IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 18, 2017, accepted December 26, 2017, date of publication January 12, 2018, date of current version March 9, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2792427

A Novel Group Recommendation Mechanism From the Perspective of Preference Distribution

ZHIWEI GUO[1], CHAOWEI TANG[1], HUI TANG[1], YUNQING FU[2], AND WENJIA NIU[3]
[1]College of Communication Engineering, Chongqing University, Chongqing 400044, China
[2]College of Software, Chongqing University, Chongqing 400044, China
[3]Beijing Key Laboratory of Security and Privacy in Intelligent Transportation, Beijing Jiaotong University, Beijing 100044, China

Corresponding authors: Chaowei Tang (cwtang@cqu.edu.cn) and Hui Tang (tangh@net-east.com)

This work was supported by the National Natural Science Foundation of China under Grant 61672091.

ABSTRACT As people have become more and more connected, there are certain scenarios where items need to be recommended to groups of users rather than individual user, which motivate studies on group recommender systems (GRSs). However, developing GRSs is not an easy task, because a group consists of multiple members with heterogeneous preferences. How to make a trade-off among their preferences remains challenging. Existing works almost aggregate members' preferences into forms of single values as group profile. However, simple aggregations fail to well reflect comprehensive group profile when it comes to groups with highly conflicting preferences. In this paper, we propose Greption, a novel group recommendation mechanism from the perspective of preference distribution. First, based on preference distributions toward items in training set, a multi-dimensional support vector regression model is established to predict preference distributions toward candidate items. Then, through a modified VIKOR method, we transform the process of selecting items for a group into a multi-criteria decision making process. Furthermore, the Greption is extended to be able to handle data sparsity. Specifically, we propose two heuristic schemes for this purpose. And we present a set of experiments to evaluate the efficiency of the Greption.

INDEX TERMS Group recommendation mechanism, preference distribution, multi-criteria decision making, collaborative filtering.

I. INTRODUCTION

With rapid development of the Internet, more and more online services inevitably suffer from information overload problem [16], [30], which makes it difficult for users to find the information they require [47], [48], [51]. Recommender systems (RSs) [5], [41], [42] is one popular technique proposed to address this issue, and it suggests items for users according to their preferences. In the past decade, RSs have been extended to more general fields, such as music, movies, news, etc. Despite a lot of research efforts devoted in RSs, most prior studies focused on providing recommendations to individual users. However, as people have become more and more connected, there are certain scenarios where items need to be recommended to groups of users rather than individual user [29], [39], [46], [52]. Fig. 1 illustrates some typical examples of these scenarios:

1) Members of a family gather together to watch TV. It is not easy to select suitable TV shows for them all.
2) Friends gather to have dinner. Due to their different tastes, it remains hard to recommend a table of dishes for them.
3) For a community library, it is important to configure ratios of various categories of books to satisfy people's conflicting preferences.
4) Entertainment systems on board can carry multimedia resources. But due to limited storage space, appropriate resources need to be selected to satisfy all the passengers.

Group Recommender Systems (GRSs) aim to provide recommendations that satisfy preferences of all users in a group [7]. However, making recommendations for groups of users remains hard, because groups usually consist of multiple users with heterogeneous preferences. How to make a trade-off among them is a really challenge [6], [24]. As for the group formation, there are two types of groups in GRSs [3], [19]: persistent groups and ephemeral groups. Persistent groups refer to a type of communities in which people

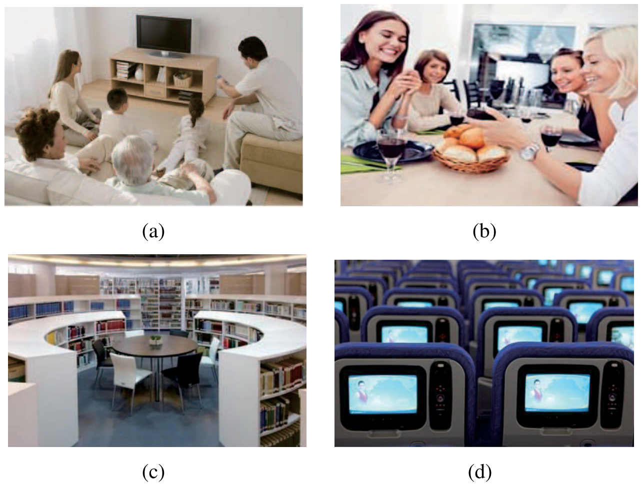

FIGURE 1. **Typical scenarios concerning recommending items to groups of users.**

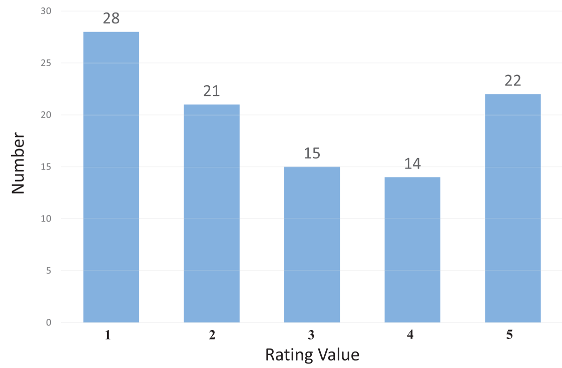

FIGURE 2. **Simple aggregations cannot well reflect comprehensive group profile.**

have social relations or share common social circles (e.g. classmates, colleagues, friends). Ephemeral groups represent a type of communities in which people gather occasionally or randomly, not because of social relationship or anything in common (e.g. passengers occasionally taking the same airplane). Most of researches concerning GRSs are related to the former, because additional information (such as social relations) can be introduced as auxiliary factor [12], [13], [17], [37], [38], [43], [45], [54]. Our research in this paper focuses on GRSs for the ephemeral groups [1], [2] [20]–[22], [25], [28], [29], [33], [36], [50], [52]. Despite difference in these methods, almost all of them just aggregate members' preferences into single values as group preferences through different aggregation functions. Note that preferences in RSs generally refer to ratings given by users towards items. Specifically, members' ratings towards items are aggregated into forms of single rating values as group profile through simple aggregation functions, such as average, least misery, weighted average, etc. Then, conventional methods of individual recommendations like matrix factorization (MF) are utilized to generate recommendations for the group. Nevertheless, simple aggregations usually fail to reflect conflicting characteristics of group preferences. Thus, these methods cannot well deal with groups with highly conflicting preferences, especially large-scale groups. We give an example of such case. We assume that 100 members constitute an ephemeral group and give ratings to an item according to their preferences. The ratings are integers ranging from 1 to 5. Rating Distribution of the 100 members is shown in Fig. 2. Although average value of their ratings is 2.8, half of members give ratings as 1 or 5. Therefore, simple aggregation-based methods cannot model comprehensive group profile.

In order to capture the conflicting characteristics of group preferences, this research regards group profile as the distributions of members' preferences towards items. Thus, comprehensive group profile can be well reflected. In this paper, a novel *G*roup *Re*commendation Mechanism from the Perspective of *P*reference Distribu*tion* (**Greption**) is proposed. First, idea of Label Distribution Learning (LDL) [14], [15] is introduced to model group profile. Specifically, based on preference distributions towards items in training set, a multi-dimensional support vector regression model is established to predict preference distributions towards candidate items. Then, through a modified VIKOR method, we transform the process of selecting items for a group into a multi-criteria decision making process (MCDM). Furthermore, the Greption is extended to handle the sparse data. Specifically, we propose two heuristic schemes—Preference Average (PA) and User-based Collaborative Filtering (CF) for this purpose. To the best of our knowledge, we are the first to develop group recommendation mechanism from the perspective of preference distribution. The main contributions of this paper are summarized as following aspects:

- We consider the limitation of simple aggregation-based methods, and regard the group profile as the distribution of members' preferences towards items.
- In order to generate recommendations under preference distributions, we transform the process of selecting items for a group into an MCDM to optimize recommendation results.
- The Greption approach is extended to be able to handle the sparse data.
- A set of experiments are conducted on two real-world datasets to demonstrate efficiency of the proposed approach.

The rest of this paper is organized as follows. In Section II, we give summary about the related work. In Section III, we describe Greption framework and functionality of its components. The group recommendation mechanism from the perspective of preference distribution is proposed in Section IV. Section V presents the experimental results with analysis. Finally, we conclude this paper in Section VI.

II. RELATED WORK

This study attempts to propose a novel group recommendation mechanism. The following subsections briefly state prior studies related to our research.

A. GROUP RECOMMENDATION FOR PERSISTENT GROUPS

Most of relevant researches focus on GRSs for the persistent groups, because additional information (such as social relations) can be introduced as auxiliary factor. Salehi-Abari and Boutilier [45] developed a probabilistic inference method to predict individual preferences given observed social connections as well as partially observed preferences of others. Then, they exploited these predictions in a social choice context to make recommendations for groups. Quijano-Sanchez et al. [40] proposed ARISE, an approach that makes use of social information to produce enhanced recommendations to groups. Gartrell et al. [13] proposed a group consensus function. The function studies the social, expertise, and interest dissimilarity among members. Seko et al. [47] proposed an algorithm that calculates recommendation scores using a feature space based on relationships among group members. Yuan et al. [55] proposed a probabilistic model named COmcensus Model (COM) to model the generative process of group activities, and infer users' preferences to make recommendations. Bento et al. [3] formalized the problem of making recommendations to persistent groups as a suitable Multiarmed bandit (MAB) problem. Hong et al. [17] proposed Group Recommendation based on social affinity and trustworthiness. Feng and Cao [12] proposed a approach for group recommendation based on random walk with restart model. This approach predicts more accurate group preferences with the aid of graphical relationships among users and items. Zhang et al. [56] pioneered a Nash equilibrium based Item Group Recommendation approach (NIGR). Specifically, they evaluated each user's preference towards an item group from two perspectives: individual preference and social affection. Then, recommendation process was modeled as a game to achieve Nash equilibrium. In [9], Christensen et al. presented a group recommender system for tourism, which models group profile by analyzing not only members' preferences, but also the social relationships among group members. Kim and El Saddik [24] introduced a new graph model utilizing fruitful tagging information. Based on this, they presented a stochastic method that generates recommendations according to link-structure analysis in a probabilistic manner. Liu et al. [27] modeled group profile by considering the personal preferences and personal impacts.

B. GROUP RECOMMENDATION FOR EPHEMERAL GROUPS

GRSs for ephemeral groups had also been researched in recent years. In [1], Ardissono et al. used a weighted form of the Average Strategy to make recommendations. In [29], O'connor et al. used the Least Misery Strategy. But the method focused on groups with small sizes. In [28], McCarthy et al. used a variant of the Average Without Misery Strategy for this purpose. In [52], Yu et al. proposed a TV program recommendation strategy for groups of viewers based on a variant of the Average Strategy. In [50], Wang et al. developed a group recommender system which assigns each member a contribution score and aggregates members' profiles into group profile. In [33], Ortega et al. aggregated the preferences of members via the Average Strategy, and performed group recommendations using MF. Kaššák *et al.* [22] proposed a hybrid recommendation mechanism that combines content-based and collaborative strategies. In [36], Queiroz et al. suggested a method of making recommendations for groups through fuzzy majority. In [25], Lin et al. merged members' historical records as a group's historical records, and then proposed a ranking method to suggest recommendations. In [2], Baskin et al. presented a preference aggregation algorithm that search for a Kemeny-optimal ordering of items by aggregating users' relative preferences. In [20], Kagita et al. took all members' transitive precedence of items into consideration, and constructed a virtual user to represent that group. In [22], Kim et al. presented a recommender system which adopts graph-based approach to model relations between users and items. In [18], Hu et al. proposed a group recommendation approach which takes individual choices and group decisions in a joint model built with restricted Boltzmann machines. With such a deep model, high-dimensional features can be obtained to represent group preference. In [8], Cheng et al. abstracted group features into latent factors. Specifically, they proposed a novel recommendation model named GLFM, which attempts to learn a factorization of latent factor space into latent subspaces concerning behaviors.

C. LABEL DISTRIBUTION LEARNING

Conventionally, a learning process attempts to learn a mapping from examples to labels. There are two main ways about the mapping in existing learning schemes [55]:

- Single-label learning (SLL): One instance is associated with one label.
- Multiple-label learning (MLL): One instance is associated with not less than one label.

Recently, learning with ambiguity drew much attention. Label distribution learning (LDL) was developed for this purpose. LDL is a novel learning framework that assigns each instance a distribution of labels. In a label distribution, a real number $P_x^y \in [0, 1]$ is assigned to each label y, representing the degree that the corresponding label describes instance x. The sum of all the labels' values is 1, which means that an instance is fully described by labels. The goal of LDL is to learn a conditional probability density function (p.d.f.) with parameter set θ:

$$d_x^y = p\,(y|\boldsymbol{x}; \boldsymbol{\theta}) \tag{1}$$

where $\boldsymbol{\theta}$ is the set of all parameters.

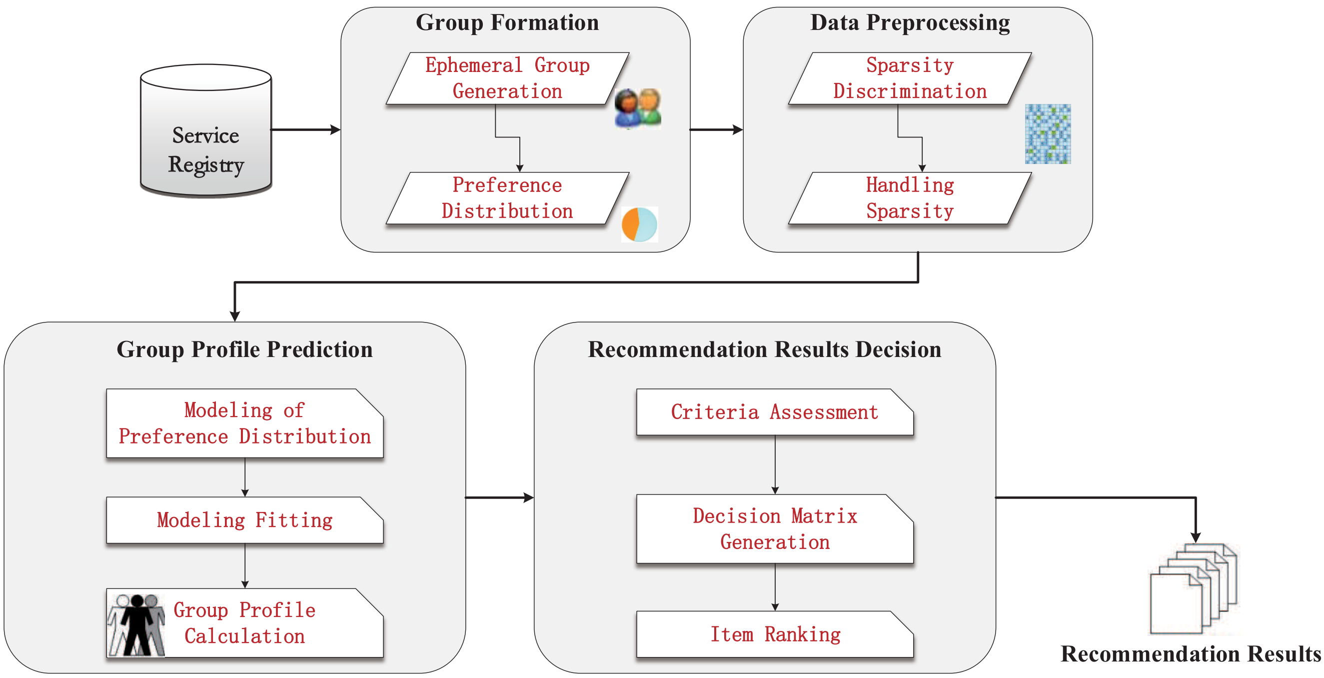

FIGURE 3. Infrastructure of Greption.

D. VIKOR METHOD

VIKOR was proposed for multi-criteria optimization problems [31], [32]. This method focuses on giving optimal ranking results from a set of candidate items under conflicting criteria. The final ranking results are obtained through balancing multiple criteria.

The main work of the paper is different from above works. Firstly, rather than simple aggregations, this paper regards group profile as distributions of preferences. Secondly, a scheme of generating recommendation results under preference distribution is proposed. Thirdly, the proposed mechanism is extended to be able to handle data sparsity.

III. GREPTION FRAMEWORK

In this section, we describe the proposed Greption framework shown in Fig. 3. To begin with, we formulate statement of research problem in this paper. Suppose that a number of users form a group G, and that a set of candidate items I_l $(l = 1, 2, \ldots, M)$ in which some of them will be selected for the group. And output is top-m recommendation results for the group, represented as $L_G = \{L_1, L_2, \ldots, L_m\}$. To achieve this, LDL is firstly introduced to formulate a model to predict group profile concerning candidate items I_l. Members' historical ratings towards items will be utilized to train the model. Items of training set are denoted as x_i $(i = 1, 2, \ldots, n)$. Then, a modified VIKOR method is proposed to generate recommendation results for the group.

The Greption framework consists of one service registry and four main modules: group formation, data preprocessing, group profile prediction, and recommendation results decision. Among them, service registry is a repository that stores users' historical rating records towards items, and is actually the source of training set in this research. Functionality of four main modules is described as follows:

A. GROUP FORMATION

We assume that service registry stores users' historical rating records towards items. In this module, it is supposed to form groups firstly. As we focus on recommendation for ephemeral groups in this paper, groups are formed by randomly selecting users, regardless of social relations among them. Then, it is expected to compute preference distribution of the group towards each item in service registry for training. Specifically, the preference distribution is members' rating distribution towards each item.

B. DATA PREPROCESSING

Our group recommendation mechanism is particularly designed based on the assumption that items of training set possess sufficient rating information from group members. Because calculation precision of preference distribution will be reduced if rating information is not sufficient. However, in real-world recommendations, a large number of items may have limited ratings from group members. Thus, it is significant to recover members' missing ratings towards items through various prediction means. And the Greption is also extended to be able to handle data sparsity. In this module, we firstly set a discrimination strategy to control the trigger of data recovery scheme. We assume that the data recovery scheme will be trigged according to the following rule:

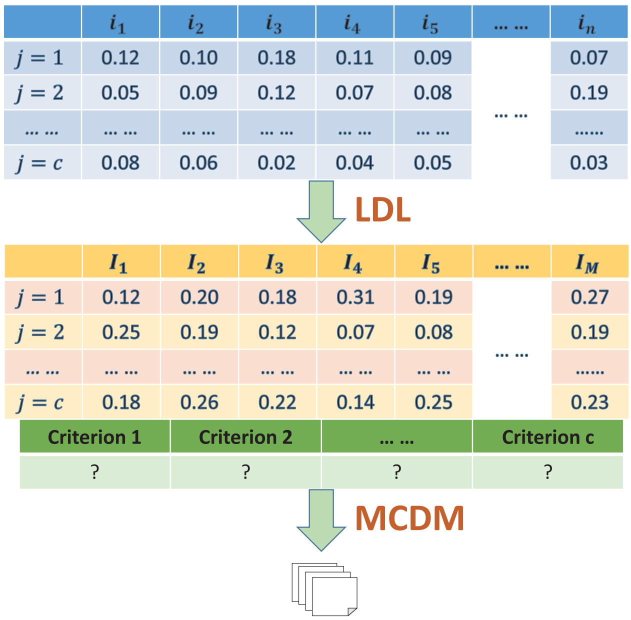

FIGURE 4. Complete sketch map of group profile prediction and recommendation results decision.

- The group size is denoted as $|G|$. For an item x_i, there should be $|G|$ members' ratings towards the item. If the number of missing ratings is not less than $|G|/3$, the data recovery scheme will be trigged.

Then, to recover missing rating information, we employ two schemes to handle this issue. The two schemes are described in Section IV-C. After processing of this module, final preference distributions concerning training set are obtained.

C. GROUP PROFILE PREDICTION

In this module, a learning strategy LDL is implemented. As depicted in upper part of Fig. 4, the preference distributions towards items x_i of service registry are firstly input as the training set. Then, a model with parameters that can output preference distributions is formulated. Next, with parameters of the model estimated, group profile concerning candidate items I_l can be calculated accordingly.

D. RECOMMENDATION RESULTS DECISION

Having modeled group profile, the aim of this stage is to select appropriate items for the group. As depicted in lower part of Fig. 4, predicted preference distributions of candidate items can be transformed into multi-criteria with criterion values respectively. Then, a modified VIKOR method incorporating fuzzy set theory is implemented to transform the item selection problem into an MCDM. After processing of this module, candidate items are ranked as recommendation results for the group.

IV. GROUP RECOMMENDATION MECHANISM

In this section, we present mathematical description of the proposed Greption approach. First, idea of LDL is introduced to finely model group profile concerning candidate items. Then, by proposing a modified VIKOR method, we transform the process of selecting items for a group into an MCDM. Furthermore, the Greption approach is extended to be able to handle data sparsity through two heuristic schemes: PA and CF.

A. MODELING OF GROUP PROFILE PREDICTION

As mentioned in Section 2.3, LDL is a novel learning framework that views each instance as a distribution of labels. And it is well suitable for modeling preference distributions.

1) PREFERENCE DISTRIBUTION MODELING

It is assumed that group members' rating distribution towards an item is viewed as a label distribution. We further assume that training set contains $|G|$ users as well as n items rated by users, and that the ratings given by users comprise c levels. Not all the items are rated by each user of the group. Let x_i denote items of training set, y_j $(j = 1, 2, \ldots, c)$ denote all the labels, and $P_i\left(y_j\right)$ denote preference distribution of x_i. The input space of LDL is obtained as:

$$S = \left\{\left(x_1, P_1\left(y_j\right)\right), \left(x_2, P_2\left(y_j\right)\right), \ldots, \left(x_n, P_n\left(y_j\right)\right)\right\},$$

Given above, the goal is to learn the group's preference distributions towards candidate items I_l.

Preference distribution of an instance x_i contains several components, denoting several dimensions of it. We utilize linear function to compute values of each dimension:

$$\boldsymbol{d} = f\left(\boldsymbol{x}\right) = \boldsymbol{W}\varphi\left(\boldsymbol{x}\right) + \boldsymbol{b} \tag{2}$$

where $\varphi\left(\boldsymbol{x}\right)$ denotes a transformation of $\boldsymbol{x}$ to a higher dimensional space of features $\mathbb{R}^{\mathcal{H}}$ ($\boldsymbol{W} \in \mathbb{R}^c$ and $\boldsymbol{b} \in \mathbb{R}^c$).

Next, in order to output distribution of the c labels, multi-dimensional regression analysis can be adopted to realize it. As for research problem in this paper, linear regression estimation problem can be regarded as finding the mapping between an item vector $\boldsymbol{x}$ and an label distribution vector y_j. General regressions cannot well deal with operations with high computational complexity. A more rational solution is to utilize multi-dimensional SVR [34], leading to solving the following optimization problem:

$$\min\left(\frac{1}{2}\sum_{j=1}^{c}\left\|\boldsymbol{w}^j\right\|^2 + \beta\sum_{i=1}^{n}L\left(\boldsymbol{h}_i\right)\right) \tag{3}$$

where $\boldsymbol{w}^j$ is the transpose of the j-th row of $\boldsymbol{W}$, $L\left(\boldsymbol{h}_i\right)$ is loss function of instance $\boldsymbol{x}_i$ and β is penalty parameter. The implicit meaning of Eq.(3) can be described as: the SVR finds the regressor $\boldsymbol{W}$ and $\boldsymbol{b}$ that minimizes the empirical error $\frac{1}{2}\sum_{j=1}^{c}\left\|\boldsymbol{w}^j\right\|^2 + \beta\sum_{i=1}^{n}L\left(\boldsymbol{h}_i\right)$. As for common multi-dimensional SVR, its loss function is depicted as L_1-norm:

$$L\left(\boldsymbol{h}_i\right) = \begin{cases} 0 & \boldsymbol{h}_i < \varepsilon \\ \boldsymbol{h}_i - \varepsilon & \boldsymbol{h}_i > \varepsilon \end{cases} \tag{4}$$

$$\boldsymbol{h}_i = \left|\boldsymbol{d}_i - \left(\boldsymbol{W}\varphi\left(\boldsymbol{x}\right) + \boldsymbol{b}\right)\right| \tag{5}$$

where $\boldsymbol{h}_i$ is a vector that represents the loss about c dimensions of the i-th item. But as pointed by [34], in multi-dimensional SVR, L_1-norm loss function will be calculated for each dimension independently, which will make computation complexity grow with the number of dimensions, thus loss function is expected to be formulated as L_2-norm for sake of simplicity:

$$L(\boldsymbol{h}_i) = \begin{cases} 0 & \boldsymbol{h}_i < \varepsilon \\ (\boldsymbol{h}_i - \varepsilon)^2 & \boldsymbol{h}_i \geq \varepsilon \end{cases} \tag{6}$$

$$\boldsymbol{h}_i = \|\boldsymbol{e}_i\| = \sqrt{\boldsymbol{e}_i{}^T \boldsymbol{e}_i} \tag{7}$$

$$\boldsymbol{e}_i = \boldsymbol{d}_i - (\boldsymbol{W}\varphi(\boldsymbol{x}) + \boldsymbol{b}) \tag{8}$$

where $\boldsymbol{d}_i$ denotes real label distribution of i-th instance, $\boldsymbol{W}\varphi(\boldsymbol{x}) + \boldsymbol{b}$ is the label distribution calculated by prediction model, and thus $\boldsymbol{e}_i$ denotes the empirical error. Note that $L(\boldsymbol{h}_i)$ is the Vapnik ε-insensitive loss function [4], [11] and $\boldsymbol{h}_i$ is the loss. It builds an abstract insensitive zone determined by ε around the estimate [49], meaning that loss less than insensitive parameter ε will be ignored. In order to predict preference distributions, it is expected to solve the optimization problem in Eq.(3). The following subsection is organized for this purpose.

2) OPTIMIZATION FOR MULTIDIMENSIONAL SVR

Optimization problems manage to iteratively update previous solutions in each iteration to search for the following ones until convergence. It is hard to directly solve the problem in Eq.(3). Instead, a quasi-Newton based solution IRWLS [35] is utilized to search for approximate solution. Firstly, first order Taylor expansion of loss function term in objective function in Eq.(3) is utilized to replace the loss function term as its approximation:

$$T'(\boldsymbol{W}, \boldsymbol{b}) = \frac{1}{2}\sum_{j=1}^{c}\left\|\boldsymbol{w}^j\right\|^2 + \beta\sum_{i=1}^{n} L\left(\boldsymbol{h}_i^{(g)}\right) + \beta\sum_{i=1}^{n}\frac{dL(h)}{dh}\Big|_{\boldsymbol{h}_i^{(g)}}\frac{\left(\boldsymbol{e}_i^{(g)}\right)^T\left(\boldsymbol{e}_i - \boldsymbol{e}_i^{(g)}\right)}{\boldsymbol{h}_i^{(g)}} \tag{9}$$

Submitting Eq.(7) into Eq.(9) leads to the following transformed formula:

$$\begin{aligned} T''(\boldsymbol{W}, \boldsymbol{b}) &= \frac{1}{2}\sum_{j=1}^{c}\left\|\boldsymbol{w}^j\right\|^2 + \beta\sum_{i=1}^{n} L\left(\boldsymbol{h}_i^{(g)}\right) \\ &\quad + \beta\sum_{i=1}^{n}\frac{dL(h)}{dh}\Big|_{h_i^{(g)}}\frac{\left(\boldsymbol{h}_i^2 - \left(\boldsymbol{h}_i^{(g)}\right)^2\right)}{\boldsymbol{h}_i^{(g)}} \\ &= \frac{1}{2}\sum_{j=1}^{c}\left\|\boldsymbol{w}^j\right\|^2 + \frac{1}{2}\sum_{i=1}^{n} a_i \boldsymbol{h}_i^2 + \xi \end{aligned} \tag{10}$$

where

$$a_i = \begin{cases} 0 & \boldsymbol{h}_i{}^{(g)} < \varepsilon \\ \dfrac{2\beta\left(\boldsymbol{h}_i{}^{(g)} - \varepsilon\right)}{\boldsymbol{h}_i{}^{(g)}} & \boldsymbol{h}_i{}^{(g)} \geq \varepsilon \end{cases} \tag{11}$$

and ξ is sum of constant terms never relying on $\boldsymbol{W}$ or $\boldsymbol{b}$. Finding minimum of Eq. (10) can be solved by letting the gradient equal to zero [35]. Due to the space limit, we leave out detailed derivation process of optimal solution finding. Finally, the optimal solutions $\boldsymbol{w}^j$ and b^j can be obtained and then substituted into Eq. (2) to calculate predicted preference distributions of candidate items I_l. So far, group profile concerning candidate items, denoted as $\bar{P}_l(y_j)$, is obtained. As all the predicted must follow the condition $\sum_{j=1}^{c}\bar{P}_l(y_j) = 1$, thus the final preference distribution need to be obtained by normalizing $\bar{P}_l(y_j)$:

$$P_l(y_j) = \frac{\bar{P}_l(y_j)}{\sum_{j=1}^{c}\bar{P}_l(y_j)} \tag{12}$$

B. DECISION OF RECOMMENDATION RESULTS

Given predicted group profile, the aim of this stage is to select appropriate items for the group. However, preference distributions are highly ambiguous, and are composed by some components. And components correspond to preference levels. To achieve this goal, we propose a modified VIKOR method incorporating fuzzy set theory, and transform the item selection process into an MCDM. To construct MCDM, following assumptions are established:

- Let I_l be the set of items to be recommended, that is, alternatives.
- For items I_l, they have group preference distribution $P_l(y_j)$ concerning c-level rating. Let y_j denote the criteria for decision and $P_l(y_j)$ denote its values.
- Preference distribution values of I_l corresponding to criteria y_j are called VC_{lj}.

Given candidate items I_l and predicted group profile $P_l(y_j)$, the objective of the modified VIKOR method is to find optimal ranking order for items I_l under c criteria.

1) CRITERIA ASSESSMENT

As c criteria possess different meanings, we firstly adopt fuzzy set theory to assess the nature of criteria. Here, the c criteria are divided into three clusters C_1, C_2, and C_3 with the aid of membership function:

$$y_j \in \begin{cases} C_1 & 0 < \tilde{\mu}_j \leq 0.3 \\ C_2 & 0.3 < \tilde{\mu}_j \leq 0.6 \\ C_3 & 0.6 < \tilde{\mu}_j \leq 1 \end{cases} \tag{13}$$

where

$$\tilde{\mu}_j = \frac{1}{2} + \frac{1}{2}\sin\frac{\pi}{c}\left(j - \frac{c}{2}\right), \quad j = 1, 2, \ldots, c \tag{14}$$

2) DECISION MATRIX GENERATION

A pseudo decision maker is simulated to assess criteria values of items I_l and to give decision scores for each criterion of items I_l. As for criteria of C_3, higher values signify utility. And for criteria of C_1, higher values imply dissatisfaction.

Their decision scores are assumed to range from 0 to 100. Yet criteria of C_2 are relatively neutral, their decision scores are not expected to be too high or too low, and are assumed to range between two positive integers q_1 and q_2. Therefore, decision score of criteria can be calculated as follows:

$$DS_{lj} = \begin{cases} 50 \cdot \left[\frac{2}{\pi} \arcsin\left(2VC_{lj} - 1\right) + 1 \right] & y_j \in C_3 \\ \left(1 - VC_{lj}\right) \cdot q_1 + VC_{lj} \cdot q_2 & y_j \in C_2 \\ 50 \cdot \left[\frac{2}{\pi} \arcsin\left(1 - 2VC_{lj}\right) + 1 \right] & y_j \in C_1 \end{cases} \tag{15}$$

where DS_{lj} denotes the decision score of j-th criterion of l-th item. Thus, decision matrix D of the pseudo decision maker can be constructed:

$$D = \begin{bmatrix} DS_{11} & DS_{12} & \cdots & DS_{1c} \\ DS_{21} & DS_{22} & \cdots & DS_{2c} \\ \vdots & \vdots & \ddots & \vdots \\ DS_{M1} & DS_{M2} & \cdots & DS_{Mc} \end{bmatrix} \tag{16}$$

3) ITEM RANKING

For all criteria, we denote the best value by F_j^* and the worst value by F_j^- as follows:

$$\begin{aligned} F_j^* &= \begin{cases} \max DS_{lj} & y_j \in C_3 \\ \min DS_{lj} & y_j \in (C_1 \cup C_2) \end{cases} \\ F_j^- &= \begin{cases} \min DS_{lj} & y_j \in C_3 \\ \max DS_{lj} & y_j \in (C_1 \cup C_2) \end{cases} \end{aligned} \tag{17}$$

We denote the aggregated value of l-th item with a maximum group utility by GU_l and compute it as following formula:

$$GU_l = \sum_{j=1}^{c} \eta_j \left(F_j^* - DS_{lj}\right) / \left(F_j^* - F_j^-\right) \tag{18}$$

We denote the aggregated value of l-th item with a minimum individual regret of "opponent" by IR_l and compute it as following formula:

$$IR_l = \max_{j=1,\ldots,c} \left[\eta_j \left(F_j^* - DS_{lj}\right) / \left(F_j^* - F_j^-\right) \right] \tag{19}$$

where η_j is the weight of criterion j, and is initially defined as:

$$\eta_j = \sum_{l=1}^{M} VC_{lj} \tag{20}$$

We denote value of benefit ratio for lth item by Q_l and compute it as following formula:

$$\begin{aligned} Q_l &= \lambda \left(GU_l - GU^*\right) / \left(GU^- - GU^*\right) \\ &\quad + (1 - \lambda) \left(IR_l - IR^*\right) / \left(IR^- - IR^*\right) \\ GU^* &= \min_{l=1,\ldots,M} GU_l, \; GU^- = \max_{l=1,\ldots,M} GU_l \\ IR^* &= \min_{l=1,\ldots,M} IR_l, \; IR^- = \max_{l=1,\ldots,M} IR_l \end{aligned} \tag{21}$$

where λ is a weight parameter for the strategies.

So far, all the items are ranked by sorting the values GU_l, IR_l, and Q_l respectively in ascending order, obtaining three ranking lists. The scheme with minimum value of Q is considered the optimal solution, if these two conditions are satisfied:

1) The formula:

$$Q\left(I^{\nabla}\right) - Q\left(I^{\Delta}\right) \geq 1/(M-1) \tag{22}$$

where I^{∇} is the second item in the ranking list by Q, I^{Δ} is the first item, and M is the number of items.

2) The item I^{Δ} must be also the best ranked by GU and/or IR.

C. HANDLING OF DATA SPARSITY

Our group recommendation mechanism is particularly designed based on the assumption that items of training set possess sufficient ratings from group members. Because calculation precision of preference distribution will be reduced if rating information is not sufficient. However, in real-world situations, a large fraction of items may have limited ratings from group members. To handle this issue, we extend the Greption approach by proposing two heuristic schemes—Preference Average (PA) and User-based Collaborative Filtering (CF) [40], as follows.

1) PREFERENCE AVERAGE

We assume that rating towards item k from user u is absent, and is to be predicted. Although user u lacks rating towards item k, we can simply calculate the average of her historical rating records towards other items:

$$\overline{R}_{uk} = \frac{\sum R_{uz}}{N_u} \tag{23}$$

where R_{uz} is user u's past rating records, N_u is the number of user u's past rating records, and $\overline{R}_{uk}$ is the estimated rating of user u towards item k. $\overline{R}_{uk}$ takes the integer.

2) COLLABORATIVE FILTERING

Another way to handle data sparsity is to apply information of other similar users. User-based Collaborative Filtering algorithm was proposed in 1994. And it has been widely used for infering unknown user preferences from similar users.

The first step is calculation of similarity. In general, similarity between two users u and v is measured by Pearson correlation coefficient [44] and computed as follows:

$$sim(u, v) = \frac{\sum_{k \in \widetilde{I}} \left(r_{v,k} - \overline{r}_v\right)\left(r_{u,k} - \overline{r}_u\right)}{\sqrt{\sum_{k \in \widetilde{I}} \left(r_{v,k} - \overline{r}_v\right)^2} \sqrt{\sum_{k \in \widetilde{I}} \left(r_{u,k} - \overline{r}_u\right)^2}} \tag{24}$$

where $\widetilde{I}$ is the set of items rated by both user u and v, $r_{u,k}$ denotes rating towards item k given by user u and $r_{v,k}$ denotes rating towards item k given by user v. The rating averages $\overline{r}_u$ and $\overline{r}_v$ are taken over the common items rated by both user u and user v.

After that, the unknown rating $\overline{R}_{uk}$ is predicted as follows:

$$\overline{R}_{uk} = \overline{r}_u + \frac{\sum \left(r_{v,k} - \overline{r}_v\right) \cdot sim(u, v)}{\sum |sim(u, v)|} \tag{25}$$

TABLE 1. Metadata included in the experimental dataset.

Attribute	Type	ϕ	Attribute	Type
Genre	Categorical	0	Year	Numerical
Director	Categorical	5	Running Time	Numerical
1st Actor	Categorical	5	Budget	Numerical
2nd Actor	Categorical	5		
Country	Categorical	10		
Language	Categorical	10		
Production Company	Categorical	20		

and we will compare performance of these two heuristic solutions in Section V.

V. EXPERIMENTS AND ANALYSIS

In this section, we conduct a set of experiments on two real-world datasets to evaluate the proposed approach. In Section V-A, we introduce basic assumptions and setup of our experiments. Section V-B presents relevant metrics and baselines for the experiments. We finally present results and analysis of the experiments in Section V-C.

A. EXPERIMENTAL SETUP

Compared with building pure simulation scenario via computer technology, we tend to set up experimental scenarios from real-world datasets. As far as we know, there are no special datasets that have been published for group recommendations. Therefore, we employ the "MovieLens 10M" dataset[1] (referred to as MovieLens) and the "Netflix" dataset[2] (referred to as Netflix), which are popular datasets when evaluating methods of individual recommendations. The MovieLens includes 10681 items and 10000054 ratings from 71567 different users, while the Netflix includes 17770 items and 100480507 ratings from 480507 different users. And the ratings of both datasets range from 1 to 5. It is assumed that items which have been assessed by members of groups will be filtered out of candidate items. Also, we select 65% of the data for training and 35% for testing.

Note that because these datasets do not contain information of group formation. Here we present group generating rules in this research. Our experiments attempts to construct ephemeral groups that are probable to have highly conflicting preferences. For this type of groups, two important features affect evaluation results—group size and social relations among members:

- For the former, group sizes are set to these levels: 300, 350 400, 450 and 500.
- For the latter, we randomly select users possessing no explicit shared preference relations to construct groups.

To extract features from items, metadata of the experimental datasets is obtained from IMDb.[3] Tab. 1 lists metadata used for the dataset, in which we divide all the attributes into two types: numeric attributes and categorical attributes. Some categorical attributes, such as names of actors, are unstructured data. In this experiment, we set a threshold ϕ to filter those who appear less times than the threshold value, in order to simplify the complexity of data. The categorical attributes are then changed into numerical ones by numbering each value of all the categorical attributes. Finally, normalizing all the attributes values to the same scale.

We set the parameter β of Eq. (3) to 1.0, the tolerance parameter ε in Eq. (15) to 0.12. q_1 and q_2 in Eq. (16) are set to 15 and 85 respectively, and the weight λ in Eq. (21) is set to 0.5. As for the two datasets, the number of labels is $c = 5$.

B. METRICS AND BASELINES

In our experiments, we evaluate performance of the Greption through efficiency of recommendation lists. We adopt several metrics that have been utilized in some relevant researches: nDCG, MRR@m, MAP@m.

nDCG refers to normalized discounted cumulative gain. It has been widely used in assessment of recommendation algorithms. Detailed explanations of nDCG measure are described in [50], and are left out here.

MRR refers to mean reciprocal rank. It is a universal ranking metric to measure recommendation quality by finding out how far from the top of the list the first successfully predicted item is (averaged over all test cases). Detailed descriptions are described in [10]. MRR for a group is the average of all the group members', denoting global rank efficiency. And MRR@m denotes value of MRR while making top-m recommendation. In our experiments, we set m as 5 and 10 for evaluation.

MAP refers to mean average precision. It is a popular metric to measure precision of retrieval by finding average precision of retrieved items. Detailed explanations are described in [27]. MAP for a group is the average of all the group members'. And MAP@m denotes value of MAP while making top-m recommendation. In our experiments, we set m as 5 and 10 for evaluation.

As for baselines for comparison, we select the following classical approaches about group recommendation as baselines:

- LM [29]—The group preference is viewed as forms of single values, and is generated through the least misery strategy.
- AVG [29]—The group preference is viewed as forms of single values, and is generated through the average strategy.

[1] http://www.grouplens.org/
[2] http://www.netflixprize.com/
[3] http://www.imdb.com/

- AM [28]—the group preference is generated through the average without misery strategy. In this aggregation strategy, A threshold is used to filter out items that will cause disappointment for members. In our experiment, this threshold is set to rating of 2.
- MCS [50]—The group profile is also viewed as forms of single values, and is established with a newly proposed concept weighted individual preference, in which weights are computed for each member.

To further evaluate performance of handling data sparsity, we denote pure Grepation approach without any schemes to handle data sparsity as **Gre**. The Gre is able to represent the proposed Grepation approach. And we also define two variants of Grepation approach. **Gre+PA** is a variant of Grepation with employing PA to handle data sparsity. **Gre+CF** is another variant of Grepation with employing user-based CF to handle data sparsity.

C. RESULTS AND ANALYSIS

In this subsection, our experiments can be divided into two parts: performance evaluation of pure Grepation approach (Gre), and performance evaluation of handling data sparsity. For the former, the Gre is compared with baselines using both two datasets with respect to metrics: nDCG, MAP@5, MAP@10, MRR@5, and MRR@10. For the latter, Gre is compared with two variants Gre+PA and Gre+CF on MovieLens with respect to same metrics.

Fig. 5 shows the obtained nDCG results on MovieLens with different group sizes, in which Fig. 5(a) corresponds to group size of 300, Fig. 5(b) corresponds to group size of 400, and Fig. 5(c) corresponds to group size of 500. The X-axis indicates the number of top-m items that will be selected after ranking, while the Y-axis represents the values of metrics. It can be seen from each sub-figure that the recent proposed MCS performs better than conventional methods like LM, AVG, and AM, regardless of group size. Obviously, simple aggregations cannot well reflect comprehensive group profile. However, performance of MCS is not comparable to the Gre. Take top-5 recommendation as an example, the Gre is about 11% better than MCS with group size of 300, about 9% better than MCS with group size of 400, and about 10% better than MCS with group size of 500. Take top-8 recommendation as another example, the Gre is about 3% better than MCS with group size of 300, about 2% better than MCS with group size of 400, and about 5% better than MCS with group size of 500. The same experiments are also conducted on Netflix who possesses richer ''user-item'' information. Fig. 6 shows the obtained nDCG results on Netflix with different group sizes, in which Fig. 6(a) corresponds to group size of 300, Fig. 5(b) corresponds to group size of 400, and Fig. 5(c) corresponds to group size of 500. The X-axis indicates the number of top-m items that will be selected after ranking, while the Y-axis represents the values of metrics. From this set of results, it is easy to find that MCS also performs better than conventional methods like LM, AVG, and AM

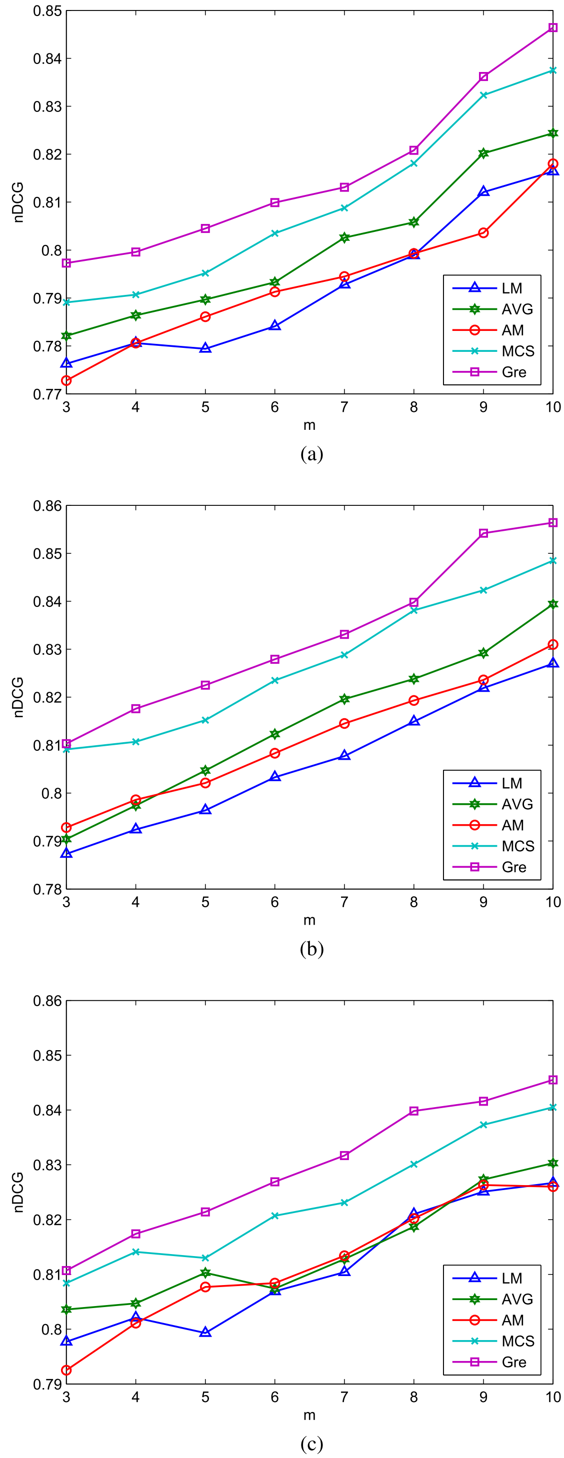

FIGURE 5. nDCG results on "MovieLens" dataset with different group sizes respectively. (a) Group size: 300. (b) Group size: 400. (c) Group size: 500.

with any group size. This can also prove the inefficiency of simple aggregation-based methods. Furthermore, the Gre exceeds MCS to some extent. Take top-5 recommendation as an example, the Gre is about 6% better than MCS with group size of 300, about 6% better than MCS with group size

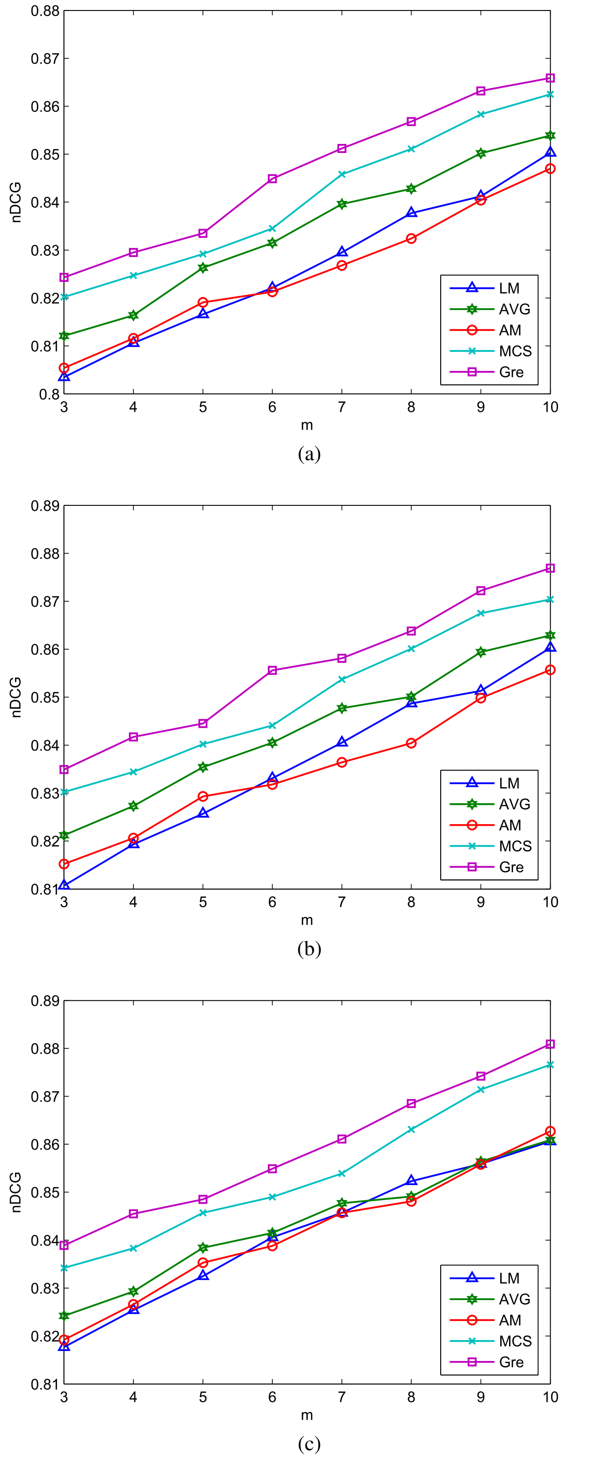

FIGURE 6. nDCG results on "Netflix" dataset with different group sizes respectively. (a) Group size: 300. (b) Group size: 400. (c) Group size: 500.

of 400, and about 5% better than MCS with group size of 500. Take top-8 recommendation as another example, the Gre is about 7% better than MCS with group size of 300, about 5% better than MCS with group size of 400, and about 7% better than MCS with group size of 500. Viewing group profile as preference distribution rather than forms of single values will certainly be able to capture more comprehensive preference features for a group. It is established from a global insight. While regarding group profile as forms of single values can only represent preferences of some members, not all the members. It is established from a local insight. Thus no matter which type of aggregation method is utilized, viewing group profile as forms of single values cannot well reflect comprehensive group profile. Instead, the Greption approach models group profile from the perspective of preference distribution, and can well overcome this issue. Therefore, above experimental results show that the proposed Greption approach performs better than baselines regarding metric nDCG.

Tab. 2 shows the experimental results on two datasets with group size of 350, in which Gre is compared with baselines concerning metrics: MAP@5, MAP@10, MRR@5, MRR@10. Left part of the table illustrates experimental results on MovieLens, while the right part shows experimental results on Netflix. And the experimental results are average of top-3, top-5, and top-8 recommendations. The first column of the table lists baselines and the Gre, while other columns reveal values of different metrics. It can be seen from the table that MCS performs better than conventional methods like LM, AVG, and AM, yet cannot compare with Gre. As for MovieLens, Gre is about 17% better than MCS with respect to MAP@5 and MAP@10, and about 9% better than MCS concerning MRR@5 and MRR@10. As for Netflix, Gre is about 19% better than MCS with respect to MAP@5 and MAP@10, and about 12% better than MCS concerning MRR@5 and MRR@10. Tab. 3 shows the experimental results on two datasets with group size of 450, in which Gre is compared with baselines concerning metrics: MAP@5, MAP@10, MRR@5, MRR@10. Basic setups are the same with the above series of experiments. Left part of the table illustrates experimental results on MovieLens, while the right part shows experimental results on Netflix. And the experimental results are also average of top-3, top-5, and top-8 recommendations. The first column of the table lists baselines and the Gre, while other columns reveal values of different metrics. It can be seen from the table that MCS performs better than conventional methods like LM, AVG, and AM, yet cannot compare with Gre. As for MovieLens, Gre is about 18% better than MCS with respect to MAP@5 and MAP@10, and about 9% better than MCS concerning MRR@5 and MRR@10. As for Netflix, Gre is about 19% better than MCS with respect to MAP@5 and MAP@10, and about 8% better than MCS concerning MRR@5 and MRR@10. Results in the two tables also proves that modeling group profile from the perspective of preference distribution is able to well represent conflicting characteristics of group preferences. Because simple aggregation-based methods almost model group profile from a local view. In these methods, forms of group profile cannot cover preferences of all the members. They cannot well capture comprehensive preference features of groups. yet the Greption models group profile from a global view. In Greption, forms of group profile contain complete preference characteristics of

TABLE 2. Experimental results of Gre and baselines concerning metrics: MAP@5, MAP@10, MRR@5, MRR@10 (Group size: 350).

Method	"MovieLens" dataset				"Netflix" dataset			
	MAP@5	MAP@10	MRR@5	MRR@10	MAP@5	MAP@10	MRR@5	MRR@10
LM	0.487	0.461	0.522	0.555	0.511	0.497	0.539	0.561
AVG	0.503	0.492	0.558	0.576	0.516	0.511	0.562	0.581
AM	0.464	0.437	0.509	0.525	0.470	0.445	0.518	0.533
MCS	0.571	0.549	0.608	0.596	0.582	0.557	0.610	0.609
Gre	**0.657**	**0.649**	**0.653**	**0.658**	**0.679**	**0.673**	**0.671**	**0.680**

TABLE 3. Experimental results of Gre and baselines concerning metrics: MAP@5, MAP@10, MRR@5, MRR@10 (Group size: 450).

Method	"MovieLens" dataset				"Netflix" dataset			
	MAP@5	MAP@10	MRR@5	MRR@10	MAP@5	MAP@10	MRR@5	MRR@10
LM	0.472	0.459	0.536	0.571	0.485	0.474	0.551	0.578
AVG	0.510	0.487	0.544	0.563	0.527	0.496	0.553	0.568
AM	0.451	0.437	0.509	0.525	0.469	0.450	0.522	0.547
MCS	0.567	0.543	0.615	0.592	0.581	0.560	0.623	0.607
Gre	**0.671**	**0.659**	**0.660**	**0.643**	**0.692**	**0.684**	**0.675**	**0.656**

group members. Therefore, the Greption provides much improvement in this aspect. Overall, experiments on both two datasets show that the proposed Greption approach performs better than baselines regarding these metrics: MAP@5, MAP@10, MRR@5, MRR@10. It can be also discovered from the two tables that when group size is increasing, performance of baselines show a descending trend. Specifically, most values of metrics of LM, AVG, AM, and MCS are declining with the group size growing. However, relevant values of Gre overall present an ascending trend. This can be attributed as two aspects of reasons. For one thing, Gre, viewing group profile from the perspective of preference distribution, is able to tackle with large-scale groups better than others. For another, a modified VIKOR method is proposed to decide the ranking of items in Gre, which contributes to optimizing recommendation results as well as ranking positions compared with baselines.

Also, we evaluate the performance of two schemes for handling data sparsity, in which Gre is compared with Gre+PA and Gre+CF. Among the two datasets utilized in our experiments, Netflix dataset is rich with user-item information, while MovieLens is quite sparse. Thus, we evaluate performance of handling data sparsity with experiments conducted on MovieLens. Fig. 7 shows the obtained nDCG results on MovieLens with different group sizes, in which Fig. 7(a) corresponds to group size of 300, Fig. 7(b) corresponds to group size of 400, and Fig. 7(c) corresponds to group size of 500. The X-axis indicates the number of top-*m* items that will be selected after ranking, while the Y-axis represents values of metrics. It can be seen from each sub-figure that performance of Gre+PA is below the Gre, regardless of group sizes. Obviously, simple average-based schemes cannot well recover missing information. However, performance of Gre+CF is better than Gre. Take top-5 recommendation as an example, the Gre+CF is about 8% better than Gre with group size of 300, about 7% better than Gre with group size of 400, and about 10% better than Gre with group size of 500. Take top-8 recommendation as another example, Gre+CF is about 16% better than Gre with group size of 300, about 7% better than Gre with group size of 400, and 9% better than Gre with group size of 500. Fig. 8 shows the experimental results on MovieLens with group sizes of 350 and 450, in which performance of Gre is compared with Gre+PA and Gre+CF concerning metrics: MAP@5, MAP@10, MRR@5, MRR@10. The X-axis indicates those metrics that will be used for evaluation, while the Y-axis represents the values of metrics. And the results are average of top-3, top-5, and top-8 recommendation. It can be also seen from each sub-figure that performance of Gre+PA is below the pure Greption approach Gre, regardless of group size. Obviously, simple average-based schemes cannot well recover missing information. Because the method just views unknown preference information as average of historical records. But average numbers never analyze internal rules and cannot represent all, thus average-based methods have quite strong limitations. However, performance of Gre+CF is better than the Gre. Because when predicting unknown preference through CF, CF analyzes behavioral features of all the users and searches

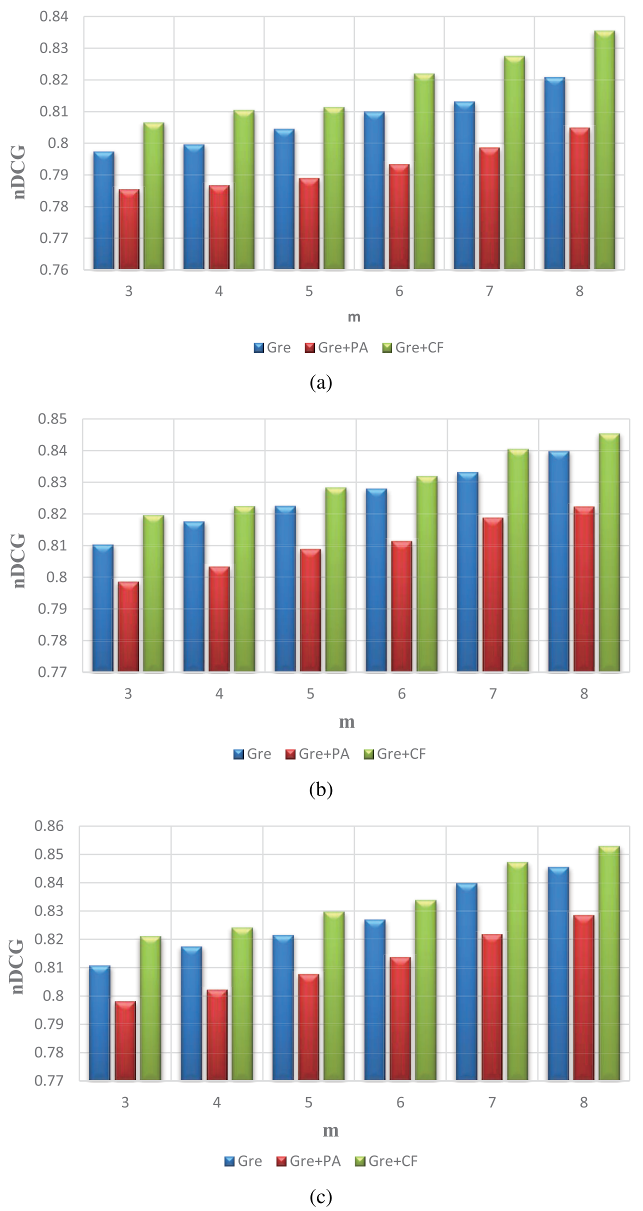

FIGURE 7. Evaluating schemes of handling sparsity concerning nDCG with different group sizes respectively. (a) Group size: 300. (b) Group size: 400. (c) Group size: 500.

for the most similar user relative to objective users. On the basis of similar neighbors, preferences of objective users can be infered. This method to complete unknown preference information is more rational and more probable to obtain relatively high prediction precision. As for MAP@5, Gre+CF is about 1.2% better than Gre with group size of 350, and about 1.7% better than Gre with group size of 450. As for MRR@5, Gre+CF is about 1.7% better than Gre with group size of 350, and about 1.4% better than Gre with group size of 450. Obviously, Gre+CF is better, yet just produces a little improvement compared with Gre. Furthermore, performance of Gre+PA is not below Gre too much. This is because the view of preference distribution concerning large-scale groups

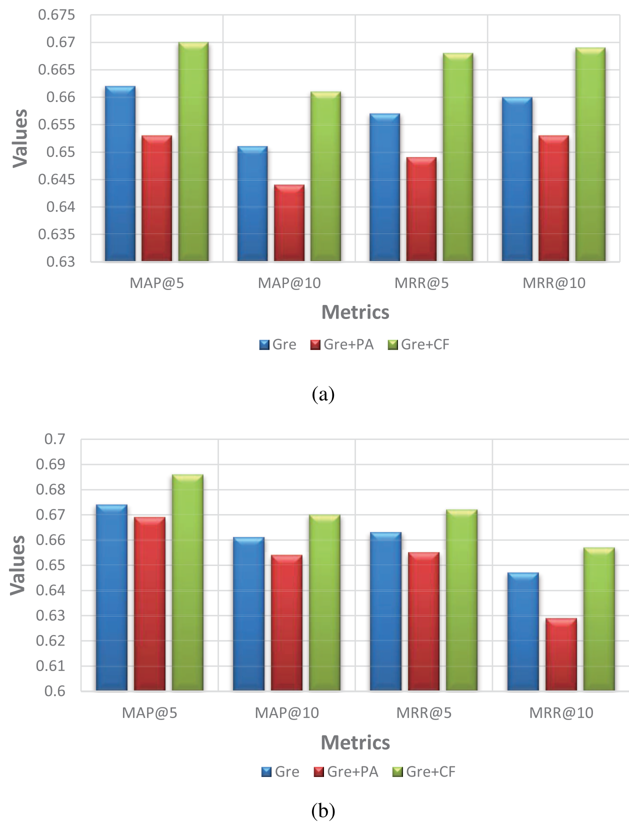

FIGURE 8. Evaluating schemes of handling sparsity concerning several metrics with different group sizes of 350 and 450. (a) Group size: 350. (b) Group size: 450.

possesses really strong robustness. Specially, the proposed method views group profile from an insight of global feature distribution that will not be influenced by small amount of deviation.

In all, experimental results described in this section can well prove that Greption can make recommendations for a group of users better than prior works. It is superior to the previous for the following aspects. First, it considers preference distribution instead of simple aggregation of members' preferences, which is able to finely model group profile. Second, a modified VIKOR method is developed to view the decision of recommendation results as an MCDM, which optimizes the recommendation results. Third, it can be extended with respect to handling data sparsity, which can further improve performance of Greption.

VI. CONCLUSIONS

This paper focuses on recommending items to group of users rather than individual user. To model group profile, existing researches almost aggregate preferences of members into forms of single values, and thus cannot reflect comprehensive group profile. Therefore, we propose a novel group recommendation mechanism. It views group profile as preference distributions, and then transorms item recommendation process into an MCDM. Futhermore, the approach is extended to be able to tackle with sparse data. Finally, experiments are conducted to verify the proposed approach. As for groups whose members have relatively similar preference features, aggregation-based methods are suitable solutions. Because they can achieve both ideal precision as well as rapid running speed. But when facing groups with highly conflicting preference characteristics, the proposed approach Greption will perform better, because it can reflect more comprehensive group profile, which exceeds those based on aggregations. The research problem in this paper mainly focuses on large-scale groups. This type of groups usually consist of members with heterogeneous preference features. Therefore, the proposed Greption is well suitable for scenarios in this paper.

In the future, we plan to carry out more similar datasets for massive experiments to verify the applicability of the proposed Greption approach. What's more, we also plan to further research GRSs targeting persistent groups.

REFERENCES

[1] L. Ardissono *et al.*, "Tailoring the recommendation of tourist information to heterogeneous user groups," in *Proc. Workshop Adaptive Hypermedia*, 2001, pp. 280–295.

[2] J. P. Baskin and S. Krishnamurthi, "Preference aggregation in group recommender systems for committee decision-making," in *Proc. 3rd ACM Conf. Recommender Syst.*, 2009, pp. 337–340.

[3] J. Bento, S. Ioannidis, S. Muthukrishnan, and J. Yan, "Group recommendations via multi-armed bandits," in *Proc. ACM 21st Int. Conf. World Wide Web*, 2012, pp. 463–464.

[4] X. Bi *et al.*, "An advanced partial discharge recognition strategy of power cable," *J. Elect. Comput. Eng.*, vol. 2015, 2015, Art. no. 48.

[5] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, Jul. 2013.

[6] L. Boratto and S. Carta, "State-of-the-art in group recommendation and new approaches for automatic identification of groups," in *Information Retrieval and Mining in Distributed Environments*. Berlin, Germany: Springer, 2010, pp. 1–20.

[7] Y.-L. Chen, L.-C. Cheng, and C.-N. Chuang, "A group recommendation system with consideration of interactions among group members," *Expert Syst. Appl.*, vol. 34, no. 3, pp. 2082–2090, 2008.

[8] J. Cheng, T. Yuan, J. Wang, and H. Lu, "Group latent factor model for recommendation with multiple user behaviors," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2014, pp. 995–998.

[9] I. Christensen, S. Schiaffino, and M. Armentano, "Social group recommendation in the tourism domain," *J. Intell. Inf. Syst.*, vol. 47, no. 2, pp. 209–231, 2016.

[10] N. Craswell, "Mean reciprocal rank," in *Encyclopedia of Database Systems*. New York, NY, USA: Springer, 2009, p. 1703.

[11] H. J. Dai, P. T. Lai, and R. T. Tsai, "Multistage gene normalization and SVM-based ranking for protein interactor extraction in full-text articles," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 7, no. 3, pp. 412–420, Jul. 2010.

[12] S. Feng and J. Cao, "Improving group recommendations via detecting comprehensive correlative information," *Multimedia Tools Appl.*, vol. 76, no. 1, pp. 1355–1377, 2017.

[13] M. Gartrell *et al.*, "Enhancing group recommendation by incorporating social relationship interactions," in *Proc. 16th ACM Int. Conf. Supporting Group Work*, 2010, pp. 97–106.

[14] X. Geng, "Label distribution learning," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 7, pp. 1734–1748, Jul. 2016.

[15] X. Geng and R. Ji, "Label distribution learning," in *Proc. IEEE 13th Int. Conf. Data Mining Workshops*, Dec. 2013, pp. 377–383.

[16] Z. Guo *et al.*, "Fine-grained recommendation mechanism to curb astroturfing in crowdsourcing systems," *IEEE Access*, vol. 5, pp. 15529–15541, 2017.

[17] M. Hong, J. J. Jung, and D. Camacho, "GRSAT: A novel method on group recommendation by social affinity and trustworthiness," *Cybern. Syst.*, vol. 48, no. 3, pp. 140–161, 2017.

[18] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and W. Cao, "Deep modeling of group preferences for group-based recommendation," in *Proc. AAAI*, vol. 14. 2014, pp. 1861–1867.

[19] A. Jameson, "More than the sum of its members: Challenges for group recommender systems," in *Proc. ACM Work. Conf. Adv. Vis. Interfaces*, 2004, pp. 48–54.

[20] V. R. Kagita, A. K. Pujari, and V. Padmanabhan, "Virtual user approach for group recommender systems using precedence relations," *Inf. Sci.*, vol. 294, pp. 15–30, Feb. 2015.

[21] O. Kaššák, M. Kompan, and M. Bieliková, "Personalized hybrid recommendation for group of users: Top-N multimedia recommender," *Inf. Process. Manage.*, vol. 52, no. 3, pp. 459–477, 2016.

[22] H.-N. Kim, M. Bloess, and A. El Saddik, "Folkommender: A group recommender system based on a graph-based ranking algorithm," *Multimedia Syst.*, vol. 19, no. 6, pp. 509–525, 2013.

[23] H.-N. Kim and A. El Saddik, "A stochastic approach to group recommendations in social media systems," *Inf. Syst.*, vol. 50, pp. 76–93, Jun. 2015.

[24] M. Kompan and M. Bielikova, "Group recommendations: Survey and perspectives," *Comput. Inform.*, vol. 33, no. 2, pp. 446–476, 2014.

[25] K.-H. Lin, D.-H. Shiue, Y.-S. Chiu, W.-T. Tsai, F.-J. Jang, and J.-S. Chen, "Design and implementation of face recognition-aided IPTV adaptive group recommendation system based on NLMS algorithm," in *Proc. IEEE Int. Symp. Commun. Inf. Technol. (ISCIT)*, Oct. 2012, pp. 626–631.

[26] X. Liu, Y. Tian, M. Ye, and W. C. Lee, "Exploring personal impact for group recommendation," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage.*, 2012, pp. 674–683.

[27] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[28] J. F. McCarthy and T. D. Anagnost, "MusicFX: An arbiter of group preferences for computer supported collaborative workouts," in *Proc. ACM Conf. Comput. Supported Cooperat. Work*, 1998, pp. 363–372.

[29] M. O'Connor, D. Cosley, J. A. Konstan, and J. Riedl, "PolyLens: A recommender system for groups of users," in *Proc. ECSCW*, 2001, pp. 199–218.

[30] J. Oh, S. Kim, J. Kim, and H. Yu, "When to recommend: A new issue on TV show recommendation," *Inf. Sci.*, vol. 280, pp. 261–274, Oct. 2014.

[31] S. Opricovic and G.-H. Tzeng, "Multicriteria planning of post-earthquake sustainable reconstruction," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 17, no. 3, pp. 211–220, 2002.

[32] S. Opricovic and G.-H. Tzeng, "Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS," *Eur. J. Oper. Res.*, vol. 156, no. 2, pp. 445–455, 2004.

[33] F. Ortega, A. Hernando, J. Bobadilla, and J. H. Kang, "Recommending items to group of users using matrix factorization based collaborative filtering," *Inf. Sci.*, vol. 345, pp. 313–324, Jun. 2016.

[34] F. Pérez-Cruz, G. Camps-Valls, E. Soria-Olivas, J. J. Pérez-Ruixo, A. R. Figueiras-Vidal, and A. Artés-Rodríguez, "Multi-dimensional function approximation and regression estimation," in *Proc. Int. Conf. Artif. Neural Netw.*, 2002, pp. 757–762.

[35] F. Pérez-Cruz, A Navia-Vázquez, P. L. Alarcón-Diana, and A. Artes-Rodriguez, "An IRWLS procedure for SVR," in *Proc. IEEE 10th Eur. Signal Process. Conf.*, Sep. 2000, pp. 1–4.

[36] S. R. de M. Queiroz, F. D. A. T. de Carvalho, G. L. Ramalho, and V. Corruble, "Making recommendations for groups using collaborative filtering and fuzzy majority," in *Advances in Artificial Intelligence*. Berlin, Germany: Springer, 2002, pp. 248–258.

[37] L. Quijano-Sánchez, J. A. Recio-Garcia, and B. Díaz-Agudo, "Using personality to create alliances in group recommender systems," in *Proc. Int. Conf. Case-Based Reasoning*, 2011, pp. 226–240.

[38] L. Quijano-Sánchez, J. A. Recio-Garcia, and B. Díaz-Agudo, "An architecture and functional description to integrate social behaviour knowledge into group recommender systems," *Appl. Intell.*, vol. 40, no. 4, pp. 732–748, 2014.

[39] J. A. Recio-Garcia, G. Jimenez-Diaz, A. A. Sanchez-Ruiz, and B. Díaz-Agudo, "Personality aware recommendations to groups," in *Proc. 3rd ACM Conf. Recommender Syst.*, 2009, pp. 325–328.

[40] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooperat. Work*, 1994, pp. 175–186.

[41] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender Systems Handbook*. New York, NY, USA: Springer, 2011, pp. 1–35.

[42] P. B. Kantor, *Recommender Systems Handbook*. Berlin, Germany: Springer, 2015.
[43] A. Salehi-Abari and C. Boutilier, "Preference-oriented social networks: Group recommendation and inference," in *Proc. 9th ACM Conf. Recommender Syst.*, 2015, pp. 35–42.
[44] P. Sedgwick *et al.*, "Pearson's correlation coefficient," *BMJ*, vol. 345, no. 7, pp. 1–2, 2012.
[45] S. Seko, T. Yagi, M. Motegi, and S. Muto, "Group recommendation using feature space representing behavioral tendency and power balance among members," in *Proc. 5th ACM Conf. Recommender Syst.*, 2011, pp. 101–108.
[46] P. Skowron, P. Faliszewski, and J. Lang, "Finding a collective set of items: From proportional multirepresentation to group recommendation," in *Proc. 29th AAAI Conf. Artif. Intell.*, vol. 241. 2016, pp. 191–193.
[47] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," in *Adv. Artif. Intell.*, vol. 2009, Aug. 2009, Art. no. 421425.
[48] Á. Tejeda-Lorente, C. Porcel, E. Peis, R. Sanz, and E. Herrera-Viedma, "A quality based recommender system to disseminate information in a University digital library," *Inf. Sci.*, vol. 261, pp. 52–69, Mar. 2014.
[49] V. Vapnik, S. E. Golowich, and A. J. Smola, "Support vector method for function approximation, regression estimation and signal processing," in *Proc. Adv. Neural Inf. Process. Syst.*, 1997, pp. 281–287.
[50] W. Wang, G. Zhang, and J. Lu, "Member contribution-based group recommender system," *Decision Support Syst.*, vol. 87, pp. 80–93, Jul. 2016.
[51] K. Yoshii, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, "An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model," *IEEE Trans. Audio, Speech, Language Process.*, vol. 16, no. 2, pp. 435–447, Feb. 2008.
[52] Z. Yu, X. Zhou, Y. Hao, and J. Gu, "TV program recommendation for multiple viewers based on user profile merging," *User Model. User-Adapted Interaction*, vol. 16, no. 1, pp. 63–82, 2006.
[53] Q. Yuan, G. Cong, and C. Lin, "COM: A generative model for group recommendation," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 163–172.
[54] L. Zhang, R. Zhou, H. Jiang, H. Wang, and Y. Zhang, "Item group recommendation: A method based on game theory," in *Proc. 26th Int. Conf. World Wide Web Companion*, 2017, pp. 1405–1411.
[55] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, Aug. 2014.

ZHIWEI GUO received the B.S. degree from Zhengzhou University in 2013. He is currently pursuing the Ph.D. degree with Chongqing University. Since 2014, he has been involved in data mining and machine learning. His current research interests include recommender system and neutral networks.

CHAOWEI TANG received the B.Eng. and M.Eng. degrees in communication and information system and the Ph.D. degree in electromechanical integration technology from Chongqing University in 1987, 1990, and 1993, respectively. He is currently a Professor with the College of Communication Engineering, Chongqing University. His research interests include intelligent information processing, computer vision, machine learning, and future network.

HUI TANG received the B.S. degree from Lanzhou University in 1992, the M.S. degree from the Institute of Computing Technology, Chinese Academy of Sciences, in 1995, and the Ph.D. degree from the Institute of Acoustics, Chinese Academy of Sciences, in 1998. Since 2004, he has been the Founding Director of the High Performance Network Laboratory, Institute of Acoustics, Chinese Academy of Sciences. And he is currently an Adjunct Professor with Chongqing University. His research interest include next-generation broadband wireless mobile network.

YUNQING FU received the B.S., M.S., and Ph.D. degrees in computer software and theory from Chongqing University in 1992, 1995, and 2001, respectively. He is currently a Professor with the College of Software, Chongqing University. His research interests include intelligent information processing, computer networks, and Internet of Things.

WENJIA NIU received the B.S. degree from Beijing Jiaotong University in 2005 and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences. Since 2017, he has been a Professor and a Doctoral Supervisor with Beijing Jiaotong University. His research interests include computer networks, pattern recognition, data mining, and network security.