user importance. Therefore, we adopt the attention mechanism to learn this a latent variable, which can assign different weights to users. In this way, we can dynamically learn preference aggregation strategy to make a good decision for group.

Let $\mathbf{u}_i$ be the embedding vector of user that represent her/his preferences in the latent space, and $\mathbf{v}_j$ be the embedding vector of item. Both of them as the input of attention network. Our target is to obtain the representation of $\mathbf{g}_l(j)$ to estimate the group's preference on the item j through the representation of $\mathbf{u}_i$ and $\mathbf{v}_j$. Formally, it can be defined as:

$$\boldsymbol{g}_l(j) = \mathcal{F}(\{\boldsymbol{u}_t\}_{t\in K_l}, \boldsymbol{v}_j) \tag{1}$$

where $\mathbf{g}_l(j)$ denotes the representation learning of group g_l tailored for predicting its preference on target item v_j, K_l contains the user indexes of group, and $\mathcal{F}$ is the aggregation function to be specified. In GRUIFI, given group g_l and target item v_j, the group representation is merging the weighted sum over the user representations of all members and group preference embedding. Formally, it can be abstracted as:

$$\mathbf{g}_l(j) = \sum_{t\in K_l} \beta(t,j)\mathbf{u}_t + \mathbf{g}_q \tag{2}$$

To obtain the weighted sum over the user representations of all group members, we introduce an attention network, which uses the group-item interaction data to dynamically calculate the weights representing the importance of users. By capturing the attention weight distribution of users in the group on different items, the user embedding aggregation is obtained. We perform a weighted sum on the embeddings of users in group $g_l(j)$, where $\beta(t,j)$ is a learnable parameter denoting the user importance of user u_t in deciding the group choice on item v_j. And g_q is the representation of group preference embedding.

For a given group g_l, the attention network takes $\mathbf{u}_i$ and $\mathbf{v}_j$ as input, and then returns the attention weight $\beta(t,j)$ of all users in group g_l. It can be abstracted as:

$$o(t,j) = \mathbf{h}^T RuLU(\mathbf{P}_u u_t + \mathbf{P}_v v_j + \mathbf{b}) \quad (3)$$

$$\beta(t,j) = soft\max(o(t,j)) = \frac{\exp o(t,j)}{\sum_{t'\in K} \exp o(t',j)} \quad (4)$$

where $\mathbf{P}_u$, $\mathbf{P}_v$ and $\mathbf{b}$ are learnable parameters, $\mathbf{P}_u$ and $\mathbf{P}_v$ are weight matrices of the attention network that convert user embedding and item embedding, and $\mathbf{b}$ is the bias vector. When training the model, the weights and bias vectors of this part are randomly initialized, and then the corresponding values are automatically calculated during the training iteration of the model. The model uses ReLU as the activation function of the attention network layer. And then it maps the hidden layer to the output layer through the weight vector $\mathbf{h}$ to obtain the preference score $o(t,j)$. Lastly, the softmax function is used to normalize the preference score to get the preference weight of each user in the group.

3.2.2 Interaction Learning

From the work [21], the MLP could achieve better performance than conventional matrix factorization [25] in interaction learning. GRUIFI design a multi-head attention to learn fine-grained interactions with the inputs of the obtained representations of groups (or users) and items to learn the interaction between group g (or user u) and item v.

Figure 2 illustrates our customized solution. Given a user-item pair (u_i, v_j) or a group-item pair (g_l, v_j), the group representation, user representation and item representation obtained in the first part are used as input. Then the embedding vector is input to interaction layer and the hidden layers to obtain the group prediction score.

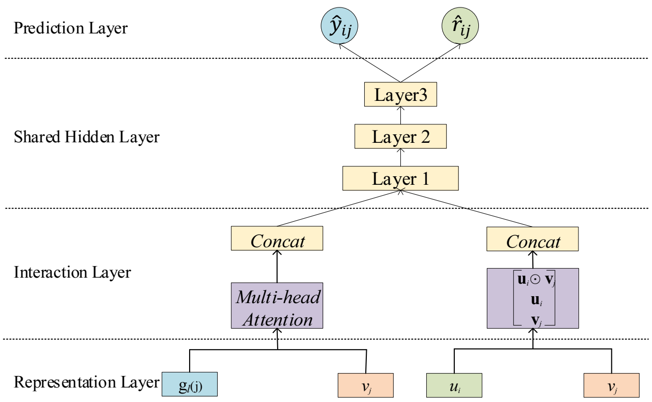


Figure 2. Interaction learning based a neural network combine with multi-head attention

In the interaction layer, for user-item pair (u_i, v_j), the element-wise product is used to model the interaction between the user and the item, which symbol is expressed as $\mathbf{u}_i \odot \mathbf{v}_j$. The element-wise product subsumes MF, which decomposes the user-item-rating matrix into a user factor matrix form and performs item-factor matrix multiplication [26]. At the same time, in order to reduce the missing information after the element-wise product, it is spliced with the original embedding of the user and the item to form a vector $\mathbf{e}_0$.

$$\mathbf{e}_0 = \phi_{interaction}(\mathbf{u}_i, \mathbf{v}_j) = \begin{bmatrix} \mathbf{u}_i \odot \mathbf{v}_j \\ \mathbf{u}_i \\ \mathbf{v}_j \end{bmatrix} \quad (5)$$

For group-item pair (g_l, v_j), the interaction layer first performs element-wise product on their embeddings, and then concatenates it with the original embeddings form a vector $\mathbf{e}_0$. After that, it input the multi-head attention with the original embeddings. The multi-head attention mechanism can extract different feature vectors in different subspace, and then concatenate all feature vectors into its output. Firstly perform linear transformation of $\mathbf{g}_l(j) \odot \mathbf{v}_j$, $\mathbf{g}_l(j)$ and $\mathbf{v}_j$, and then calculate the attention of the newly generated $(\mathbf{g}_l(j) \odot \mathbf{v}_j)'$, $(g_l(j))'$ and $(\mathbf{v}_j)'$ repeat this operation h times. It combines the result of h times. Finally do a linear transformation again, is the output of this small block of multi-head attention. The details are as follows:

$$\begin{aligned} \mathbf{e}_0 &= multihead(\mathbf{g}_l(j) \odot \mathbf{v}_j, \mathbf{g}_l(j), \mathbf{v}_j) \\ &= [head_1, \ldots\ldots, head_h] \end{aligned} \quad (6)$$

$head_i$ is denoted as

$$head_i = Attention((\mathbf{g}_l(j) \odot \mathbf{v}_j) W_i^H, \mathbf{g}_l(j) \mathbf{W}_i^S, \mathbf{v}_j \mathbf{W}_i^X) \quad (7)$$

and $\mathbf{W}_i^H$, $\mathbf{W}_i^S$ and $\mathbf{W}_i^X$ $\in R^{D\times d}$ are weighting matrices, h is the number of the basic attention mechanism, and $d = \frac{D}{h}$.

In Shared Hidden layers, a stack of fully connected layers above the interaction layer, which can capture the nonlinear and higher-order correlations among users, groups, and items.

$$\begin{cases} \mathbf{e}_1 = \mathrm{Re}LU(\mathbf{W}_1 \mathbf{e}_1 + \mathbf{b}_1) \\ \mathbf{e}_2 = \mathrm{Re}LU(\mathbf{W}_2 \mathbf{e}_2 + \mathbf{b}_2) \\ \ldots\ldots \\ \mathbf{e}_h = \mathrm{Re}LU(\mathbf{W}_h \mathbf{e}_h + \mathbf{b}_h) \end{cases} \quad (8)$$

where $\mathbf{W}_h$, $\mathbf{b}_h$ and $\mathbf{e}_h$ denote the weight matrix, bias vector, and output neurons of the *h*-th hidden layer. We use the ReLU function as the non-linear activation function, which has empirically shown to work well. Utilizing a FCL with the ReLU activation function predict the score of group g (or user u) on item v:

$$\begin{cases} \hat{r}_{ij} = \mathbf{w}^T \mathbf{e}_h,\ if \quad \mathbf{e}_0 = \varphi_{interaction}(u_i, v_j) \\ \hat{y}_{lj} = \mathbf{w}^T \mathbf{e}_h,\ if \quad \mathbf{e}_0 = \varphi_{interaction}(g_l(j), v_j) \end{cases} \tag{9}$$

where $\mathbf{w}$ is learnable parameters, and it denotes the weights of the prediction layer. $\hat{y}_{lj}$ represent the prediction for a group-item pair $g_l(j)$, and $\hat{r}_{ij}$ for a user-item pair.

3.3 Training Optimization

In this article, we recommend top-K items for groups based on implicit feedback and the perspective of ranking, so the pairwise learning method is selected to optimize the model parameters. The hypothesis of pairwise learning is that observed interactions should get a higher predicted score than its unobserved interactions. The popularly used pairwise learning methods in recommendation are regression-based pairwise loss and Bayesian personalized ranking (BPR). In BPR, in order to reduce the BPR loss of the multi-layer model, a simple solution is to increase the weight in each update. At this time, each weight needs to be L2 regularized, which will limit the weight learning. In the regression-based pairwise loss, the loss selected by the algorithm in this paper optimizes the marginal term to 1, avoiding L2 regularization to adjust the weight. And there is no restriction on weight learning. Thus regression-based pairwise loss is used:

$$L_u = \sum_{(i,j,s)\in R} (r_{ijs} - \hat{r}_{ijs})^2 = \sum_{(i,j,s)\in R} (\hat{r}_{ij} - \hat{r}_{is} - 1)^2 \tag{10}$$

where R denotes the training set, in which each instance is a triple (i, j, s) meaning that user u_i has interacted with item v_j, but has not interacted with item v_s. $\hat{r}_{ijs} = \hat{r}_{ij} - \hat{r}_{is}$ denotes the marginal term that predicts the observed and unobserved interactions. In this article, the focus is on implicit feedback, the value of each observed interaction is 1, and the unobserved is 0, we have $r_{ijs} = r_{ij} - r_{is} = 1$.

Similarly, we can obtain the pairwise loss function for optimizing the group recommendation task:

$$L_g = \sum_{(l,j,s)\in R'} (y_{ljs} - \hat{y}_{ljs})^2 = \sum_{(l,j,s)\in R'} (\hat{y}_{lj} - \hat{y}_{ls} \text{-} 1)^2 \tag{11}$$

In this paper, small batch training is used to train the GURIFI model, first disrupting all observed interactions, and then sampling a small batch of observed interactions. For each observed interaction, a fixed number of negative instances are selected as sample to form training instances together.

4 Experiment

In this section, we describe our experimental process and results analyses to evaluate the performance of our proposed GRUIFI. First, we introduce the dataset, baselines, and evaluation metrics. Then, we compare GRUIFI with other baseline methods. Finally, we evaluate GRUIFI performance under different attention mechanisms.

4.1 Experimental Settings

4.1.1 Datasets

We conduct experiments on two real-world datasets to evaluate the performance of our method GRUIFI. The two datasets were collected from Mafengwo[1] and CAMRa2011 [2] respectively, which used [10]. Mafengwo is a travel website in which users can record places where they travel and create or join group tours. The locations of each group member's travels were also collected. According to the above criteria, we obtained 5,275 users, 995 groups, 1,513 items, 39,761 user-item interactions, and 3,595 group-item interactions. CAMRa2011 is a real data set containing movie ratings records of individual users and families. Since the most users have no group information in the dataset, these users are filtered out and users who have joined the group are retained. The user-item interaction and group-item interaction are explicit feedback. The score ranges from 0 to 100. The scored records are converted into positive instances with a target value of 1, and the remaining missing data is retained as a negative target value of 0. The dataset contains 602 users, 290 groups, 7,710 items, 116,344 user-item interactions, and 145,068 group-item interactions finally. In the recommended scenario in this article, the same user can belong to multiple groups. The data statistics of the two datasets are shown in Table 2.

Table 2. The data statistics of the two datasets

Dataset	Mafengwo	CAMRa2011
Numbers of user	5275	602
Numbers of group	995	290
Numbers of item	1513	7710
Numbers of group-item	3595	145068
Numbers of user-item	39761	116344

In this paper, the negative sampling ratio is set to 4. For the hidden layer, their parameters are initialized randomly. The mean value of the Gaussian distribution is 0, and the standard deviation is 0.1. Debug the most appropriate learning rate in [0.0001, 0.00005, 0.000005]. In attention networks and multi-head attention networks, the size of the first hidden layer is set to be the same as the embedded size.

1 http://www.mafengwo.cn/
2 http://2011.camrachallenge.com/2011

4.1.2 Compared Methods

To verify the superiority of our model, we compare the proposed model with the following models. At the same time, in order to verify the effectiveness of the aggregation strategy learned from the interactive data, we designed the traditional static aggregation strategy combined with NCF as a comparison method. In these methods, we first use NCF to predict the preference scores of individual users in the group, and then apply aggregation strategy to obtain the preference scores of the group.

- NCF [21] is a state-of-art collaborative filtering model that model the interaction between users and items. This method treats a group as a user, which is embedded into the NCF.
- NCF-AVG, NCF-LM and NCF-MS combine NCF with the predefined aggregation strategies including average [13], least misery [14], and maximum satisfaction [15] that take the average, minimal and maximal score of all members as the group's score.
- AGREE [10] aggregates the users' preferences with the group preference via an attention network and adopts NCF to model the interactions between groups and items.
- soAGREE [24] uses a dual-level attention network to obtain group representations through user social attribute information.

In order to test the effectiveness of attention and multi-headed attention, we set up the following two methods.

- GRUIFI-M is a variant of GRUIFI method by removing the multi-head attention component that is applied to the interaction between the group and the item.
- GRUIFI-A is removing the attention component in GRUIFI, and the method sets a uniform weight on the member embeddings.

4.1.3 Evaluation Metrics

In this paper, we use the leave-one-out method for evaluation, which is widely used to evaluate the performance of top-K. We evalutate recommendation accuracy with K = 5 or 10, K is the number of recommendations. Specifically, for each user (group), we randomly removed one of its interactions for testing. This results in disjoint training set D_{train} and testing set D_{test}. Randomly select 100 items that have no interaction with the group. The evaluation metric used in this article are Hit Ratio (HR) and normalized discounted cumulative gain (NDCG). HR can measure the accuracy of recommendations. NDCG measures the performance of recommendation lists.

We pick the K items with the highest score to form the top K recommendation list. If the most authentic recommendation item v_j appears in the top K recommendation list, we will hit. Otherwise, we will miss it. The metric Hits ratio (HR) is defined as follow:

$$HR@K = \frac{\#hit@K}{|Dtset|} \quad \textbf{(12)}$$

where $\#hit@K$ denotes the number of hits in the test set, and $|Dtest|$ is the total number of test cases in the test set. HR measures the recall rating and measures the accuracy of recommendations. The higher the Hits@K, the better the effect.

Besides HR, we also adopt the commonly used Normalized Discounted Cumulative Gain (NDCG) to measure the ranking quality of a commendation list. NDCG can simultaneously pay attention to the relevance of items in the recommended list and the ordering of items, and it is defined as follow:

$$NDCG@K = Z_K \sum_{i=1}^{K} \frac{2^{r_i}-1}{\log_2(i+1)} \quad \textbf{(13)}$$

where Z_K is the normalization coefficient, which represents the reciprocal of the sum of the summation formula in the best case, and the purpose is to keep the value of NDCG within 0-1. r_i represents the correlation of the recommended results in the position, and if it hits, it is 1, otherwise 0. NDCG measures the performance of different recommendation lists by comparing the difference between the current DCG value and the ideal IDCG (Idea DCG) value. Therefore, the larger the NDCG value, the better the performance of the algorithm.

Similarly, we use the above evaluation indicators to the personalized recommendation for individual users.

4.2 Overall Performance Comparison

To verify the superiority of our model, we compare the performance of GRUIFI with the baselines of state-of-the-art group recommendation system. We divide baselines into two categories: the first category is the traditional method that does not consider the importance of users and user ratings is static distributed (i.e., NCF, NCF-AVG, NCF-LM and NCF-MS methods). The second category is the model-based method (i.e., AGREE and soAGREE) that uses the attention network and considers the dynamic distribution of user importance.

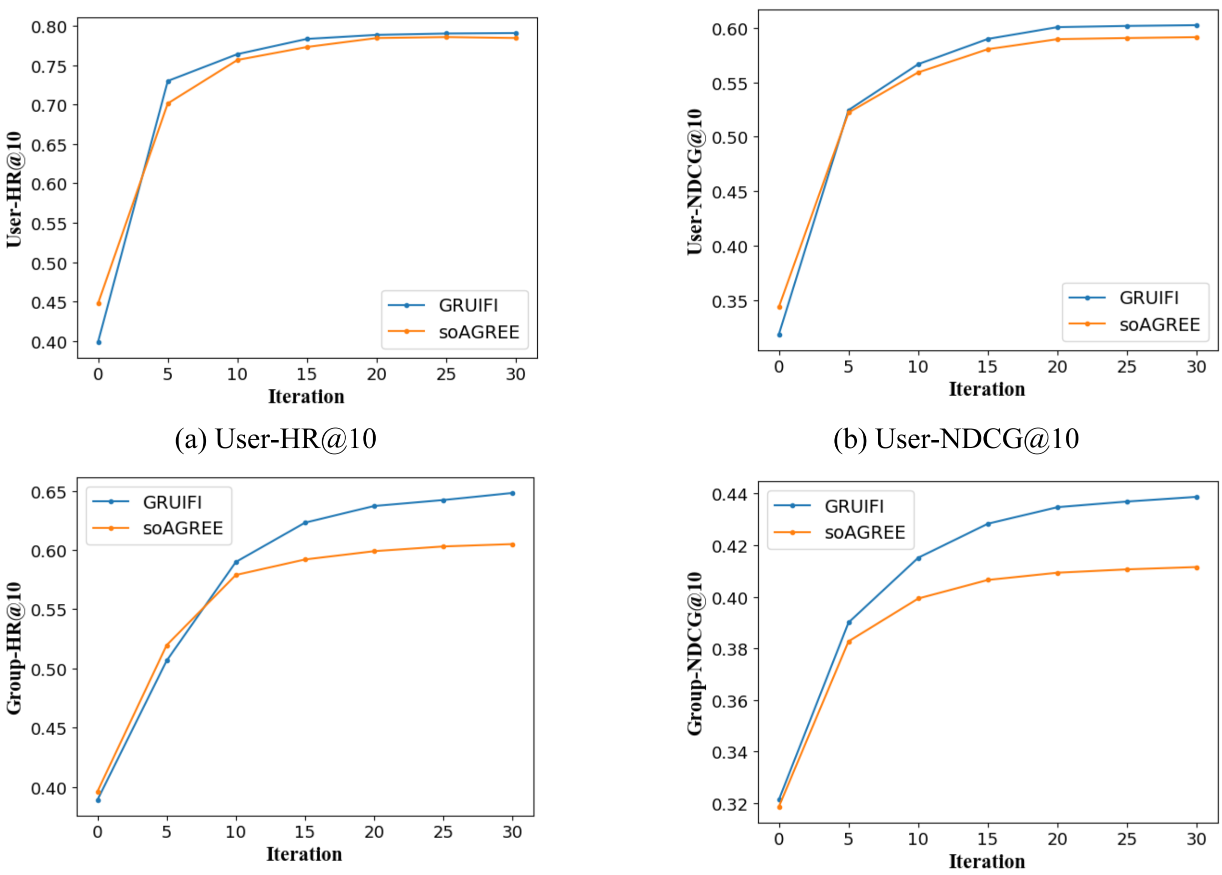
Table 3 and Table 4 show the results on Mafengwo and CAMRa2011 under different settings in terms of HR@K and NDCG@K. The recommendation number K is 5 and 10 for experiments respectively. We have the following observations. Firstly, it can be seen that when the recommendation number is 10, better results are achieved on both datasets. Secondly, regardless of recommendation for a user or a group, the HR or the NDCG based on the GRUIFI method is higher than that of the baselines, which indicates the superiority of our method over other strategies in terms of the quality of ranking. This was to be expected, because the GRUIFI method mined the user's hermit characteristics from the user's history and learned the dynamic fusion strategy to build the appropriate model for the group to further recommend. In addition, the GRUIFI method can explore the interaction of fine-grained features of group members from interaction data, so the GRUIFI method has a good effect. Lastly, the approach combining NCF with AVG, MS and LM strategies has no obvious better effect. In particulart, both AGREE and GRUIFI are generally better than NCF-AVG, NCF-MS and NCF-LM method, as they consider the distinct importance of the members in a group. This paper proves that the predefined static score aggregate strategy is not enough to predict group decision-making well. It also illustrates the necessity of dynamically learning user weights.

Table 3. Overall performance comparison for Users on the Mafengwo and CAMRa2011 datasets

Metric	Mafengwo				CAMRa2011			
	K=5		K=10		K=5		K=10	
	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
AGREE	0.6383	0.5502	0.7491	0.5775	0.6223	0.4118	0.7967	0.4687
GRUIFI	**0.7067**	**0.5699**	**0.7904**	**0.6022**	**0.6282**	**0.4239**	**0.8013**	**0.4799**

Table 4. Overall performance comparison for Groups on the Mafengwo and CAMRa2011 datasets

Metric	Mafengwo				CAMRa2011			
	K=5		K=10		K=5		K=10	
	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
NCF	0.4701	0.3657	0.6269	0.4141	0.5803	0.3896	0.7693	0.4448
NCF+avg	0.4774	0.3669	0.6222	0.4140	0.5689	0.3819	0.7611	0.4452
NCF+lm	0.4744	0.3631	0.6302	0.4152	0.5593	0.3788	0.7648	0.4455
NCF+ms	0.4700	0.3616	0.6281	0.4114	0.5434	0.3710	0.7607	0.4348
NCF+exp	0.4724	0.3647	0.6251	0.4015	0.5648	0.3787	0.7621	0.4426
AGREE	0.4814	0.3747	0.6400	0.4244	0.5883	0.3955	0.7807	0.4575
GRUIFI	**0.5769**	**0.4726**	**0.6482**	**0.4387**	**0.5890**	**0.3995**	**0.7972**	**0.4673**



(a) User-HR@10

(b) User-NDCG@10

(c) Group-HR@10

(d) Group-NDCG@10

Figure 3. Performance of GRUIFI and soAGREE in each training iteration on Mafengwo datasets

Figure 3 shows the change trend of the evaluation indicators of GRUIFI and soAGREE in each iteration. It can be seen from the figure. Compared with SoAGREE, the method of GRUIFI in this paper has relative progress on both data sets, which depends on the deep mining of group members interaction learning. The convergence speed of the two methods is very fast, and basically stabilized at the 20th iteration. Except for Group-NDCG@10, all others have intersection points. That is, when the number of iterations is small, the result of soAGREE is better than GRUIFI. On the contrary, when the number of iterations increases, the result of GRUIFI is better than soAGREE.

4.3 Study on the Components of GRUIFI

In this paper, the overall performance comparison shows that GRUIFI obtains the best results, demonstrating the effectiveness of the integrated part of our model. To further understand the importance of group preference embedding and attention in learning group representation, and the multi-attention in interaction leraning, we performed some ablation studies. Here we study the components of GRUIFI by evaluating the three variants: (1) GRUIFI-M is a variant of GRUIFI method by removing the multi-head attention component that is applied to the interaction between the group and the item. This demonstrates the effectiveness of learning high-order feature interactions by multi-attention. (2) GRUIFI-A is removing the attention component in GRUIFI, and it sets the average weight on the member embeddings. This is to demonstrate the importance of the attention network, which can dynamically learn different weight. (3) GRUIFI-G is removing the group preference embedding, which is to prove the importance of embedding group preference.

Table 5 show the results of AGREE and the three simplified variants. We have the following observations. GRUIFI consistently and significantly outperforms GRUIFI-M, GRUIFI-A and GRUIFI-G on both datasets with respect to both metrics. This indicates that both components of attention and multi-attention are beneficial to model group decisions, and combining them leads to better performance. The group preference embedding has a larger impact in learning group representation in our method.

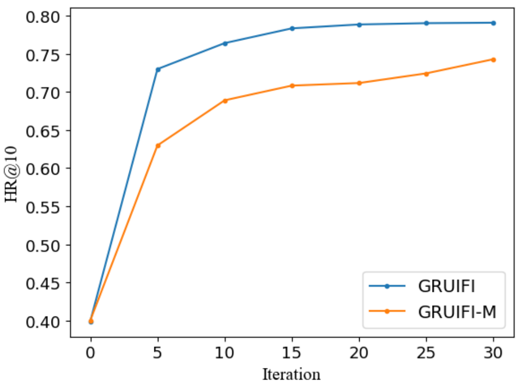
Table 5. Top-10 performance of GRUIFI and its three simplified variants on the Mafengwo and CAMRa2011 datasets

Metric	Mafengwo				CAMRa2011			
	User		Group		User		Group	
	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
GRUIFI-M	0.7431	0.5834	0.4744	0.3701	0.7815	0.4626	0.7779	0.4154
GRUIFI-A	0.7739	0.5788	0.5618	0.3890	0.7642	0.4313	0.7472	0.4368
GRUIFI-G	0.7808	0.5873	0.5769	0.3976	0.8003	0.4519	0.7807	0.4382
GRUIFI	0.7904	0.6022	0.6482	0.4387	0.8013	0.4799	0.7972	0.4673

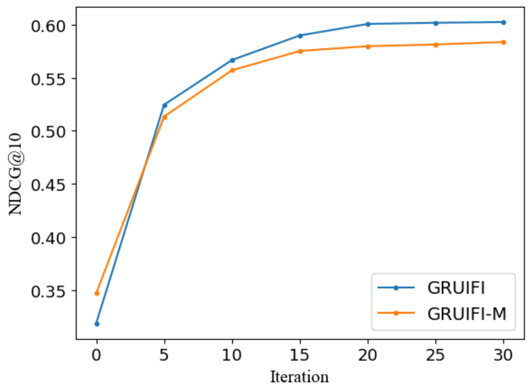
Figure 4 shows the performance of GRUIFI and GRUIFI -M in Mafengwo dataset under the under different settings in terms of HR@10 and NDCG@10. We have the following observations. Compared with GRUIFI-M, GRUIFI achieves a better result on Mafengwo dataset under the metrics of HR and NDCG. Especially for group recommendations, this method has achieved better results. But the data of the previous three iterations, GRUIFI-M is higher than GRUIFI for the metrics NDCG of user. It may be that there is no deep-level information that can be extracted from the user's internal structure. By contrast with GRUIFI-M, GRUIFI uses the multi-attention that has achieved certain results in interaction. This indicates that integrating implicit feature interactions indeed boosts the predictive ability of our proposed model. The utilization of multi-head attention contributes to model complexity of group interaction in real life. It is desirable to improve the recommendation function through interaction between group members.

Figure 5 shows the performance of GRUIFI-A and GRUIFI in each training iteration under different settings in terms of HR@10 and NDCG@10. We have the following observations. Compared with GRUIFI-A, GRUIFI achieves a relative improvement on Mafengwo datasets with respect to both the HR and NDCG metrics.

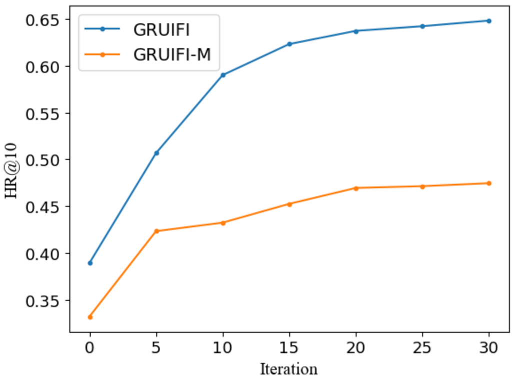
When the model is recommended for users, both GRUIFI and GRUIFI-A converge quickly and tends to be flat. When the number of iterations is 0 to 5, the metric values of HR@10 and NDCG@10 rise rapidly, reaching stable performance before and after the 20th iteration. While for a group, the convergence speed is gradually increasing from fast to slow. This reflects the importance of the attention network for group recommendation, because it can dynamically learn the weight that represents the importance of group members. This method pays more attention to the different user importance of different users in the group decision-making process, and it can dynamically learn the influence of users, thereby alleviating conflicts between group members. Without the attention mechanism, it is impossible to model complexity of real life group decision-making.



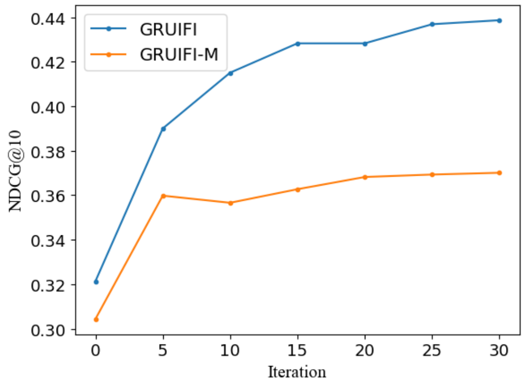
(a) User-HR@10



(b) User-NDCG@10

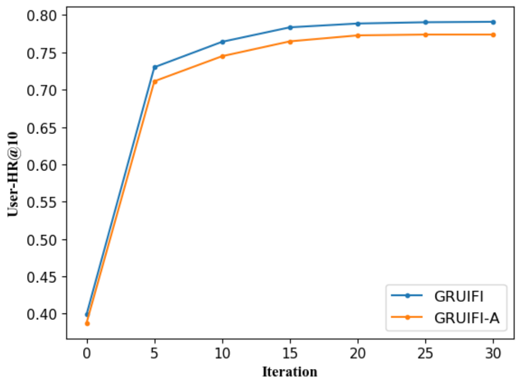


(c) Group-HR@10

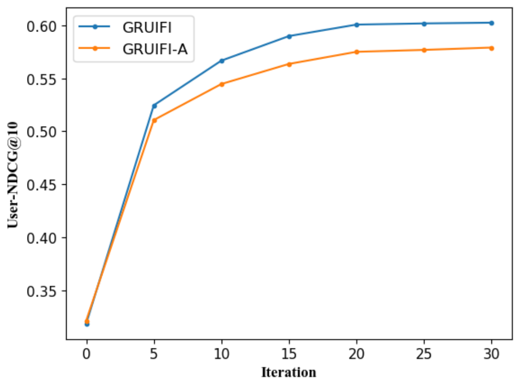


(d) Group-NDCG@10

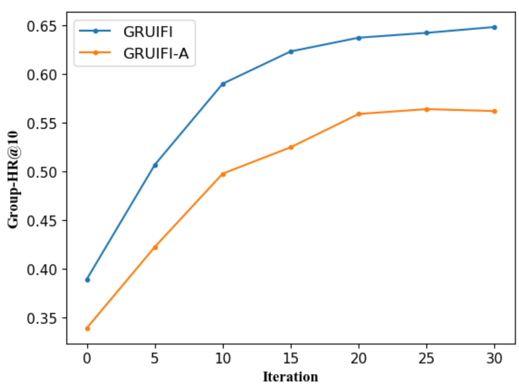
Figure 4. Performance of GRUIFI and GRUIFI-M in each training iteration on Mafengwo datasets



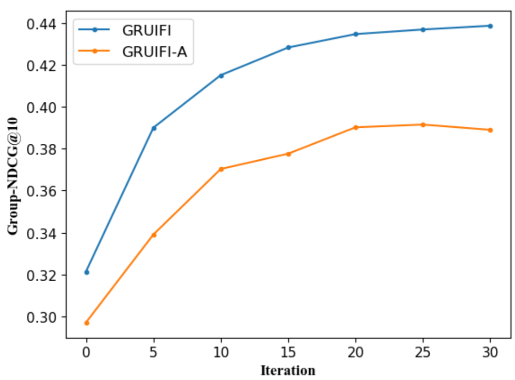
(a) User-HR@10



(b) User-NDCG@10



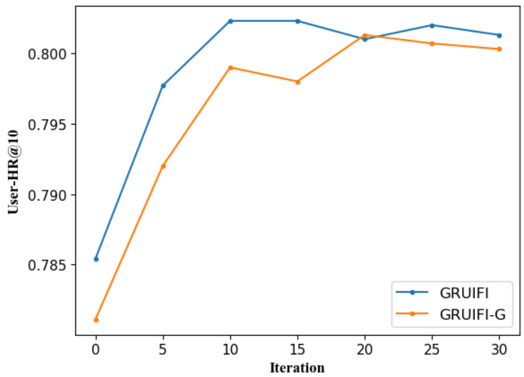
(c) Group-HR@10



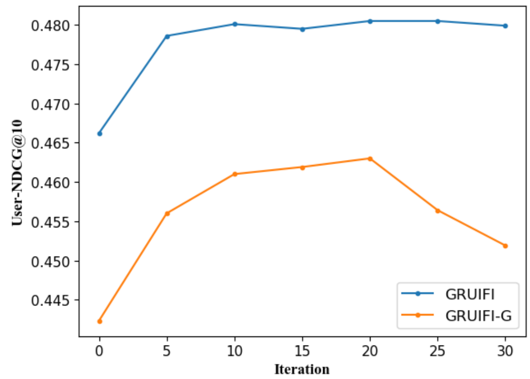
(d) Group-NDCG@10

Figure 5. Performance of GRUIFI and GRUIFI-A in each training iteration on Mafengwo datasets

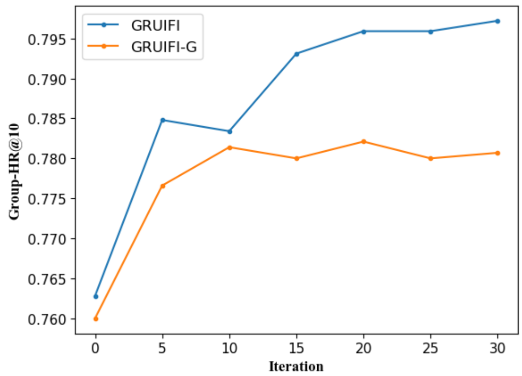
Figure 6 shows the performance of GRUIFI-G and GRUIFI in each training iteration under different settings in terms of HR@10 and NDCG@10. We have the following observations. Compared with GRUIFI-G, GRUIFI achieves a relative improvement on CAMRa2011 datasets with respect to both the HR and NDCG metrics. GRUIFI consistently and significantly outperforms GRUIFI-G on both datasets with respect to both metrics. This indicates that the group preference embedding are effective for modeling group preference, and combining them can lead to better performance. The group recommendation method proposed in this paper considers the comprehensiveness and effectiveness of the group representation obtained by the project and the user when predicting the score.



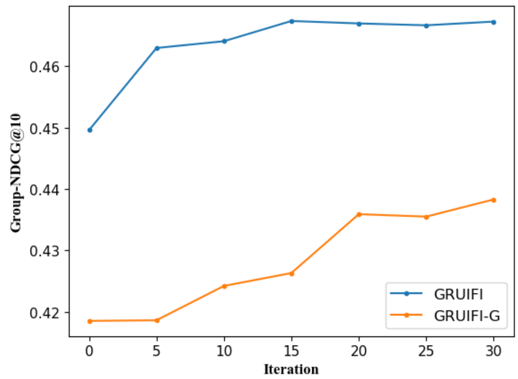
(a) User-HR@10



(b) User-NDCG@10



(c) Group-HR@10



(d) Group-NDCG@10

Figure 6. Performance of GRUIFI and GRUIFI-G in each training iteration on CAMRa2011 datasets

Compared with the other three methods that only use attention or multi-head attention, and removes the group preference embedding, GRUIFI has better recommendation results. This shows that the group recommendation method proposed in this article also considers the different importance of users in the group and the influence of group characteristics and interactions between group members when predicting the score.

5 Conclusion

This paper studied two key issues in the group recommendation problem from the perspective of neural representation learning, namely obtaining group representation and modeling the interaction between groups and items. Specifically, according to the idea of Model-based approaches, this article modeled the representations of users and items through taking advantage of an embedding layer. Adopting the attention mechanism learn the weight of group members that represents the importance of users, and then aggregated the representations of group members as the group representation. Then using the deep learning framework learn interaction for group recommendation. This article proposed a novel idea, designing a multi-head attention for granular interaction between group members, not only considered the relationship between user preferences and items, but also considered that users will be affected by other users in group activities. This is not fully considered in previous studies. The evaluation metric of our proposed method is better than other methods, which further verifies the effectiveness of our proposed method. However, this article did not consider other characteristics of the group. It is the future work to consider more attributes to improve the accuracy of group recommendation and improve the quality of the recommendation list.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61862013, 61662015, U1811264, U1711263), the Natural Science Foundation of Guangxi (2020GXNSFAA159117, 2018GXNSFAA 281199, 2017GXNSFAA198035), the Foundation of both Guangxi Key Laboratory of Trusted Software (KX202052) and Guangxi Key Laboratory of Automatic Dectection Technology and Instrument (YQ19109).

References

[1] H. Gao, C. Liu, Y. Li, X. Yang, V2VR: Reliable Hybrid-Network-Oriented V2V Data Transmission and Routing Considering RSUs and Connectivity Probability, *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, pp. 1-14, April, 2020.

[2] X. Ma, H. Gao, H. Xu, M. Bian, An IoT-based task scheduling optimization scheme considering the deadline and cost-aware scientific workflow for cloud computing, *EURASIP Journal on Wireless Communications and Networking*, Vol. 2019, Article No. 249, November, 2019.

[3] H. Gao, L. Kuang, Y. Yin, B. Guo, K. Dou, Mining Consuming Behaviors with Temporal Evolution for Personalized Recommendation in Mobile Marketing Apps, *ACM/Springer Mobile Networks and Applications (MONET)*, Vol. 25, No. 4, pp. 1233-1248, August, 2020.

[4] Q. Yang, H. Wang, M. Bian, Y. Lin, J. Zhang, Incorporating Reverse Search for Friend Recommendation with Random Walk, *The International Arab Journal of Information Technology*, Vol. 17, No. 3, pp. 291-298, May, 2020.

[5] Q. Yang, S. Zhou, H. Li, J. Zhang, Embedding Implicit User Importance for Group Recommendation, *Computers, Materials & Continua*, Vol. 64, No. 3, pp. 1691-1704, 2020.

[6] D. Herzog, W. Wörndl, A User Study on Groups Interacting with Tourist Trip Recommender Systems in Public Spaces, *2019 27th ACM Conference on User Modeling, Adaptation and Personalization (UMAP)*, Larnaca, Cyprus, 2019, pp. 130-138.

[7] Y. Du, X. Meng, Y. Zhang, P. Lv, GERF: A Group Event Recommendation Framework Based on Learning-To-Rank, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 32, No. 4, pp. 674-687, April, 2020.

[8] X. Liu, Y. Tian, M. Ye, W.-C. Lee, Exploring Personal Impact for Group Recommendation, *2012 21st ACM International Conference on Information and Knowledge Management (CIKM)*, Maui, Hawaii, USA, 2012, pp. 674-683.

[9] H. Yin, Q. Wang, K. Zheng, Z. Li, J. Yang, X. Zhou, Social Influence-Based Group Representation Learning for Group Recommendation, *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, Macao, China, 2019, pp. 566-577.

[10] D. Cao, X. He, L. Miao, Y. An, C. Yang, R. Hong, Attentive Group Recommendation, *2018 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR)*, Ann Arbor, MI, USA, 2018, pp. 645-654.

[11] Q. Yuan, G. Cong, C. -Y. Lin, COM: A Generative Model for Group Recommendation, *2014 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'14)*, New York, USA, 2014, pp. 163-172.

[12] H. Gao, W. Huang, Y. Duan, The Cloud-edge-based Dynamic Reconfiguration to Service Workflow for Mobile Ecommerce Environments: A QoS Prediction Perspective, *ACM Transactions on Internet Technology*, Vol. 21, No. 1, pp. 1-23, February, 2021.

[13] L. Baltrunas, T. Makcinskas, F. Ricci, Group Recommendations with Rank Aggregation and Collaborative Filtering, *2010 4th ACM Conference on Recommender Systems (RecSys)*, Barcelona, Spain, 2010, pp. 119-126.

[14] S. Amer-Yahia, S. B. Roy, A. Chawlat, G. Das, C. Yu, Group recommendation: Semantics and efficiency, *2009 35th International Conference on* Very Large Data Bases *(VLDB)*, Lyon, France, 2009, pp. 754-765.

[15] L. Boratto, S. Carta, State-Of-The-Art in Group Recommendation and New Approaches for Automatic Identification of Groups, in: A. Soro, E. Vargiu, G. Armano, G. Paddeu (Eds.), *Information Retrieval and Mining in Distributed Environments*, Vol. 324, Springer, Berlin, Heidelberg, 2010, pp. 1-20.

[16] L. V. Tran, T.-A. N. Pham, Y. Tay, Y. Liu, G. Cong, X. Li, Interact and Decide: Medley of Sub-Attention Networks for Effective Group Recommendation, *2019 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, Paris, France, 2019, pp. 255-264.

[17] Z. Yu, X. Zhou, Y. Hao, J. Gu, TV Program Recommendation for Multiple Viewers Based on User Profile Merging, *User Modeling and User-Adapted Interaction*, Vol. 16, No. 1, pp. 63-82, March, 2006.

[18] J. Guo, Y. Zhu, A. Li, Q. Wang, W. Han, A Social Influence Approach for Group User Modeling in Group Recommendation Systems, *IEEE Intelligent Systems*, Vol. 31, No. 5, pp. 40-48, September-October, 2016.

[19] H. Zhao, Q. Liu, Y. Ge, R. Kong, E. Chen, Group Preference Aggregation: A Nash Equilibrium Approach, *2016 IEEE 16th International Conference on Data Mining (ICDM)*, Barcelona, Spain, 2016, pp. 679-688.

[20] S. Okura, Y. Tagami, S. Ono, A. Tajima, Embedding-Based News Recommendation for Millions of Users, *2017 23rd ACM SIGKDD international conference on Knowledge Discovery and Data Mining (KDD'17)*, Halifax, NS, Canada, 2017, pp. 1933-1942.

[21] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, T. Chua, Neural Collaborative Filtering, *2017 26th International Conference on World Wide Web (WWW)*, Perth, Australia, 2017, pp. 173-182.

[22] Y. Yin, Z. Cao, Y. Xu, H. Gao, R. Li, Z. Mai, QoS Prediction for Service Recommendation with Features Learning in

Mobile Edge Computing Environment, *IEEE Transactions on Cognitive Communications and Networking*, Vol. 6, No. 4, pp. 1136-1145, December, 2020.
[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention Is All You Need, *2017 31st Conference on Neural Information Processing Systems (NIPS)*, Long Beach, CA, USA, 2017, pp. 6000-6010.
[24] D. Cao, X. He, L. Miao, G. Xiao, H. Chen, J. Xu, Social-Enhanced Attentive Group Recommendation, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 33, No. 3, pp. 1195-1209, March, 2021.
[25] Y. Koren, R. Bell, C. Volinsky, Matrix Factorization Techniques for Recommender Systems, *Computer*, Vol. 42, No. 8, pp. 30-37, August, 2009.
[26] X. Yang, S. Zhou, M. Cao, An Approach to Alleviate the Sparsity Problem of Hybrid Collaborative Filtering Based Recommendations: The Product-Attribute Perspective from User Reviews, *Mobile Networks and Applications*, Vol. 25, No. 2, pp. 376-390, April, 2020.

Biographies

Jingwei Zhang, born in 1977. He is currently a Professor with the School of Computer and Information Security, Guilin University of Electronic Technology, China. His main research interests include big data management, analysis and smart applications.

Jing Cheng, born in 1996. She received the bachelor's degree from JiShou University, China, in 2018. She is currently pursuing the master's degree with the Guilin University of Electronic Technology, China. Her research interest includes recommendations.

Ya Zhou, born in 1966. She is currently a Professor with the School of Computer and Information Security, Guilin University of Electronic Technology, China. Her main research interests include distributed computing and optimization, and big data analysis.

Qing Yang, born in 1976. She is currently a Professor with the School of Electronics Engineering and Automation, Guilin University of Electronic Technology, China. Her main research interests include intelligent information process and recommendations.