Group Preference Aggregation: A Nash Equilibrium Approach

Hongke Zhao[§†], Qi Liu[§*], Yong Ge[†], Ruoyan Kong[‡], Enhong Chen[§]
[§]School of Computer Science and Technology, University of Science and Technology of China
zhhk@mail.ustc.edu.cn, {qiliuql, cheneh}@ustc.edu.cn
[†]Eller College of Management, University of Arizona, yongge@email.arizona.edu
[‡]School of Economics and Management, Tsinghua University, kry16@mails.tsinghua.edu.cn

***Abstract*—Group-oriented services such as group recommendations aim to provide services for a group of users. For these applications, how to aggregate the preferences of different group members is the toughest yet most important problem. Inspired by game theory, in this paper, we propose to explore the idea of *Nash equilibrium* to simulate the selections of members in a group by a game process. Along this line, we first compute the preferences (group-dependent optimal selections) of each individual member in a given group scene, i.e., an equilibrium solution of this group, with the help of two pruning approaches. Then, to get the aggregated unitary preference of each group from all group members, we design a matrix factorization-based method which aggregates the preferences in latent space and estimates the final group preference in rating space. After obtaining the group preference, group-oriented services (e.g., group recommendation) can be directly provided. Finally, we construct extensive experiments on two real-world data sets from multiple aspects. The results clearly demonstrate the effectiveness of our method.**

***Keywords*-Preference Aggregation, Group Recommendation, Nash Equilibrium**

I. Introduction

In our daily life, people often participate in some activities with others or as group members, e.g., watching movies with spouses, traveling with friends and having picnics with families. To facilitate these group activities, some Internet-based group services are emerging, e.g., group recommendation [1]–[4] and group buying [5]–[7].

Different from the individual-oriented services, there are some unique practical challenges in group-oriented services because we need to consider the preferences of all group members simultaneously. Indeed, the most difficult challenge is how to aggregate the preferences of different group members. Though some efforts on preference aggregation have been conducted, such as *preference aggregation* [8] and *score aggregation* [9], [10], the interactions and fairness of group members are still largely ignored. Therefore, these aggregation approaches, which are unable to figure out the optimal selections that can be accepted by all members in a group, may lead to unsatisfying services.

Inspired by *Nash equilibrium* [11], [12] from game theory, in this paper, we propose a focused study to explore the preferences of group members thoroughly. In fact, when planning a consumption decision in a group, each member should measure her own preference with other members' and then make a tradeoff. According to the definition of Nash equilibrium, all members in a group can get the acceptable benefits from the equilibrium solution, and only the equilibrium solution can satisfy all members at the same time [11], [13]. In other words, game process could capture the group members' interactions and Nash equilibrium solution considers the fairness as much as possible. Thus, we first propose to learn the individual preferences of members in a given group scene by simulating the group members' selections in this group with a game process and finding the Nash equilibrium solution for this group. To this end, for each group member, we first compute her *strategies* (optimal selections/items) with the help of two pruning methods (i.e., Skyline and Skyband pruning). Then, we can generate a *strategy profile set* (the *Cartesian product* of all members' strategies) for each group. For measuring the gain/utility from each group member to every strategy profile (i.e., one combination of the preferred items) in this strategy profile set, we design a *payoff function* which considers both the gains of given member and the acceptance possibilities of other members in a group. Based on the payoff gains of group members to each strategy profile, we can get the Nash equilibrium solution.

Unfortunately, Nash solution only stores the optimal selection probabilities from each group member for different strategies/items in the group scene, which is actually the individuals' group-dependent optimal selections. For group-oriented services (which provide the same service/item to all members in a group), we have to figure out the specific *ideal item* that can perfectly match the preferences of all group members. Thus, we have a second step to aggregate a determined unitary preference for each group by integrating the members' individual preferences and their group-dependent selections (i.e., Nash equilibrium solution). To this end, we further design a matrix factorization-based method for this aggregation. Specifically, for a given group, an *'ideal item-feature prototype'* is constructed in latent space and an *'ideal item-rating prototype'* is constructed in rating space to capture the features and ratings of this unitary preference, respectively. Through this aggregating process, we can also smooth the preferences of group members by low-rank matrix approximation, which is needed especially when the members'

[*]Corresponding author.

2374-8486/16 $31.00 © 2016 IEEE
DOI 10.1109/ICDM.2016.64

preferences conflict heavily.

After obtaining the aggregated preference that is measured by ideal item, group-oriented services can be easily applied. For instance, for group recommendation, the *top-K* items that are most similar to the ideal item can be recommended to the target group. For evaluation, we construct extensive experiments on CAMRa2011 (with real group structure) [14] and Yelp (with simulated group structure) data sets for this task, and the results clearly demonstrate the effectiveness of our methods.

The remainder of this paper is organized as follows. In Section II, we introduce the related work. Then, the details of our approach are given in Section III. In Section IV, experiments on group recommendation are presented. Finally, we conclude our work in Section V.

II. Related Work

In this section, we review the related work from two aspects. The first category of research is those about preference aggregation, especially applied on group recommendation task; the second category of research is the application of game theory, especially in computer science field.

A. Preference Aggregation for Group Recommendation.

Preference aggregation has been studied in many fields, such as operational research [15], [16], human decision [17], [18], and some group-oriented services [1]–[7]. Among these services, group recommendation is a popular and representative one [1]–[4]. In this paper, we also conduct our methodology and experiments on the group recommendation task. Thus, in this part, we mainly review the studies of preference aggregation on group recommendation.

Group recommendations have been highlighted in many domains, e.g., tourism [3], [19], movies [14] and crowdfunding [20]. Preference aggregation for group recommendation can be classified into two categories: *preference aggregation* (PA) approaches [8] and *score aggregation* (SA) approaches [9], [10]. PA approaches first aggregate the profiles of group members into one profile, and then make recommendations based on the aggregated profile. SA approaches, by contrast, first recommend items for each group member respectively, and then aggregate the final recommendation for the group. In almost all the previous methods, average (AVG) [10] and least misery (LM) [9] are two most widely-used strategies. AVG strategy aims to maximize the overall satisfactions of a group by averaging the recommendation scores of all group members as the group's final score, and LM strategy tries to make each group member happy by taking the lowest score of members as the group's final score. However, these strategies can't always generate a solution that is fair for each member in a group, leading to the low satisfaction or diversity. Recently, [21] proposed to find Nash equilibrium solutions for group recommendation. However, their recommendation for a group was a union of the separate recommendations for each member without aggregation operation. Thus, this method had to recommend several different items for a group since each only satisfies one or several group members, which is different from the group recommendation scenario in the real world.

Besides the studies of preference aggregation, in some specific practices of group recommendation, some unique recommending techniques [4], [14], [20], [22]–[25] were also proposed. These works tried to design complicated models to explore some available features, e.g., social features and location features, to achieve better recommendations. However, the focus of these research is not exactly the same with our study. Indeed, their approaches often depend on the results of preference aggregation (e.g., LM and AVG), and our solution is also helpful to their studies and other group-oriented services. In [22], [26], [27], authors explored social factors, e.g., relationship in real life or social network of members, for better group recommendation. These studies must be conducted with the real social relationship. However, in many cases, we can not access users' real social information or even the group structure is temporal, e.g., group buying. These work will fail but our approach can still work in these scenarios since we only use the individual information.

B. Game Theory Applications in Computer Science.

The traditional game theory [28], [29] is a branch of modern mathematics, economics or operations research. Specifically, game theory is the study of mathematical models of conflict and cooperation between intelligent rational decision-makers [11], [28]. It helps decision-makers in a game make the most optimum strategies to maximize the benefits they can gain. In game theory, Nash equilibrium is an important concept which refers to the solution of a game that no player can gain more benefits by changing her strategy when this solution is adopted [11], [13]. Nash equilibrium is a stable solution because all players have no motives to change their strategies in this case.

In computer science, game theory has been applied in robot systems [30], computational advertising [31], [32], social lending [33], network security [34], small world networks [35], etc. However, to the best of our knowledge, this is the first attempt applying Nash equilibrium in preference aggregation.

III. Methodology

In this section, we will present our method in detail. Our study focuses on the general scenario of preference aggregation, and the problem formalization only depends on the original user-item rating matrix. In the following, we first give the problem statement and overview of our methodology.

Problem Statement. Let $U = (U_1, U_2, ..., U_{|U|})$ represent all the users and $I = (I_1, I_2, ..., I_{|I|})$ represent all the items. A user group G of U is given by the set of all users in this group $G = (U_1, U_2, ..., U_{|G|})$, where each $U_i (1 \leq i \leq |G|)$ is a member of group G. Given a target group, the focus of our study problem is how to aggregate the group preference from individual preferences/ratings, which can be applied to some group-oriented services, e.g., group recommendation, for providing the same service/item to satisfy all members in this target group.

Method Overview. Fig. 1 shows the framework of our approach. For preprocessing, we first conduct rating matrix

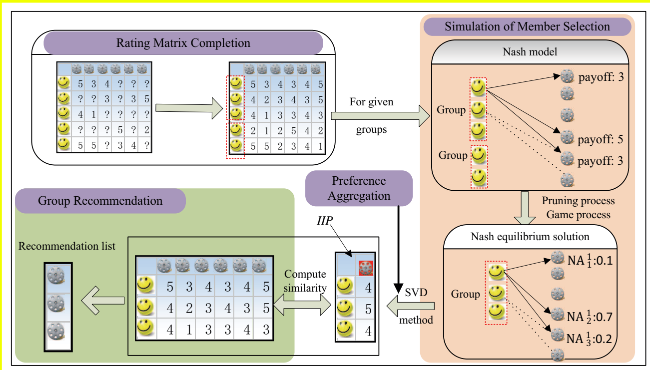


Fig. 1. The framework of our approach.

completion in order to estimate the individuals' preferences on the unseen items and avoid missing some potential strategies of members (Section III-A). Second, we simulate a static noncooperative game with complete information to model the individual selections of members in a group scene and

在團體中可能做這個決定的機率

we compute the Nash equilibrium solution, which stores the group-dependent optimal selection probabilities from each group member to different strategy items (Section III-B). Third, we design a matrix factorization-based method to find an ideal prototype that aggregates the members' preferences by integrating Nash equilibrium solution (Section III-C), and thus group recommendation can be applied.

A. Estimating Individual Preferences by Matrix Completion

As described above, the focus of our study problem, preference aggregation, is a process from individuals to groups. Thus, it is necessary to learn the individual preferences firstly. For estimating individuals' preferences to unseen/candidate items, and also, helping users select as many good strategies/items as possible in the following game process (which will be detailed in Section III-B), we conduct rating matrix completion in this subsection. Fortunately, there exist some studies aiming to solve the problem of user individual preference estimation [36]–[38]. As the focus of this paper is not to devise more sophisticated solutions on this problem, we adopt a simple yet effective method, i.e., using the nearest neighbor users for matrix completion [36]. Indeed, this step, i.e., matrix completion, has effects on the following solutions. However, this step does not affect the evaluation of preference aggregation since all the aggregating methods in our study are based on the same settings in this completion. Specifically, we achieve matrix completion as follows.

The element $\boldsymbol{R}(i,j)$ of rating matrix $\boldsymbol{R} = (\boldsymbol{R}(i,j))_{|U|\times|I|}$ is the rating that user U_i gives to the item I_j. We can estimate the unseen rating that U_i will give to item I_j by:

$$\boldsymbol{R}(i,j) = \frac{\sum_{k=1}^{M} usersim(i,k)\boldsymbol{R}(k,j)}{\sum_{k=1}^{M} usersim(i,k)}, \tag{1}$$

where M is the number of nearest neighbors, and $usersim(i,k)$ is the similarity between user U_i and U_k which is often calculated by *Cosine* measurement as follows,

$$usersim(i,k) = \frac{\sum_{I_l \in IU_i \wedge IU_k} \boldsymbol{R}(i,l)\boldsymbol{R}(k,l)}{\sqrt{\sum_{I_l \in IU_i} \boldsymbol{R}(i,l)^2 \sum_{I_l \in IU_k} \boldsymbol{R}(k,l)^2}}. \tag{2}$$

IU_i refers to the set of items that user U_i has rated. With this complete user-item matrix, we can understand the users' individual preferences to all items including the unseen ones. Based on this, in the next, we show how to use a game process to simulate the members' dependent selections in a given group scene.

B. Simulation of Member Selection in Given Group

By matrix completion, we have learned the members' individual preferences; and in this step, we try to obtain the group-dependent optimal selection of each member in a given group. This step is important and crucial by which we can learn both the members' preferences and their decision tradeoff in a group. We model the members' selections in group scene as a game process. Specifically, we first represent the strategies with two pruning algorithms for members in a given group; then we define a payoff function to compute the gain of a member to a strategy profile and compute the Nash equilibrium solution for this game, i.e., members' group-dependent optimal selection probabilities. Nash equilibrium solution implying the acceptable results of all members may bring high satisfaction to the entire group.

We choose the *static and noncooperative game with complete information* [12] in our simulation. The reason of choosing this type of game is the settings and scenes in this game are most similar to those in member decision process of group-oriented services such as group recommendation. Specifically, the reason is twofold,

- In our formalized problem, the system should initiatively and tentatively recommends same services or items for the entire target group, which means there is not extra interactions between users and system. Meanwhile, there are not any private exchanges between group members during this recommending process, too. These characteristics conform to the "static and noncooperative" game.
- In this kind of game, the numbers of players and strategies are finite and each member fully understands other members in her group, which is also the same in group recommendation scenario, i.e., the numbers of members in target groups and their favorite items are also finite. Further, we/systems also clearly know each member's individual preference to each item. Thus, these characteristics conform to the "complete information".

The direct correlations between the concepts in game theory and member selection in groups can be found in Table I. The group members are viewed as game players. An optional item that is likely to be chosen by a member is one of her strategies, and the set of optional items of this member is her optional item set, i.e., the strategy set. The optional item profile represents one combination of the strategies chosen by each member in the group, i.e., strategy profile. At last, the Nash equilibrium of the game is the members' group-dependent optimal selection probabilities to their items/strategies.

Strategy Representation and Pruning. Given a group $G = (U_1, U_2, ..., U_{|G|})$, item set $I = (I_1, I_2, ..., I_{|I|})$, the complete rating matrix is $\boldsymbol{R} = (\boldsymbol{R}(i,j))_{|G|\times|I|}$. For each member U_i, we

TABLE I
CORRELATED CONCEPTS.

Concepts in game theory	Concepts in member selection	Notations
Player	Group member	U_i
Strategy	Optional item	S_j^i
Strategy set	Optional item set	S^i
Strategy profile	Optional item profile	S_x
Set of strategy profile	Set of optional item profile	S
Payoff function	Gains from optional item profile	$p(i, S_x)$
Nash equilibrium solution	Members' optimal selection probabilities	NA

choose her favorite N items with the highest ratings (real rated or estimated in Section III-A) as her strategies. Thus, the strategy set of U_i can be denoted as:

$$S^i = \{S_1^i, S_2^i, ..., S_N^i\}, \tag{3}$$

where S_j^i is U_i's *j-th* strategy. For instance, if S_1^1 is 3, member U_1 will probably choose item I_3. Actually, strategies with conflict member preferences are more helpful to learn the game and tradeoff between members. In addition, some strategies which are not preferred by the consensus of all members in a group are useless and also cost a lot computation in the following. Thus, we propose to prune these useless strategies in advance by their dominance relationship, i.e., skyline and skyband [39], [40]. Specifically, for a given group, if all the members prefer strategy S_j^i to another one S_k^i, then strategy S_k^i is dominated by S_j^i (denoted as $S_j^i \prec S_k^i$) and strategy S_k^i can be pruned. Based on this dominance relationship, we define the strategy skyline as follows.

Definition *(Strategy Skyline) Given a strategy set S^i of group member U_i, a strategy $S_j^i \in S^i$ is a strategy skyline object if and only if any other strategies in the set can't dominate strategy S_j^i, denoted as $\forall S_k^i \in S^i, S_k^i \nprec S_j^i, k \neq j$.* ■

The definition of strategy skyline is strict. To guarantee the top-N' strategy candidates, we extend skyline to N'-skyband.

Definition *(N'-Strategy Skyband) Given a strategy set S^i of group member U_i, a strategy $S_j^i \in S^i$ is a N'-strategy skyband object if and only if strategy S_j^i is dominated by at most other N' strategies in this set. Usually, N' can be defined by N, e.g., $N' = \lceil N/|G| \rceil$.* ■

Clearly, the strategy skyline is the 1-strategy skyband. Parameter N' should be inversely proportional to group size $|G|$. Because a dominated strategy in a larger group is more likely to be worthless since more users think this strategy is dominated/worse. Based on the definition of N'-strategy skyband, we can process pruning in Algorithm 1.

After pruning, we further define the set of strategy profiles (denoted as S) of group G as the *Cartesian product* of N'-strategy skyband sets of all members:

$$\begin{aligned} S =& S^1 \times S^2 \times ... S^{|G|} \\ =& \{S_1 = (S_1^1, S_1^2, ..., S_1^{|G|}), S_2 = (S_2^1, S_1^2, ..., S_1^{|G|}), ..., \\ & S_x = (S_{x_1}^1, S_{x_2}^2, ..., S_{x_{|G|}}^{|G|}), ..., S_{N^{|G|}} = (S_{N_1}^1, S_{N_2}^2, ..., S_{N_{|G|}}^{|G|})\}, \end{aligned} \tag{4}$$

where $S_{x_i}^i$ is the x_i*-th* strategy of member U_i, and $S_x = (S_{x_1}^1, S_{x_2}^2, ..., S_{x_{|G|}}^{|G|})$ is a strategy profile of group G, N_i is the number of strategies of U_i after pruning. From this definition we can see that without strategy pruning (Algorithm 1), the number of strategy profiles of group G should be $N^{|G|}$, as each

Algorithm 1: N'-Strategy Skyband Pruning.

Input: Members' initial strategy sets $(S^1, S^2, ..., S^{|G|})$ in a given group, N';
Output: N'-Strategy Skyband sets;
for $i = 1; i \leq |G|; i++$ **do**
  for $j = 1; j \leq N; j++$ **do**
    $DomN = 0$;
    for $j' = 1; j' \leq N; j'++;\ j' \neq j$ **do**
      if $\forall i' \in \{1, ..., |G|\}$, $\boldsymbol{R}(i', S_{j'}^i) \geq \boldsymbol{R}(i', S_j^i)$ **then**
        $DomN++$;
    if $DomN > N'$ **then**
      S^i.remove(S_j^i);
return N'-Strategy Skyband sets $(S^1, S^2, ..., S^{|G|})$

group member has N strategies. After pruning, this number now becomes $\prod_{i \in \{1,...,|G|\}} N_i$. Since $N_i (1 \leq N_i \leq N)$ is much smaller than N, the time complexity of the following computations can be reduced sharply, and we will prove this experimentally. We should note that for better presentation in the matrix form, we usually use N and N_i without distinction.

Payoff Computation. In the game process, a member will get different gains from different strategy profiles. We define a payoff function ($p(i, S_x)$) to represent the gain of a member U_i to a specific strategy profile S_x:

$$p(i, S_x) = \frac{1}{|G|} \sum_{l=1}^{|G|} itemsim(S_{x_l}^l, S_{x_i}^i) \times \boldsymbol{R}(i, S_{x_i}^i), \tag{5}$$

where $itemsim(j, k)$ is the similarity between item I_j and item I_k, and $\boldsymbol{R}(i, S_{x_i}^i)$ denotes the rating that member U_i gives to item $S_{x_i}^i$. $itemsim(j, k)$ can be calculated by:

$$itemsim(j, k) = \frac{\boldsymbol{R}(1 : |G|, j) \cdot \boldsymbol{R}(1 : |G|, k)}{\|\boldsymbol{R}(1 : |G|, j)\|_2 \|\boldsymbol{R}(1 : |G|, k)\|_2}, \tag{6}$$

where $\boldsymbol{R}(1 : |G|, j)$ is $(\boldsymbol{R}(1, j), ..., \boldsymbol{R}(|G|, j))^T$. This payoff function means the expectation of benefit that U_i can get from strategy profile S_x, where she chooses item/strategy $S_{x_i}^i$. Her satisfaction expectation is viewed as the average product of the similarity $itemsim(S_{x_l}^l, S_{x_i}^i)$ and the rating that U_i gave to item $S_{x_i}^i$. For instance, if U_i chooses an item that is more similar to others', then she can get more benefit from this item because there will be a greater probability for her item to be adopted by the entire group. That is, $\boldsymbol{R}(i, S_{x_i}^i)$ term prompts members to choose items they like, and $itemsim(S_{x_l}^l, S_{x_i}^i)$ term prompts them to choose items that others probably like. The payoff function is the bridge to get the solution, i.e, Nash equilibrium, in a static and noncooperative game.

Nash Equilibrium Solution. In this game, the number of players ($|G|$) and the number of strategies (N) are finite, and we allow members to choose strategies with probabilities. For instance, U_1 may choose optimal item I_3 with probability 0.4 and choose another optimal item I_5 with probability 0.6, this is called as a mixed strategy. The *Nash's Existence Theorem* [11] tells us that if mixed strategies are accepted, a game with a finite number of players and strategies has at least one Nash equilibrium. Thus, our simulation of individual selection in a group can reach at least one Nash equilibrium, denoted as:

$$\begin{aligned} & NA = (NA^1, ..., NA^{|G|}), \\ & NA^i = (NA_1^i, ..., NA_N^i)^T, \quad \sum_{j=1}^{N} NA_j^i = 1. \end{aligned} \tag{7}$$

This above formalization means member U_i chooses or insists the optional item S^i_j with probability NA^i_j in her group.

Given the set of strategy profiles S and the payoff of each member to a strategy profile (e.g., $p(i, S_x)$), the Nash equilibrium solution NA for this group can be computed by solving a constrained optimization problem. For better description, we introduce two matrices $\boldsymbol{P}_{|G|\times N}$ and $\boldsymbol{Q}_{|G|\times 1}$. $\boldsymbol{P}_{i,j}$ represents the probability that the group member U_i will choose item S^i_j. $\boldsymbol{Q}_i$ represents the greatest expectation of satisfaction of member U_i. For better formalization, we turn $\boldsymbol{P}$ and $\boldsymbol{Q}$ into a vector $\boldsymbol{x}$ of size m=$|G| \times N + |G|$ defined as follows, $(\boldsymbol{x}((i-1)N+1), ..., \boldsymbol{x}((i-1)N+N))$ represents $(\boldsymbol{P}_{i,1}, ..., \boldsymbol{P}_{i,N})$ and $\boldsymbol{x}(|G|N + l), 1 \leq l \leq |G|$ represents $\boldsymbol{Q}$. Then the Nash equilibrium solution can be formalized as the following non-linear optimization problem. In this optimization problem, $\boldsymbol{x}$ with an initial $\boldsymbol{P}$ ($\boldsymbol{P}_{i,j} = 1/N$) and initial $\boldsymbol{Q}$ ($\boldsymbol{Q}_i$ is a large constant) is the input variable, and the $\boldsymbol{P}$ term of the final optimal $\boldsymbol{x}$ is the solved Nash equilibrium solution NA, i.e., members' group-dependent optimal selection probabilities to each strategy/item in a group:

$$
\begin{aligned}
&\min_{\boldsymbol{x}} f(\boldsymbol{x}) \\
&s.t.\ g(\boldsymbol{x}) \leq 0, \quad h(\boldsymbol{x}) = 0 \\
&\quad \boldsymbol{x}(i) \geq 0, \forall i = 1, ..., N|G|. \\
&\quad \boldsymbol{x}(i) \ \textit{are unrestricted}, \forall i = |G|N + 1, ..., |G|N + |G|. \\
&where: \ f(\boldsymbol{x}) = \sum_{i=1}^{|G|} (\boldsymbol{x}(|G|N + i) - \sum_{S_j \in S} p(i, S_j) \prod_{l=1}^{|G|} \boldsymbol{x}((l-1)N + j_l)), \\
&\quad g(\boldsymbol{x}) = \sum_{S_j \in S} p(i, S_j) \prod_{l=1}^{|G|} \boldsymbol{x}'((l-1)N + j_l) - \boldsymbol{x}(|G|N + i), \\
&\quad h(\boldsymbol{x}) = \sum_{j_l=1}^{N} \boldsymbol{x}((i-1)N + j_l) - 1, \forall i = 1, ..., |G|, \\
&\quad \boldsymbol{x}((l-1)N + j_l) = \boldsymbol{x}'((l-1)N + j_l), \forall l = 1, ..., |G|, l \neq i.
\end{aligned}
\tag{8}
$$

In the above formalization, $f(\boldsymbol{x})$ is the sum of the difference between the greatest expectation of satisfaction (i.e., $\boldsymbol{x}(|G|N+i)$) and the realistic payoff (i.e., $\sum_{S_j \in S} p(i, S_j) \prod_{l=1}^{|G|} \boldsymbol{x}((l-1)N+j_l)$) of each group member (i.e., U_i). The boundary condition $g(\boldsymbol{x}) \leq 0$ means that U_i will get fewer benefits if she changes her mixed optional item in this situation, and $\boldsymbol{x}'$ is different from $\boldsymbol{x}$ only by replacing U_i's mixed strategy with her single optimal strategy [41]. Another condition $h(\boldsymbol{x}) = 0$ is the normalization which means the sum of each member's selection probabilities to her strategies equals to 1. As a result, the solution of this optimization will be a Nash equilibrium (the stable situation that no member wants to change her mixed strategies) from which the entire group can reach the greatest satisfaction. We adopt the sequential quadratic programming based quasi Newton method [41] to solve this minimization problem, which can get one approximate Nash equilibrium solution.

By assuming the individual selections of members in a group as a game process, we get the members' group-dependent optimal selection probabilities to strategy items. For example, if $NA^i = (0.3, ..., 0.5)^T$, member U_i will choose items S^i_1,...,S^i_N with probabilities 0.3,...,0.5 respectively in this given group. However, Nash equilibrium solution just expresses the members' group-dependent individual selections in a given group while group-oriented services should provide the same item/service to all members in a group. Thus, Nash solution can't be applied directly. In the next, we will show the way to extract the aggregated unitary preference for each group by integrating this Nash solution.

C. Preference Aggregation

Here, we deal with preference aggregation, i.e., figure out the specific *ideal item* that can perfectly match the unitary group preferences (or selections in the above simulation of all group members) of each group. Considering the best choice of a group should be the items with factors that Nash equilibrium solution owns, we first do SVD on the group member-strategy matrix and construct an *'ideal item-feature prototype'* (*IFP*) which concludes the features of ideal item in latent factor space by integrating the Nash equilibrium solution. In this way, we can also achieve dimensionality reduction and compression by discarding some unimportant factors to smooth the preferences of group members. Second, based on *IFP*, we define an *'ideal item-rating prototype'* (*IIP*) to represent the aggregated preference (i.e., the ideal item) of target group in rating space. After getting the ideal item, we can finish group recommendation by recommending the items which are the most similar to this ideal one.

Ideal Item-Feature Prototype. Before introducing the *ideal item-feature prototype*, we first represent the optional items/strategies of each group into latent space. Specifically, for a group $G = (U_1, U_2, ..., U_{|G|})$, members' strategy set $IS = (I_1, ..., I_{|IS|}) = \bigcup_{i=1}^{|G|} \bigcup_{j=1}^{|N|} I_{S^i_j}$. Please note that, IS is the group's strategy set in which each item belongs to at least one of the group member's strategy sets. We compute singular value decomposition of matrix $(\boldsymbol{R}(i, j))_{|G|\times|IS|}$ as:

$$(\boldsymbol{R}(i, j))_{|G|\times|IS|} = \boldsymbol{A}_{|G|\times|G|} \boldsymbol{D}_{|G|\times|IS|} \boldsymbol{V}^T_{|IS|\times|IS|}, \tag{9}$$

where $\boldsymbol{R}(i, j)$ is the rating of member U_i to the *j-th* item, $\boldsymbol{A}$ is the member-feature matrix, $\boldsymbol{D}$ is the weight matrix which is diagonal, and $\boldsymbol{V}$ is the strategy-feature matrix. One step further, we can achieve dimensionality reduction by low-rank matrix approximation:

$$
\begin{aligned}
\tilde{\boldsymbol{R}} &= \boldsymbol{A}_{|G|\times w} \boldsymbol{D}_{w\times w} \boldsymbol{V}^T_{|IS|\times w} \\
&= \tilde{\boldsymbol{A}} \tilde{\boldsymbol{D}} \tilde{\boldsymbol{V}}^T, \\
where: \quad w &= \min \left\{ w \middle| \frac{\sum_{k=1}^{w} \boldsymbol{D}(k,k)}{\sum_{k=1}^{|G|} \boldsymbol{D}(k,k)} > \alpha \right\},
\end{aligned}
\tag{10}
$$

w is the number of remaining features. Parameter α controls the degree of smoothness or denoising, e.g., when α is smaller (w is also smaller), the smoothness become more heavily. This process is necessary especially when the members' preferences conflict very much.

Then, for aggregating the preferences of group members in the decomposed latent space by integrating Nash equilibrium solution NA, we define *ideal item-feature prototype* for each group (denoted as $IFP_{1\times w}$) as:

$$IFP = \frac{1}{|G|} \sum_{i=1}^{|G|} \sum_{j=1}^{|IS|} NA^i_j \tilde{\boldsymbol{V}}(IS^i_j, 1:w), \tag{11}$$

where IS^i_j is the position of U_i's strategy S^i_j in strategy set IS. Thus, IFP could be viewed as the ideal item in the latent factor space after preference smoothness and aggregation.

Ideal Item-Rating Prototype and Group Recommendation. Given *IFP*, if we want to provide group-oriented services, e.g., group recommendation, we have to figure out the items which are most similar to the *IFP*. A straightforward solution is to project all the candidate items (not rated by all members in target group) into latent space for each group like those in strategy set IS, and then compute the similarity of items in this space. However, the size of candidate item set is much larger than that of strategy set, and it is also difficult and time consuming to project each item's ratings in latent space by SVD once again. Alternatively, in this paper we consider a reverse process, i.e., project the *IFP* back to the prime matrix $\boldsymbol{R}$ by matrix product and find the similar items in rating space, which is much easier and quicker.

Thus, we define *ideal item-rating prototype* $IIP_{|G|\times 1}$ as:

$$IIP = \tilde{\boldsymbol{A}} \cdot \tilde{\boldsymbol{D}} \cdot IFP^T. \tag{12}$$

IIP is indeed the ideal item or prototype of aggregated unitary preference in rating space. Actually, when a group is making decisions, each term in *IIP* implies in what degree of this member' preference can be expressed or considered in this group. Thus, for group-oriented service, we should recommend the candidate items whose ratings given by each group member are very close to the terms of given group's *IIP*. To measure the difference/similarity between each candidate item I_j and *IIP*, we compute the *ideal-item distance* $IID(I_j, IIP)$ as:

$$IID(I_j, IIP) = \|\boldsymbol{R}(1 : |G|, I_j) - IIP\|_2. \tag{13}$$

Here, we do not need to consider the situation that there exist items that all of its ratings are higher than *IIP*. Actually, this situation does not exist because if it exists, it will contribute to an *IIP* with higher ratings as well. Then, we can choose the *top-K* items with lowest *IID* values as our final recommendations for group G, as group members are more likely to reach an agreement on the items which are similar to the aggregated preferences.

Discussion. In summary, our solution for preference aggregation mainly contains two steps: In the first step, we model the individual selections of members in a group as a static noncooperative game and obtain the members' group-dependent optimal selection probabilities by Nash equilibrium. In the second step, we leverage SVD to achieve preference aggregation for each group by integrating Nash equilibrium, and there are two advantages: First, SVD enables us to conclude the features but not just the ratings of the group preference, as a pair of similar items can have different ratings but similar features. Second, SVD can smooth the preferences of group members by low-rank matrix approximation, which filters the extreme preferences of some members that are too different from others'. The smoothness could also be controlled by changing parameter α in Equation (10).

IV. Experiments

In this section, we evaluate our approach by constructing experiments on group recommendation task. Specifically, we first introduce the experimental data sets and setup. Then, we report the experimental results from the following aspects: we first present a simple case study for readers to understand our approach easily (Section IV-C1); then we present extensive results on efficiency (Section IV-C2); effectiveness (Section IV-C3) and robustness (Section IV-C4); at last, we give some discussions about the effects of parameters in our method (Section IV-C5).

A. Experimental Data

In the experiments, we use two real-world data sets, i.e., *CAMRa2011* and *Yelp*[1]. CAMRa2011 has 290 households with 602 users who gave 145,069 ratings over 7,740 movies. Yelp hosts an online database of 2,906 users generated 22,333 ratings[2] on 1,555 businesses, e.g., restaurants. In CAMRa2011, the households/families are treated as groups [14]. Since there is no group information in Yelp data set, we take the users who visited the same restaurants on same days more than two times as a group [25]. Thus, a user may be grouped into more than one group in Yelp. Table II shows the basic statistical information of the experimental data.

TABLE II
Group statistics in experimental data.

GroupNumber\GroupSize	2	3	4	5	6	7
CAMRa2011	272	14	4	-	-	-
Yelp	1,494	1,402	1,272	1,124	996	877

B. Experimental Setup

In the experiments, without loss of generality, the neighbor number (M) in matrix completion is set as 10. The number of each member's optional items (N) is set as 6 empirically and the pruning parameter N' is set as $\lceil N/|G| \rceil$. The stop criteria of Equation (8) is that the changing of $\boldsymbol{x}$ (or $f(\boldsymbol{x})$) is less than 10^{-4}(or 10^{-7}), which we think is of high-quality enough. The smooth parameter in SVD (α) is set as 0.97 without special illustration. For each data set, the data is divided into 5 parts, and we adopt the 5-fold cross-validation. Each time 4 parts are selected as training set, and the rest one is used for test.

1) Experimental Methods: We denote our approaches for preference aggregation and group recommendation as **Nash-NoP**, **Nash-Skyband** and **Nash-Skyline** which represent methods without pruning, pruning by skyband and pruning by skyline, respectively. We also select three state-of-the-art preference aggregation methods used on group recommendation as baselines.

Least misery strategy (LM) [9], [14], [24]. LM first estimates the rating of each item, and then choose the lowest value from the ratings given by all group members as the target group's predicted rating on each item. At last, choose items with the highest predicted ratings as recommendations.

Averaging strategy (AVG) [10], [14], [24]. AVG sets the

[1]http://www.yelp.com.au/dataset_challenge

[2]The rating scales in CAMRa2011 and Yelp are [1,100] and [1,5] respectively. We preprocess ratings into the same scale, i.e., [1,5].

average rating given by the group members to each item as the predicted rating of target group. At last, choose the items with the highest predicted ratings as recommendations.

Relevance and disagreement (RD) [2], [25]. Different from LM and AVG, RD calculates recommendation score for a candidate item based on the relevance and disagreement of a group. The relevance is calculated based on AVG strategy, and the disagreement can be the variance of members' ratings.

2) Metrics: For each target group G, we recommend an item list L from the candidate items, and we adopt four widely-used metrics in group recommendation for evaluation. In the following experiments, all the results on these metrics are calculated by the averages of all the groups.

Hit Rate (HR) [42]. HR measures the proportion of recommendation lists which contain items that group members adopt in all recommendation lists. $HR = \sum_{L \in all\ lists} \boldsymbol{I}(L,G)/ListNum$, where $\boldsymbol{I}(L,G)$ is an indicator function whose value is 1 when L contains at least one item adopted by G and 0 otherwise. *Adopted items* refer to the items that are real rated 3 or higher by all members.

Satisfaction Gain (SG) [43]. SG measures the satisfaction of a group to a recommendation list. $SG = \frac{1}{|G|}\sum_{j=1}^{|G|}\sum_{k=1}^{|L|}\boldsymbol{R}(j,k)$, where U_j is a member in G, I_k is an *adopted item* in L, $\boldsymbol{R}(j,k)$ is the rating that U_j gives to I_k.

Hamming Distance (HD) [44]. HD measures the diversity of recommendation lists between different groups. HD between group G_i and group G_j, $HD(i,j) = 1 - Q_{i,j}/|L|$, where $Q_{i,j}$ is the number of items that are recommended to both G_i and G_j. High HD means high diversity, and the recommendations to all groups make full use of all items and few items will be left without being recommended.

Harmonic (H) [45]. Harmonic metric measures the equity of the recommendation list to a group in the condition of high satisfaction, i.e., $H = |G|/(\sum_{j=1}^{|G|}\frac{1}{\sum_{k=1}^{|L|}\boldsymbol{R}(j,k)})$. If harmonic is high, recommendation is fair to all members.

C. Experimental Results

1) Case Study: We randomly select a small group from CAMRa2011 data which contains 3 family members and 5 items (Table III). We also suppose $\{I_1, I_2, I_3, I_4, I_5\}$ are all the candidate items that can be recommended to all these three members. That means the elements of the table below are the evaluated/estimated ratings that each user will give to each item. Please note that, in our approach, we consider both the real ratings and estimated ratings in strategy selection, and in this case study, we simplify this process. We only show the Nash-NoP process in this case due to the limited space.

TABLE III
SELECTED GROUP IN CASE STUDY.

	I_1	I_2	I_3	I_4	I_5
U_1	4.86	3.62	3.80	2.76	2.76
U_2	5.00	4.30	3.63	3.78	5.00
U_3	4.02	3.53	3.60	3.17	5.00

For Nash-NoP, we pick the 3 items with the highest ratings from I_1 to I_5 as a member's strategy set, i.e., S^1=(I_1, I_3, I_2), S^2=(I_1, I_5, I_2), S^3=(I_5, I_1, I_3). Then we calculate the value of payoff function of each possible strategy profile S_x. For example, S_x=(I_1, I_1, I_5) means that U_1, U_2, U_3 chooses I_1, I_1, I_5 respectively. By Equation (5), the benefit of U_1 from this optional item profile S_x is: $p(1, S_x)$=$(4.86+cos((4.86, 5.00, 4.02), (4.86, 5.00, 4.02)) \times 4.86 + cos((2.76, 5.00, 5.00), (4.86, 5.00, 4.02)) \times 4.86)/3 = 4.79$. After computing the payoff function values of all strategy profiles and solving the non-linear optimization problem (8), we get the Nash equilibrium solution NA in Equation (7) as shown in the following table, which means that U_1 chooses I_1 with probability 1, U_2 chooses I_1 with probability 0.64 and chooses I_5 with probability 0.36, U_3 chooses I_5 with probability 1.00.

	U_1	U_2	U_3
S_1^i	1.00	0.64	1.00
S_2^i	0.00	0.36	0.00
S_3^i	0.00	0.00	0.00

Then we use SVD decomposition to conclude the features of the Nash equilibrium solution[3]. By Equation (9), the SVD decomposition of $\boldsymbol{R}(1:3,[1,2,3,5])$=$\boldsymbol{ADV}^T$ is as follows:

$$\boldsymbol{A}= \begin{matrix} -0.53 & -0.82 & 0.22 \\ -0.63 & 0.21 & -0.75 \\ -0.57 & 0.53 & 0.63 \end{matrix}$$

$$\boldsymbol{D}= \begin{matrix} 14.28 & 0 & 0 & 0 \\ 0 & 1.72 & 0 & 0 \\ 0 & 0 & 0.46 & 0 \end{matrix}$$

$$\boldsymbol{V}= \begin{matrix} -0.56 & -0.47 & -0.32 & 0.60 \\ -0.46 & -0.12 & -0.44 & -0.76 \\ -0.44 & -0.26 & 0.84 & -0.18 \\ -0.52 & 0.83 & 0.02 & 0.18 \end{matrix}$$

where A(i,k) measures the preference of member U_i to feature F_k, $D_{k,k}$ measures the importance of F_k, $V_{j,k}$ measures the preference of item I_j to feature F_k.

After the decomposition, in the next, we show how to aggregate the unitary preference for this group and obtain the final recommended items. By Equation (10), We set $\alpha = 0.97$ to denoise $\boldsymbol{D}$ to $\boldsymbol{D(1:2,1:2)}$ as follows:

$$\tilde{\boldsymbol{A}}= \begin{matrix} -0.53 & -0.82 \\ -0.63 & 0.21 \\ -0.57 & 0.53 \end{matrix} \quad \tilde{\boldsymbol{D}}= \begin{matrix} 14.28 & 0 \\ 0 & 1.72 \end{matrix} \quad \tilde{\boldsymbol{V}}= \begin{matrix} -0.56 & -0.47 \\ -0.46 & -0.12 \\ -0.44 & -0.26 \\ -0.52 & 0.83 \end{matrix}$$

By Equation (11), the Nash equilibrium solution is used to calculate the ideal item-feature prototype IFP.

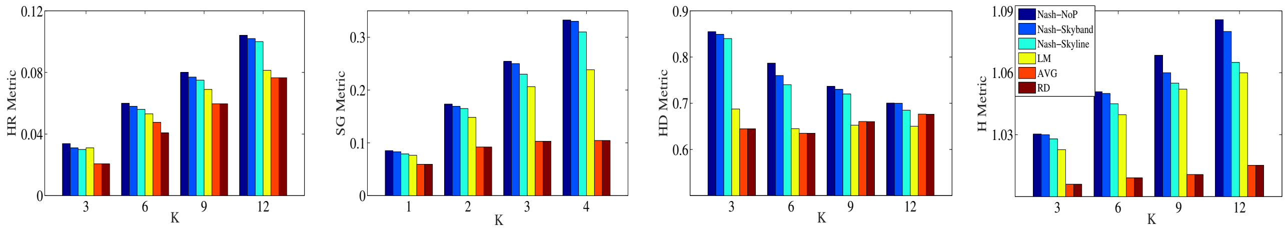
$$\begin{aligned} IFP =&((1.00, 0.00, 0.00).\tilde{\boldsymbol{V}}([1,3,2], 1:2) \\ &+ (0.64, 0.36, 0.00).\tilde{\boldsymbol{V}}([1,4,2], 1:2) \\ &+ (1.00, 0.00, 0.00).\tilde{\boldsymbol{V}}([4,1,3], 1:2))/3 \\ =&((-0.56, -0.47) + (-0.56, -0.47) \times 0.64 \\ &+ (-0.52, 0.83) \times 0.36 + (-0.52, 0.83))/3 \\ =&(-0.54, 0.12). \end{aligned}$$

By Equation (12), the ideal item-rating prototype IIP is computed as:

$$IIP = \tilde{A} \times \tilde{D} \times IFP^T = (3.92, 4.90, 4.50)^T.$$

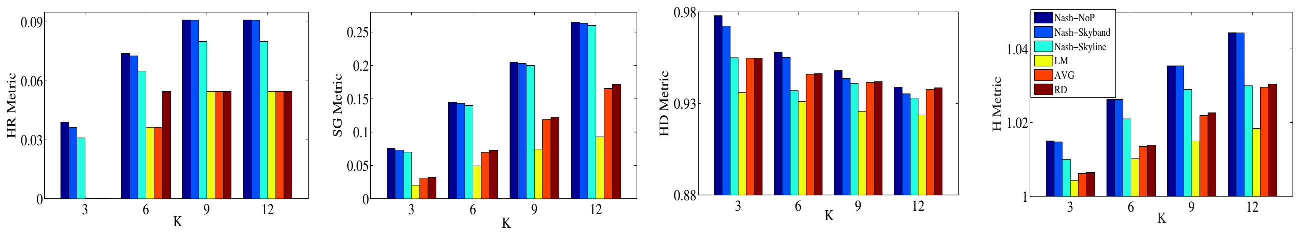
IIP is the ideal item of unitary preference in rating space for this group. The items with estimated individual ratings from three members more similar *IIP*'s corresponding terms are better. By Equation (13), the ideal-item distance *IID* is:

[3]Note: $\{I_1, I_2, I_3, I_5\}$ is the members' strategy set IS which will be used in SVD decomposition, and the recommended candidate set is $\{I_1, I_2, I_3, I_4, I_5\}$.



(a) Hit Rate (b) Satisfaction (c) Diversity (d) Harmonic

Fig. 2. Results w.r.t. item number (K) on CAMRa2011.



(a) Hit Rate (b) Satisfaction (c) Diversity (d) Harmonic

Fig. 3. Results w.r.t. item number (K) on Yelp.

$$
\begin{aligned}
IID(I_1, IIP) =& \|\boldsymbol{R}(1:|G|, I_1) - IIP\|_2 \\
=& \|(4.86, 5.00, 4.02) - (3.92, 4.90, 4.50)\|_2 = 1.06 \\
IID(I_2, IIP) =& 1.18 \quad IID(I_3, IIP) = 1.56 \\
IID(I_4, IIP) =& 2.09 \quad IID(I_5, IIP) = 1.27
\end{aligned}
$$

$IID(I_1, IIP)$ and $IID(I_2, IIP)$ are the lowest, which means I_1 and I_2 are similar to the ideal item-rating prototype IIP. In other words, I_1, I_2 are closer to the best choice of members. If we pick I_1 and I_2 as recommendations, it's more likely that each member has a relatively high satisfaction to them and they have minimal motives to replace them with other items. Thus, I_1, I_2 are our final recommendations.

Then, we show the recommendations of our baseline methods. Intuitively, the average ratings for $I_1 \sim I_5$ are 4.63, 3.82, 3.68, 3.24, 4.25. For AVG method, it will pick the items with the highest average ratings, which are I_1 and I_5. The lowest ratings for $I_1 \sim I_5$ are 4.02, 3.53, 3.60, 2.76, 2.76. For LM method, it will pick I_1 and I_3. The RD method gives $I_1 \sim I_5$ 4.60, 3.97, 3.92, 3.49, 4.14. It will pick I_1 and I_5.

We can find that both AVG and RD recommend I_5. Though U_2 and U_3 give I_5 5 points, U_1 only gives I_5 2.76. I_5 is an unfair recommendation for U_1. Thus, I_5 is not an adopted item of this group. LM recommends I_3, which the 3 ratings given by the members are close. However, I_2's ratings are close too, and the ratings are relatively higher than I_3's. However, LM does not recommend I_2. Compared with these methods, Nash-NoP recommends I_1 and I_2 whose ratings given by all the members are relatively high.

2) Efficiency Results: In this part, we report the efficiency results by comparing the strategy spaces of different methods and their running times. The number of all generated strategies and the average running times of each group on two data sets are shown in Table IV. These time records are the total running times of the whole recommendation processes of different methods except the cost of first step (i.e., the time of estimating individual preferences by matrix completion is not recorded). From the aspect of strategy space, we can see that, Nash-Skyband prunes about 40% strategies on CAMRa2011 and about 35% strategies on Yelp, while Nash-Skyline prunes about 48% strategies on CAMRa2011 and about 45% strategies on Yelp. From the aspect of running time, LM, AVG and RD are very efficient since they directly compute the item scores for each group. LM, AVG and RD need more time on CAMRa2011 than on Yelp. That is because the item is denser on CAMRa2011, and these methods have to compute scores with all the candidate items for each group. Nash-Series are not sensitive to item density since they always use a fixed number (N) of strategies. However, Nash-Series take much more time on Yelp since there are many large-size groups in Yelp. Further, Nash-Skyband needs about 50% and 27% times of Nash-Nop on CAMRa2011 and Yelp, while Nash-Skyline only needs 12% and 1.5% times of Nash-Nop on these two data sets. Thus, we can conclude that the two pruning approaches proposed in this study are very effective compared with Nash-NoP .

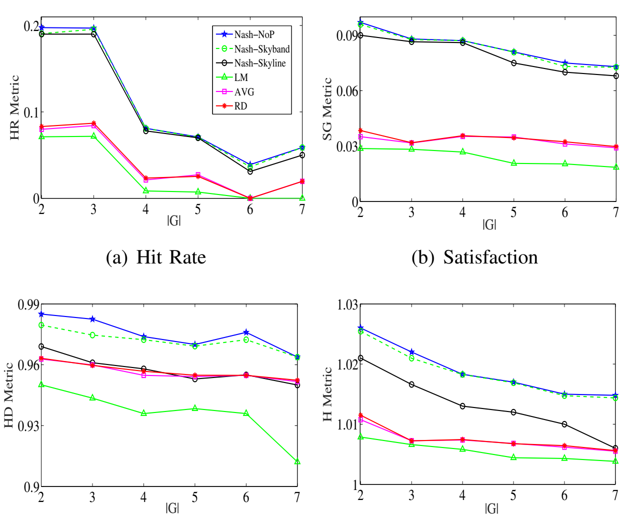
TABLE IV
EFFICIENCY RESULTS.

Methods\Results	Number of all strategies		Average running time (seconds)	
	CAMRa2011	Yelp	CAMRa2011	Yelp
Nash-Nop	3,612	180,102	2.851	121.872
Nash-Skyband	2,167	117,066	1.380	33.439
Nash-Skyline	1,878	99,056	0.344	1.793
LM	-	-	0.010	0.004
AVG	-	-	0.039	0.017
RD	-	-	0.126	0.055

3) Effectiveness Results: In this part, we mainly evaluate the recommendation performances of different methods. The average cross-validation results on four metrics with respect to the length of recommended list are shown in Fig. 2 and Fig. 3. From Fig. 2(a) and Fig. 3(a) we can see that, Nash-NoP performs best with all the list sizes on HR metric, and the HR values of baseline methods are near to zero when the item numbers are small in Fig. 3(a). Fig. 2(b) and Fig. 3(b) report the SG results on two data sets. The proposed Nash-Series

(Nash-NoP, Nash-Skyband and Nash-Skyline) provide higher SG values than baselines. Although AVG directly optimizes the satisfaction, it does not come out as the best. Because we only consider the *adopted items* when computing the SG values, AVG may lose some gains on the items whose average ratings are high but not be accepted by all the members in a group. Fig. 2(c) and Fig. 3(c) show the results on HD metric, and the items output by Nash-NoP and Nash-Skyband are more diverse than others. Fig. 2(d) and Fig. 3(d) report the harmonic results, which are similar with the satisfaction results, i.e., Nash-Series perform better compared with baselines. In terms of different data sets, the performances of almost all methods on CAMRa2011 are better than those on Yelp. The reason is that the groups in CAMRa2011 are much more homogeneous (families) and smaller than the groups in Yelp, and the agreement in homogeneous and small-size groups is more easy to achieve. That may be also the reason that LM performs much better on CAMRa2011 than on Yelp. In summary, on almost all the metrics, the performances of Nash-Skyband are very close to Nash-NoP, and Nash-Skyline performs a little worse compared to Nash-NoP and Nash-Skyband. Thus, Nash-Skyband pruning is much safer.

4) Robustness Results: We also discuss the robustness of these recommendation methods. We obtain the results on groups with different sizes only from Yelp data (since the group sizes in CAMRa2011 are almost only 2 or 3). In this experiment, we fix the length of the recommendation list as 9, and the results are shown in Fig. 4. In comparison, Nash-Series perform best on all metrics with any group sizes and Nash-Skyband performs very closely to Nash-NoP. There are different performances of LM, AVG and RD in different cases. Besides, all methods tend to perform worse when group size becomes larger, because the preference aggregation is more difficult in larger groups. These results demonstrate the robustness of Nash-Series.



(a) Hit Rate (b) Satisfaction

(c) Diversity (d) Harmonic

Fig. 4. Results w.r.t. group size on Yelp.

5) Parameter Effects: In this part, we make our efforts to study the effects of parameter α in Nash-Skyband method. In Fig. 5, we report the hit rate results with different α. Specifically, in Fig.5(a) (i.e., CAMRa2011), the group size is 2; while in Fig. 5(b) and Fig. 5(c) (i.e., Yelp), group sizes are 2 and 6 respectively, and the length of item list is 9. We can see that the hit rates get the best results with different certain α values. From the comparisons of Fig. 5(a) and Fig. 5(b), the optimal α on CAMRa2011 (around 0.92) is smaller than that on Yelp (around 1). The reason may be that CAMRa2011 data is much denser and the families are tolerant of each other, so we can smooth their preferences heavily by small α. From the comparisons of Fig. 5(b) and Fig. 5(c), the optimal α for smaller group (around 1) is larger than that for larger group (around 0.96). That is because smoothing one individual' preference in a larger group is much safer and this member may still accept the recommending results with considering the desires of the majority. Thus, larger and homogeneous-relation groups could take a more heavily smoothing, so that a smaller α is needed.

V. Conclusions and Future Work

In this paper, we presented a focused study on preference aggregation for group-oriented services. Specifically, we proposed a method that combines the Nash equilibrium and matrix factorization to aggregate the preferences of each group member, where we viewed the group members as game players and the member selection in group scene as a game process. In this way, the Nash equilibrium solution stores the group-dependent optimal selection probabilities from group members to items. To solve this Nash solution, two pruning approaches (Nash-Skyline, Nash-Skyband) were proposed for gaining efficiency. Then, we designed an aggregation method to aggregate the preferences of group members, and the items that match the aggregated preference were recommended. Finally, extensive experiments were conducted on two real-world data sets, and the results demonstrate the effectiveness and robustness of our proposed Nash-Series approaches.

In the future, we plan to include more constraints for preference aggregation, e.g., consider the different strategy numbers for different group members. Meanwhile, we will test our solutions in more complicated scenarios, e.g., more experimental settings on even larger data sets.

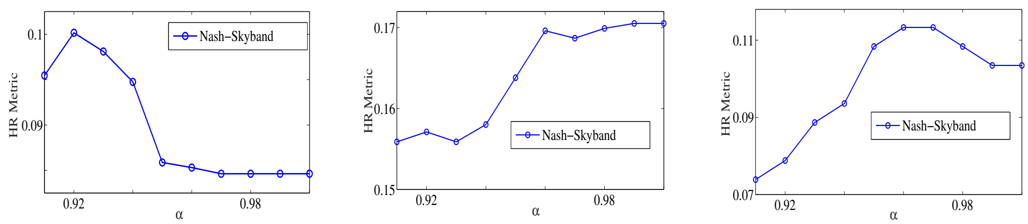
VI. Acknowledgements

This research was partially supported by grants from the National Science Foundation for Distinguished Young Scholars of China (Grant No. 61325010), the National Natural Science Foundation of China (Grant No. 61403358, 61572032, 71571093) and the National High Technology Research and Development Program of China (Grant No. 2014AA015203). Qi Liu gratefully acknowledges the support of the MOE-Microsoft Key Laboratory of USTC and the support of the CCF-Tencent Open Research Fund.

References

[1] A. Jameson and B. Smyth, "Recommendation to groups," in *The adaptive web*. Springer, 2007, pp. 596–627.

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



(a) α effects on CAMRa2011, $|G|$ = 2 (b) α effects on Yelp, $|G|$ = 2 (c) α effects on Yelp, $|G|$ = 6

Fig. 5. Effects of parameter α .

[3] K. McCarthy, M. Salamó, L. Coyle, L. McGinty, B. Smyth, and P. Nixon, "Cats: A synchronous approach to collaborative group recommendation." in *FLAIRS Conference*, vol. 2006, 2006, pp. 86–91.

[4] J. Gorla, N. Lathia, S. Robertson, and J. Wang, "Probabilistic group recommendation via information matching," in *WWW*. ACM, 2013, pp. 495–504.

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

[6] R. J. Kauffman and B. Wang, "Bid together, buy together: On the efficacy of group-buying business models in internet-based selling," pp. 99–137, 2002.

[7] W. M. Lim and D. H. Ting, "Consumer acceptance and continuance of online group buying," *Journal of Computer Information Systems*, vol. 54, no. 3, pp. 87–96, 2014.

[8] Z. Yu, X. Zhou, Y. Hao, and 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, 2006.

[9] L. Baltrunas, T. Makcinskas, and F. Ricci, "Group recommendations with rank aggregation and collaborative filtering," in *RecSys*. ACM, 2010, pp. 119–126.

[10] S. Berkovsky and J. Freyne, "Group-based recipe recommendations: analysis of data aggregation strategies," in *RecSys*. ACM, 2010, pp. 111–118.

[11] J. Nash, "Non-cooperative games," *Annals of mathematics*, pp. 286–295, 1951.

[12] K. Ritzberger, *Foundations of non-cooperative game theory*. Oxford University Press Oxford, 2002, vol. 9.

[13] R. Aumann and A. Brandenburger, "Epistemic conditions for nash equilibrium," *Econometrica: Journal of the Econometric Society*, pp. 1161–1180, 1995.

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

[15] E. Forman and K. Peniwati, "Aggregating individual judgments and priorities with the analytic hierarchy process," *European journal of operational research*, vol. 108, no. 1, pp. 165–169, 1998.

[16] S. H. Kim and B. S. Ahn, "Interactive group decision making procedure under incomplete information," *European Journal of Operational Research*, vol. 116, no. 3, pp. 498–507, 1999.

[17] D. H. Gruenfeld, E. A. Mannix, K. Y. Williams, and M. A. Neale, "Group composition and decision making: How member familiarity and information distribution affect process and performance," *Organizational behavior and human decision processes*, vol. 67, no. 1, pp. 1–15, 1996.

[18] F. Herrera, E. Herrera-Viedma, and F. Chiclana, "Multiperson decision-making based on multiplicative preference relations," *European journal of operational research*, vol. 129, no. 2, pp. 372–385, 2001.

[19] Q. Liu, E. Chen, H. Xiong, Y. Ge, Z. Li, and X. Wu, "A cocktail approach for travel package recommendation," *IEEE TKDE*, vol. 26, no. 2, pp. 278–293, 2014.

[20] V. Rakesh, W.-C. Lee, and C. K. Reddy, "Probabilistic group recommendation model for crowdfunding domains," in *WSMD*. ACM, 2016.

[21] L. A. M. C. Carvalho and H. T. Macedo, "Users' satisfaction in recommendation systems for groups: an approach based on noncooperative games," in *WWW companion*. ACM, 2013, pp. 951–958.

[22] M. Gartrell, X. Xing, Q. Lv, A. Beach, R. Han, S. Mishra, and K. Seada, "Enhancing group recommendation by incorporating social relationship interactions," in *international conference on Supporting group work*. ACM, 2010, pp. 97–106.

[23] X. Liu, Y. Tian, M. Ye, and W.-C. Lee, "Exploring personal impact for group recommendation," in *CIKM*. ACM, 2012, pp. 674–683.

[24] M. Ye, X. Liu, and W.-C. Lee, "Exploring social influence for recommendation: a generative model approach," in *SIGIR*. ACM, 2012, pp. 671–680.

[25] Q. Yuan, G. Cong, and C.-Y. Lin, "Com: a generative model for group recommendation," in *SIGKDD*. ACM, 2014, pp. 163–172.

[26] L. Quijano-Sanchez, J. A. Recio-Garcia, B. Diaz-Agudo, and G. Jimenez-Diaz, "Social factors in group recommender systems," *ACM TIST*, vol. 4, no. 1, p. 8, 2013.

[27] Q. Liu, Z. Dong, C. Liu, X. Xie, E. Chen, and H. Xiong, "Social marketing meets targeted customers: A typical user selection and coverage perspective," in *ICDM*. IEEE, 2014, pp. 350–359.

[28] R. B. Myerson, *Game theory*. Harvard university press, 2013.

[29] T. Basar, G. J. Olsder, G. Clsder, T. Basar, T. Baser, and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1995, vol. 200.

[30] E. Hernández, A. Barrientos, and J. del Cerro, "Selective smooth fictitious play: An approach based on game theory for patrolling infrastructures with a multi-robot system," *Expert Systems with Applications*, vol. 41, no. 6, pp. 2897–2913, 2014.

[31] Y. Yu, G. Q. Huang, and L. Liang, "Stackelberg game-theoretic model for optimizing advertising, pricing and inventory policies in vendor managed inventory (vmi) production supply chains," *Computers & Industrial Engineering*, vol. 57, no. 1, pp. 368–382, 2009.

[32] D. He, W. Chen, L. Wang, and T.-Y. Liu, "A game-heoretic machine learning approach for revenue maximization in sponsored search," in *IJCAI*. AAAI Press, 2013, pp. 206–212.

[33] N. Chen, A. Ghosh, and N. S. Lambert, "Auctions for social lending: A theoretical analysis," *Games and Economic Behavior*, vol. 86, pp. 367–391, 2014.

[34] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, and Q. Wu, "A survey of game theory as applied to network security," in *HICSS*. IEEE, 2010, pp. 1–10.

[35] Z. Yang and W. Chen, "A game theoretic model for the formation of navigable small-world networks," in *WWW*. ACM, 2015, pp. 1329–1339.

[36] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, "Eigentaste: A constant time collaborative filtering algorithm," *Information Retrieval*, vol. 4, no. 2, pp. 133–151, 2001.

[37] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM review*, vol. 52, no. 3, pp. 471–501, 2010.

[38] B. Recht, "A simpler approach to matrix completion," *The Journal of Machine Learning Research*, vol. 12, pp. 3413–3430, 2011.

[39] D. Papadias, Y. Tao, G. Fu, and B. Seeger, "Progressive skyline computation in database systems," *ACM TODS*, vol. 30, no. 1, pp. 41–82, 2005.

[40] Y. Tao, X. Xiao, and J. Pei, "Efficient skyline and top-k retrieval in subspaces," *IEEE TKDE*, vol. 19, no. 8, pp. 1072–1088, 2007.

[41] B. Chatterjee, "An optimization formulation to compute nash equilibrium in finite games," in *ICM2CS*. IEEE, 2009, pp. 1–5.

[42] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 143–177, 2004.

[43] C. Shahabi and Y.-S. Chen, "An adaptive recommendation system without explicit acquisition of user relevance feedback," *Distributed and Parallel Databases*, vol. 14, no. 2, pp. 173–192, 2003.

[44] T. Zhou, L.-L. Jiang, R.-Q. Su, and Y.-C. Zhang, "Effect of initial configuration on network-based recommendation," *EPL*, vol. 81, no. 5, p. 58004, 2008.

[45] Y. lun Chou, "Statistical analysis," *Holt International*, 1975.