Contents lists available at ScienceDirect

Future Generation Computer Systems

TruGRC: Trust-Aware Group Recommendation with Virtual Coordinators

Ximeng Wang [a,b], Yun Liu [a,*], Jie Lu [b], Fei Xiong [a], Guangquan Zhang [b]

[a] *Key Laboratory of Communication & Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing 100044, China*
[b] *Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney, NSW 2007, Australia*

HIGHLIGHTS

- We integrate the result and profile aggregation strategies to improve group recommendation.
- We introduce a virtual coordinator to create a balanced set of group recommendations.
- We model trust information and personal influence in group recommender systems.
- Comprehensive experiments indicate the proposed method outperforms most of baselines.

ARTICLE INFO

ABSTRACT

In recent years, an increase in group activities on websites has led to greater demand for highly-functional group recommender systems. The goal of group recommendation is to capture and distill the preferences of each group member into a single recommendation list that meets the needs of all group members. Existing aggregation functions perform well in harmonious and congruent scenarios, but tend not to generate satisfactory results when group members hold conflicting preferences. Moreover, most of current studies improve group recommendation only based on a single aggregation strategy and explicit trust information is still ignored in group recommender systems. Motivated by these concerns, this paper presents TruGRC, a novel T̲rust-aware G̲roup R̲ecommendation method with virtual C̲oordinators, that combines two different aggregation strategies: result aggregation and profile aggregation. As each individual's preferences are modeled, a virtual user is built as a coordinator to represent the profile aggregation strategy. This coordinator provides a global view of the preferences for all group members by interacting with each user to resolve conflicting preferences. Then, we also model the impact from group members to the virtual coordinator in accordance with personal social influence inferred by trust information on social networks. Group preferences can be easily generated by the average aggregation method under the effect of the virtual coordinator. Experimental results on two benchmark datasets with a range of different group sizes show that TruGRC method has significant improvements compared to other state-of-the-art methods.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

Recommender systems have attracted much attention for their ability to model user preferences and generate personalized predictions based on historical behaviors [1]. As such, they have become a useful tool for disposing the information overload problem in e-commerce systems [2]. Most previous studies have focused on personal recommender systems. However, individuals are not isolated entities and they are usually part of some sort of organizations or groups that revolve around shared activities or similar interests [3]. The behaviors of a group of individuals sharing similar interests can be considered as group activities, for example, traveling, seeing movies, and dining out with a number of friends. As group activities on websites increase, many studies on group recommender systems [4–8] have been conducted in recent years to provide recommendations to a given group of users.

Unlike individual recommender systems, group recommender systems often contain a diverse set of preferences and group recommendation is to make a single set of recommendations for a group of users with different preferences. Hence, the challenge in group recommendation is how to integrate individual's preferences into a unique recommendation list which will be satisfied

* Correspondence to: School of Electronic and Information Engineering, Beijing Jiaotong University, No. 3 Shangyuancun, Haidian District, Beijing 100044, China.
E-mail addresses: wangxm@bjtu.edu.cn (X. Wang), liuyun@bjtu.edu.cn (Y. Liu), jie.lu@uts.edu.au (J. Lu), xiongf@bjtu.edu.cn (F. Xiong), guangquan.zhang@uts.edu.au (G. Zhang).

by all group users. There are two main ways for aggregating personal preferences into group preferences: result aggregation [8,9] and profile aggregation [7,10]. Result aggregation applies personal recommendation methods to generate recommendations for every user and then makes all these recommendations into a combined recommendation list for the whole group [11,12]. Studies on these types of group recommendation system mainly focus on enhancing the rationality and effectiveness of aggregation functions to achieve better accuracy [13]. Profile aggregation strategies create a virtual user to represent the combined profile of group members, then predictions are made for the virtual user by applying personal recommendation methods. This alternative is called virtual user-based approaches [6,10]. However, most existing studies consider the above two strategies separately. Very few consider ways to benefit from both. Yet, solely relying on one strategy creates problems in complex recommendation scenarios and, generally, does not produce satisfactory recommendations for every group member. Further, websites and social networks contain a great deal of auxiliary information, e.g., trust links [14], which have not yet been considered in group recommendation. Although some social-aware group recommender systems have been proposed [4,15], but most of them only simply infer users' social relationships and influence through the Thomas–Kilmann conflict mode instrument (TKI) [16]. Actual and explicit trust links between users on social networks are still ignored.

In this paper, we propose a Trust-aware Group Recommendation method with virtual Coordinators (TruGRC), which integrates the benefits of both the result and profile aggregation strategies. Moreover, we introduce personal influence and trust links into group recommendation tasks. Group recommendation processes can be considered as a negotiation in which every member of the group hopes the group's preferences will match their own personal preferences as much as possible. Yet, when group members hold conflicting preferences, it can be difficult to aggregate individual preferences using simple aggregation functions, such as the average and least-misery strategies. Some aggregation functions, such as GROD [8], ASI [4] and MC-GR [7], can increase the accuracy of group recommendations, but these functions are complicated, and their afforded improvement is limited. Hence, to maximize the benefits for all group members, while overcoming conflicts in preferences, it is necessary to introduce a coordinator. In our method, we regard the virtual user as a virtual coordinator that is introduced into the process of modeling each group user's preferences. The virtual coordinator provides a global view of all user preferences and harmonizes their benefits by negotiating with them. These negotiations with the coordinator involve compromise but ultimately generate recommendations that are reasonable to each member of the group. Further, the resulting recommendations can be easily distilled using the average aggregation. Thus, the main contributions of this paper can be summarized as follows:

- We propose TruGRC method that integrates both the result and the profile aggregation strategies. Specifically, we introduce a virtual coordinator into group recommendation, which brings a global perspective for optimizing the evaluation process of individual user preferences and creating a balanced set of group recommendations. With the contribution of the virtual coordinator, applying the average aggregation method to generate a satisfactory recommendation list is easy.
- We introduce trust information into group recommender systems, such as explicit trust links on social networks. The personal influence is inferred from user's trust relations, and each group member impacts the virtual coordinator based on its personal influence. We also consider the interactions between group users to represent the negotiation process.

- We implement extensive experiments to evaluate the proposed TruGRC model. The comparisons between TruGRC and several cutting-edge methods indicate TruGRC outperforms its counterparts in four common evaluation metrics at various group sizes.

The remainder of this paper is organized as follows. We briefly review previous studies in Section 2 from following aspects: collaborative filtering, group recommendation and trust-aware recommendation. Section 3 presents the proposed TruGRC method. Section 4 shows the descriptions of the datasets, evaluation metrics, and experimental results. Section 5 proposes an architecture and potential applications of a trust-aware group recommender system. The conclusion of this paper and future work is given in Section 6.

2. Related work

This section briefly reviews some related studies in three respects: collaborative filtering, methods for group recommendation, and methods for trust-aware recommendation.

2.1. Collaborative filtering

Collaborative filtering (CF) is an extensively explored technology in recommender systems [9,17–19]. The assumption is that a user will make choices that are consistent with its past behaviors and that a user will accept the opinions from other users with similar preferences. CF can be classified into two major aspects: memory-based [20] and model-based [21] approaches. The memory-based methods make predictions for candidate items by combined historical behaviors from users with similar preferences calculated by similarity metrics [17,22,23]. However, this kind of method often demonstrates poor performance in situations with sparse data. By contrast, model-based approaches provide much better performance with the data sparsity problem by applying machine learning theories to make recommendations. Once all parameters have been learned, the models predict candidate items. Matrix factorization (MF) has been extensively used as a model for recommendation tasks because of its good scalability and low complexity [18,24–27]. However, most studies make recommendations for individuals, and they cannot be used effectively to generate recommendations for groups.

2.2. Group recommendation

Group recommender systems are designed to generate recommendations for a set of individuals with diverse interests, which have been implemented in several domains, e.g., tourism [28] and TV programs [29]. The main idea of group recommendation is to aggregate group members' choices and the current group recommendation approaches fall into two categories, i.e., profile aggregation and result aggregation. Profile aggregation builds a virtual user to represent the group profile by combining each individual profile. Ortega et al. [6] applied MF techniques to combine group user profiles in latent feature spaces. Wang et al. [7] proposed a group recommendation model based on member contributions which are evaluated by the degrees of user importance via the separable non-negative matrix factorization technique. Kagita et al. [10] defined a virtual user's profile using precedence relations in a group and regarded it as the group's profile. Compared to profile aggregation, result aggregation often has more flexibility and has attracted more attention. Research on result aggregation mainly focuses on designing better aggregation functions to integrate group members' preferences. Some studies [30,31] have indicated the average aggregation function (AVG) gets the best results among naive aggregation functions. Therefore, applying AVG to combine

Table 1
Comparisons between group recommendation methods.

Type	Method	Aggregation strategy		Feature information	
		Result	Profile	Rating	Trust
Memory-based	UCF [17]-AVG	✓		✓	
	Mono [10]		✓	✓	
	Predict&Cluster [32]	✓		✓	
	PSIE [33]	✓		✓	Implicit
	ASI [4]	✓		✓	Implicit
Model-based	MF [18]-AVG	✓		✓	
	MC-GR [7]		✓	✓	
	GROD [8]	✓		✓	Implicit
	AF [6]	✓		✓	
	BF [6]		✓	✓	
	TruGRC (ours)	✓	✓	✓	Explicit

Table 2
Notations in this work.

Symbol	Description
$\mathcal{U}$	User set in group g, $u \in \mathcal{U}$, $|\mathcal{U}| = m$
$\mathcal{I}$	Item set, $i \in \mathcal{I}$, $|\mathcal{I}| = n$
$\mathcal{R} \in \mathbb{R}^{m \times n}$	User–item rating matrix
$r_{u,i} \in \mathcal{R}$	The observed rating of item i given by user u
$\widehat{r}_{u,i}$	The prediction of user u on item i
$\widehat{r}_{c,i}$	The prediction of virtual coordinator c on item i
$\widetilde{r}_{c,i}$	The interaction representation of virtual coordinator c on item i
$\widehat{r}_{g,i}$	The prediction of group g on item i
d	The dimension of feature space
$\mathbf{p}_u \in \mathbb{R}^{d \times 1}$	Individual user latent feature vector
$\mathbf{p}_c \in \mathbb{R}^{d \times 1}$	Virtual coordinator latent feature vector
$\mathbf{q}_{i1} \in \mathbb{R}^{d \times 1}$	Item–user latent feature vector
$\mathbf{q}_{i2} \in \mathbb{R}^{d \times 1}$	Item–coordinator latent feature vector
b_{i1}	Item–user bias
b_{i2}	Item–coordinator bias
s_u	The personal social impact of user u
$\mathcal{T} \in \mathbb{R}^{m \times m}$	User–user trust matrix
$t_{u,l} \in \mathcal{T}$	The trust relation between user u and user l
$\lambda, \lambda_\alpha, \lambda_\beta$	Trade-off parameters

the results of individual recommendation methods, e.g. UCF [17] and MF [18], can be regarded as baselines in result aggregation. Castro et al. [8] proposed an aggregation strategy that applies opinion dynamics to simulate the information interaction process between group members and, through this opinion exchange, any conflicting preferences between group members are resolved. The clustering group is a typical kind of user groups, Boratto et al. [32] adopted clustering methods to detect user groups and tested plentiful aggregation functions to find out which function will bring the best performance of accuracy in this scenario. Quijano-Sanchez et al. [33] proposed a personalized social individual explanation approach that infers social relations from demographic information of group members. Guo et al. [4] introduced a computational model to integrate the influence of personality, expertise factor and preference similarities, and demonstrated considering social influence can improve the quality of group recommendation. Some other models to diminish the negative effects of natural noise in group recommender systems were also been studied [5,34]. Comparisons on aggregation strategies and feature information used in related works are presented in Table 1. It can be seen from Table 1, previous studies only consider a single aggregation strategy, but the proposed TruGRC takes advantage of both result and profile aggregation strategies, and also introduces explicit trust information into group recommendation tasks.

2.3. Trust-aware recommendation

With the evolution of social networks, the studies of trust-aware recommendation are increasing in recent years [35–37]. Trust relations can be used for modeling user preferences from a perspective other than item ratings, which solves some inherent difficulties with current recommender systems, e.g., data sparsity [38] and the cold-start problem [39]. Trust-aware recommender systems are mainly based on MF because of its remarkable expandability. Ma et al. [14] adopted the regularization terms of social relations to introduce the concept of social constraint into recommendation tasks. An extension to this idea was proposed in [40] that leverages the users' implicit social relationships to improve recommendations. Guo et al. [26] proposed the TrustSVD model that brings multiple information for personalized recommender systems and considers trust relations between users from the explicit and implicit feedbacks. Liu et al. [41] improved recommendation accuracy by generating domain-specific trust networks that include direct and indirect relations. Overall, trust has proven to be a very useful feature in recommender systems. However, explicit trust relationships have not yet been considered in group recommendation tasks.

3. Trust-aware group recommendation with virtual coordinators

To formalize the group recommendation problem, the user set of group g is defined as $\mathcal{U}$ and the set of items is defined as $\mathcal{I}$, where $u \in \mathcal{U}$ means each individual in group g, and $i \in \mathcal{I}$ is each item in the item set. We define $|\mathcal{U}| = m$ as the size of the user group and $|\mathcal{I}| = n$ as the number of all items. In this group recommendation scenario, only the users in a specific group are considered, and each group has no impact on another. The user–item rating matrix is denoted as $\mathcal{R} \in \mathbb{R}^{m \times n}$, and the observed rating of user u on item i is defined as $r_{u,i} \in \mathcal{R}$. The aggregation function is the vital part of integrating users' preferences, defined as:

$$\widehat{r}_{g,i} = \Phi(\widehat{r}_{u,i}) \tag{1}$$

where $\Phi(\cdot)$ is the aggregation function that generates the group preferences from each user's preferences. For item i, $\widehat{r}_{g,i}$ and $\widehat{r}_{u,i}$ are the combined rating of group g and the predicted rating of user u, respectively. For the group, a recommendation list is proposed in accordance with the top-L highest scores of $\widehat{r}_{g,i}$, which is defined as:

$$top(g, L) = \arg \max_{i \in \mathcal{I}}^{L} \widehat{r}_{g,i} \tag{2}$$

The full set of notations is given in Table 2.

3.1. Overall framework

This section introduces the framework of the proposed method which improves group recommendation by harmonizing each user's preferences and considering the trust information. We can regard group recommendation as a negotiation process [42] where the aim is to generate a single recommendation list that meets the requirements of most users in the group. In our framework, this is accomplished by integrating a virtual coordinator into a traditional group recommendation framework. This virtual coordinator plays a significant role in the negotiation between individuals, balancing individual preferences with group preferences. An overview of the framework appears in Fig. 1. We generate each individual user's preferences from the ratings they have already marked over items. In addition, we assume the virtual coordinator can observe all the historical ratings of users in the group and its preferences are generated from all group users' ratings. The virtual coordinator provides an overall perspective for modeling and modifying individual preferences. However, each group member has a level of influence
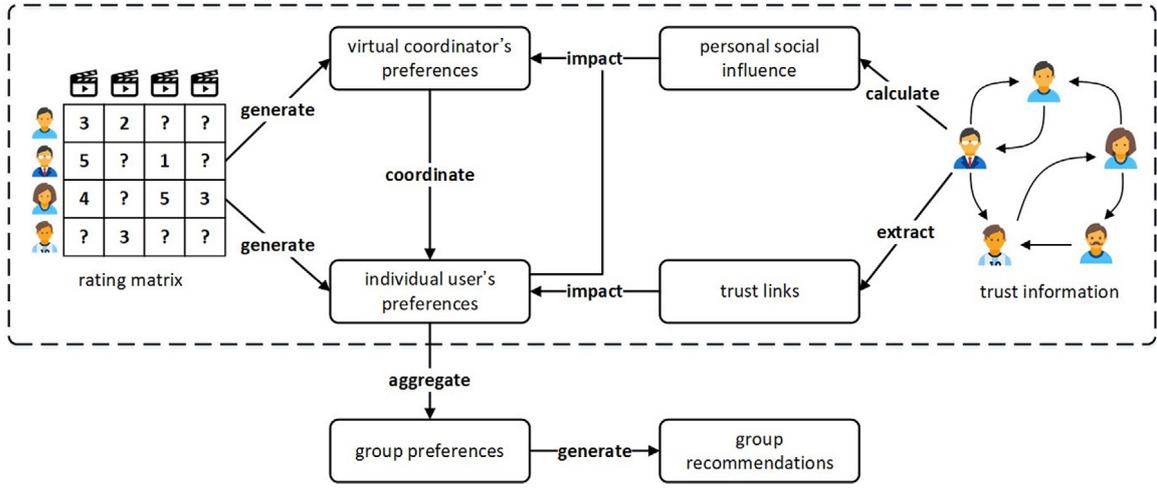
Fig. 1. The framework of the proposed TruGRC method. A virtual coordinator is introduced to harmonize each individual user's preferences in the group, and each group member simultaneously impacts the virtual coordinator's preferences according to its personal social inference calculated by trust information. Group recommendations are generated in accordance with group preferences aggregated by each group member's preferences.

over the virtual coordinator, which is inferred from its influence on trust networks. And this influence can affect negotiations with the virtual coordinator. Group preferences will be generated by simply aggregating each user's preferences once negotiations are complete, and then a group recommendation list will be proposed.

3.2. The individual recommendation method

Given the MF model [18] has demonstrated good performance for modeling user preferences and predicting missing ratings, we have selected this model as the individual recommendation method for our framework. The MF model applies latent feature vectors to represent the preferences of users and items, then the ratings can be fairly estimated from these latent feature vectors, defined as:

$$\widehat{r}_{u,i} = b_{i1} + \mathbf{p}_u^{\mathsf{T}}\mathbf{q}_{i1} \tag{3}$$

where $\widehat{r}_{u,i}$ is the predicted rating that user u gives item i, $\mathbf{p}_u \in \mathbb{R}^{d\times 1}$ is the user-specific latent feature vector, $\mathbf{q}_{i1} \in \mathbb{R}^{d\times 1}$ is the item-specific latent feature vector, d is the dimension of latent features, and b_{i1} is the bias of item i. In recommendation, there is an important task which is to ensure the predicted rating $\widehat{r}_{u,i}$ is as close to the observed rating $r_{u,i}$ as possible [26]. To accomplish this goal, following the MF model [18], the latent feature vectors and the bias can be learned by minimizing the following loss function:

$$\min_{\mathbf{p}_u,\mathbf{q}_{i1},b_{i1}} \mathcal{L}_u = \frac{1}{2}\sum_{u=1}^{m}\sum_{i=1}^{n}\delta_{u,i}\left(r_{u,i}-\widehat{r}_{u,i}\right)^2 + \frac{\lambda}{2}\left(\sum_{u=1}^{m}||\mathbf{p}_u||_F^2\right.$$
$$\left. + \sum_{i=1}^{n}||\mathbf{q}_{i1}||_F^2 + \sum_{i=1}^{n}b_{i1}^2\right) \tag{4}$$

where $\delta_{u,i}$ represents an indicator that $\delta_{u,i} = 1$ means $r_{u,i}$ is observed, and $\delta_{u,i} = 0$ otherwise. $||\cdot||_F$ is the Frobenius norm to avoid over-fitting and λ is a trade-off parameter to regulate the impact of the regularization terms.

In term of the definition in Eq. (1), the results of group recommendation is generated by aggregating every individual's predictions in the group. Previous studies [6,8] have mainly focused on improving the aggregation functions. However, they have neglected to adjust the individual's preferences to suit the final set of group recommendations. By contrast, we propose a new group recommendation model that introduces a virtual coordinator to

enhance performance and considers the trust information associated with each individual. This model is presented in the next sections.

3.3. Modeling the virtual coordinator in the group

Group recommendation can be regarded as a negotiation process, where every group member wants to maximize its benefits. However, usually, the results of any negotiation do not entirely meet the needs of all group members. Hence, the coordinator plays an important role in harmonizing each member's opinions and requirements to ensure the benefits are spread across the whole group. Based on this idea, the aim is to model a virtual coordinator that alters the preference estimations for each group member when making predictions. We assume that the virtual coordinator is capable of observing the historical data of all group members, e.g., rating information or purchasing records, so it can form a global perspective on the preferences of the entire group. This approach is unlike typical virtual user-based group recommendation methods [10] where the virtual user only makes recommendations based on its own profile. Whereas, our model combines multiple sources of feedback from both group members and the virtual coordinator to enhance the accuracy of group recommendation.

Similar to individual recommendation methods, we also map the virtual coordinator and each item into the same feature space, defined as $\mathbf{p}_c \in \mathbb{R}^{d\times 1}$ and $\mathbf{q}_{i2} \in \mathbb{R}^{d\times 1}$, respectively. Therefore, the predicted ratings of the virtual coordinator can be defined as:

$$\widehat{r}_{c,i} = b_{i2} + \mathbf{p}_c^{\mathsf{T}}\mathbf{q}_{i2} \tag{5}$$

where $\widehat{r}_{c,i}$ denotes the predicted rating of the virtual coordinator c and b_{i2} is the bias. $\mathbf{p}_c$, $\mathbf{q}_{i2}$ and b_{i2} are learned through a square loss function because the virtual coordinator needs to observe all group members' ratings. Therefore, the loss function is defined as:

$$\min_{\mathbf{p}_c,\mathbf{q}_{i2},b_{i2}} \mathcal{L}_c = \frac{1}{2}\sum_{u=1}^{m}\sum_{i=1}^{n}\delta_{u,i}\left(r_{u,i}-\widehat{r}_{c,i}\right)^2$$
$$+ \frac{\lambda}{2}\left(||\mathbf{p}_c||_F^2 + \sum_{i=1}^{n}||\mathbf{q}_{i2}||_F^2 + \sum_{i=1}^{n}b_{i2}^2\right) \tag{6}$$

Be different from Eq. (4), Eq. (6) is used to learn the preferences of the virtual coordinator c rather than individual's preferences.

Moreover, the virtual coordinator's latent feature vector $\mathbf{p}_c$ interacts with all latent feature vectors of items collected by group members in Eq. (6), whereas in Eq. (4), users only interact with items rated by themselves. Eqs. (4) and (6) are combined linearly to construct a new loss function that integrates multiple feedbacks as:

$$
\min_{\Theta} \mathcal{L} = \mathcal{L}_u + \mathcal{L}_c
$$

$$
= \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} \delta_{u,i} \left(r_{u,i} - \widehat{r}_{u,i} \right)^2 + \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} \delta_{u,i} \left(r_{u,i} - \widehat{r}_{c,i} \right)^2
$$

$$
+ \frac{\lambda}{2} \left(\sum_{u=1}^{m} ||\mathbf{p}_u||_F^2 + ||\mathbf{p}_c||_F^2 + \sum_{i=1}^{n} ||\mathbf{q}_{i1}||_F^2 \right.
$$

$$
\left. + \sum_{i=1}^{n} ||\mathbf{q}_{i2}||_F^2 + \sum_{i=1}^{n} b_{i1}^2 + \sum_{i=1}^{n} b_{i2}^2 \right) \tag{7}
$$

where $\Theta = \{b_{i1},\ b_{i2},\ \mathbf{p}_u,\ \mathbf{p}_c,\ \mathbf{q}_{i1},\ \mathbf{q}_{i2}\}$. Two latent feature vectors have been defined for each item, i.e., $\mathbf{q}_{i1}$ and $\mathbf{q}_{i2}$, along with two biases for each item, i.e., b_{i1} and b_{i2}. These vectors are used to make predictions for individuals and the virtual coordinator in Eq. (3) and Eq. (5), respectively. Although $\mathbf{q}_{i1}$ and $\mathbf{q}_{i2}$ serve for different objects, they indicate the latent features of the identical item, so they need to have the same intrinsic properties. Hence, a regularization term constrains these two latent feature vectors as follows:

$$
\frac{1}{2} \sum_{i=1}^{n} ||\mathbf{q}_{i1} - \mathbf{q}_{i2}||_F^2 \tag{8}
$$

Once the virtual coordinator has been modeled, the interactions between it and the other group members need to be modeled. The virtual coordinator plays a coordinating role in modeling each user's preferences based on its global perspective. That is, each user's preferences are adjusted so that subsequent predictions create a balance between each user and the entire group. In order to model these interactions, we use $\widetilde{r}_{c,i}$ to define another similar representation of the virtual coordinator's predictions as follows:

$$
\widetilde{r}_{c,i} = b_{i1} + \mathbf{p}_c^\mathsf{T} \mathbf{q}_{i1} \tag{9}
$$

where $\mathbf{q}_{i1}$ is the item-specific latent feature vector for making the individual's predictions in Eq. (3). This representation can be regarded as the interactions between $\mathbf{p}_c$ and $\mathbf{q}_{i1}$ in the latent feature space, which affects $\mathbf{q}_{i1}$ with the virtual coordinator's impact. In addition, a constraint is placed on the two representations of the virtual coordinator's predictions. This constraint is defined as:

$$
\frac{1}{2} \sum_{i=1}^{n} \left(\widehat{r}_{c,i} - \widetilde{r}_{c,i} \right)^2 \tag{10}
$$

where the distance between these two representations should be as short as possible because they have the same goal of modeling the virtual coordinator's preferences. These two regularization terms are incorporated into the loss function as:

$$
\min_{\Theta} \mathcal{L} = \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} \delta_{u,i} \left(r_{u,i} - \widehat{r}_{u,i} \right)^2 + \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} \delta_{u,i} \left(r_{u,i} - \widehat{r}_{c,i} \right)^2
$$

$$
+ \frac{\lambda_\alpha}{2} \sum_{i=1}^{n} ||\mathbf{q}_{i1} - \mathbf{q}_{i2}||_F^2 + \frac{\lambda_\alpha}{2} \sum_{i=1}^{n} \left(\widehat{r}_{c,i} - \widetilde{r}_{c,i} \right)^2
$$

$$
+ \frac{\lambda}{2} \left(\sum_{u=1}^{m} ||\mathbf{p}_u||_F^2 + ||\mathbf{p}_c||_F^2 + \sum_{i=1}^{n} ||\mathbf{q}_{i1}||_F^2 \right.
$$

$$
\left. + \sum_{i=1}^{n} ||\mathbf{q}_{i2}||_F^2 + \sum_{i=1}^{n} b_{i1}^2 + \sum_{i=1}^{n} b_{i2}^2 \right) \tag{11}
$$

where λ_α is a trade-off parameter to regulate the impact of above two item-related regularization terms.

3.4. Trust-aware group recommendation with the virtual coordinator

Trust is an important feature on social networks as it indicates the relationships between users. In practice, asymmetric trust is more general than symmetric trust [43]. For example, one user following another user on Twitter can be seen as a trust link between these two users, but an asymmetric one, because the trust is not mutual. Several studies on trust-aware recommendation for individuals have been conducted [14,26,43], but, to date, trust information has not been exploited in group recommendation systems.

During the negotiation process, the coordinator needs to communicate with group members, which means each individual is not only managed, they also have an impact on the coordinator. Based on the assumption that the virtual coordinator can be affected by the group members, the impact of each user is defined by their personal social influence. Determining a user's influence is similar to identifying the vital nodes in a complex network. Neighborhood-based centralities, e.g., degree centrality, localRank, and coreness, are widely used for identifying vital nodes because of their low computational complexity [44]. Here, we use degree centrality to express the personal impact of every user on social networks, defined as:

$$
dc(u) = \frac{k_u}{m-1} \tag{12}
$$

where the range of $dc(u)$ is from 0 to 1, k_u indicates the degree of user u, and $m-1$ is the largest possible degree. Further, in a group, personal influence is not absolute but rather relative. It depends on a comparison with the influence held by other group members. Therefore, to properly evaluate personal influence within a group, the degree centrality must be normalized. The normalizing function is defined as:

$$
s_u = \frac{dc(u)}{\sum_{u=1}^{m} dc(u)} \tag{13}
$$

where $s_u > 0$ indicates user u's personal influence, which is captured and calculated by trust information. $s_u = 0$ is caused by $dc(u) = 0$, which means user u does not have any trust links on the network. Hence, for a user with no observable trust links, the value of personal influence is randomly generated. In general, users with high personal influence, such as actors or public figures, will be very active on social networks. Therefore, when $dc(u) = 0$, s_u is set as a random number $r \in (0, 0.5]$ because it is highly likely that a user with no trust links will not have great influence. Thus, s_u is altered as follows:

$$
s_u = \begin{cases} \dfrac{dc(u)}{\sum_{u=1}^{m} dc(u)}, & dc(u) > 0 \\ r \in (0, 0.5], & dc(u) = 0 \end{cases} \tag{14}
$$

With personal influence defined, each group member impacts the virtual coordinator according to its influence level, as defined below:

$$
\frac{1}{2} \sum_{u=1}^{m} s_u ||\mathbf{p}_u - \mathbf{p}_c||_F^2 \tag{15}
$$

In addition, the interactions between two group members are modeled based on their trust links on social networks. If user l follows user u, a trust link $t_{u,l}$ between them is identified. This interaction is defined as:

$$
\frac{1}{2} \sum_{u=1}^{m} \sum_{l=1}^{m} t_{u,l} ||\mathbf{p}_u - \mathbf{p}_l||_F^2 \tag{16}
$$

Mathematically, Eqs. (15) and (16) introduce two constraints into the loss function that minimize the distance between two parameters. Effectively, this forces the preferences of the two users to

be close together. This makes sense because each user wants the virtual coordinator to represent its preferences better and one user will probably trust another user with similar preferences. Therefore, Eqs. (15) and (16) are combined with Eq. (11) to generate the final loss function as follows:

$$
\begin{aligned}
\min_{\Theta} \mathcal{L} = {} & \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} \delta_{u,i} \left(r_{u,i} - \widehat{r}_{u,i} \right)^2 + \frac{1}{2} \sum_{u=1}^{m} \sum_{i=1}^{n} \delta_{u,i} \left(r_{u,i} - \widehat{r}_{c,i} \right)^2 \\
& + \frac{\lambda_{\alpha}}{2} \sum_{i=1}^{n} ||\mathbf{q}_{i1} - \mathbf{q}_{i2}||_F^2 + \frac{\lambda_{\alpha}}{2} \sum_{i=1}^{n} \left(\widehat{r}_{c,i} - \widetilde{r}_{c,i} \right)^2 \\
& + \frac{\lambda_{\beta}}{2} \sum_{u=1}^{m} s_u ||\mathbf{p}_u - \mathbf{p}_c||_F^2 + \frac{\lambda_{\beta}}{2} \sum_{u=1}^{m} \sum_{l=1}^{m} t_{u,l} ||\mathbf{p}_u - \mathbf{p}_l||_F^2 \\
& + \frac{\lambda}{2} \left(\sum_{u=1}^{m} ||\mathbf{p}_u||_F^2 + ||\mathbf{p}_c||_F^2 + \sum_{i=1}^{n} ||\mathbf{q}_{i1}||_F^2 \right. \\
& \left. + \sum_{i=1}^{n} ||\mathbf{q}_{i2}||_F^2 + \sum_{i=1}^{n} b_{i1}^2 + \sum_{i=1}^{n} b_{i2}^2 \right)
\end{aligned}
\tag{17}
$$

where λ_{β} is a trade-off parameter to regulate the impact of above two trust-related regularization terms.

3.5. Learning and prediction

To learn the parameters Θ of our proposed model in Eq. (17), we use the gradient descent to reach a local minimization of the loss function. Gradient descent is an effective way for minimization when objective functions are differentiable and non-convex, and is also the most commonly used algorithm in MF-based recommender systems [18,24,26,38,40,45]. The gradients of the parameters Θ are performed as follows:

$$
\frac{\partial \mathcal{L}}{\partial b_{i1}} = \sum_{u=1}^{m} \delta_{u,i} \left(\widehat{r}_{u,i} - r_{u,i} \right) + \lambda_{\alpha} \left(\widetilde{r}_{c,i} - \widehat{r}_{c,i} \right) + \lambda b_{i1}
$$

$$
\frac{\partial \mathcal{L}}{\partial b_{i2}} = \sum_{u=1}^{m} \delta_{u,i} \left(\widehat{r}_{c,i} - r_{u,i} \right) + \lambda_{\alpha} \left(\widehat{r}_{c,i} - \widetilde{r}_{c,i} \right) + \lambda b_{i2}
$$

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \mathbf{p}_u} = {} & \sum_{i=1}^{n} \delta_{u,i} \left(\widehat{r}_{u,i} - r_{u,i} \right) \mathbf{q}_{i1} \\
& + \lambda_{\beta} s_u \left(\mathbf{p}_u - \mathbf{p}_c \right) + \lambda_{\beta} \sum_{l=1}^{m} t_{u,l} \left(\mathbf{p}_u - \mathbf{p}_l \right) + \lambda \mathbf{p}_u
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \mathbf{q}_{i1}} = {} & \sum_{u=1}^{m} \delta_{u,i} \left(\widehat{r}_{u,i} - r_{u,i} \right) \mathbf{p}_u \\
& + \lambda_{\alpha} \left(\mathbf{q}_{i1} - \mathbf{q}_{i2} \right) + \lambda_{\alpha} \left(\widetilde{r}_{c,i} - \widehat{r}_{c,i} \right) \mathbf{p}_c + \lambda \mathbf{q}_{i1}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \mathbf{p}_c} = {} & \sum_{i=1}^{n} \delta_{u,i} \left(\widehat{r}_{c,i} - r_{u,i} \right) \mathbf{q}_{i2} \\
& + \lambda_{\alpha} \left(\widehat{r}_{c,i} - \widetilde{r}_{c,i} \right) \left(\mathbf{q}_{i2} - \mathbf{q}_{i1} \right) + \lambda_{\beta} s_u \left(\mathbf{p}_c - \mathbf{p}_u \right) + \lambda \mathbf{p}_c
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \mathbf{q}_{i2}} = {} & \sum_{u=1}^{m} \delta_{u,i} \left(\widehat{r}_{c,i} - r_{u,i} \right) \mathbf{p}_c \\
& + \lambda_{\alpha} \left(\mathbf{q}_{i2} - \mathbf{q}_{i1} \right) + \lambda_{\alpha} \left(\widehat{r}_{c,i} - \widetilde{r}_{c,i} \right) \mathbf{p}_c + \lambda \mathbf{q}_{i2}
\end{aligned}
\tag{18}
$$

The pseudocode for learning the parameters is provided in Algorithm 1. The inputs include the user–item rating matrix $\mathcal{R}$, the user–user trust matrix $\mathcal{T}$, the dimension of the feature space d, the regularization parameters λ, λ_{α} and λ_{β}, and the learning rate η. The parameters Θ form the output of the algorithm.

Algorithm 1 Learning the parameters in TruGRC method

Input: $\mathcal{R}, \mathcal{T}, d, \lambda, \lambda_{\alpha}, \lambda_{\beta}, \eta$
Output: $\Theta = \left\{ b_{i1}, b_{i2}, \mathbf{p}_u, \mathbf{p}_c, \mathbf{q}_{i1}, \mathbf{q}_{i2} \right\}$
1: Initialize the parameters $\Theta \sim N(0, 0.01)$
2: Calculate the personal influence s_u according to Eq. (14)
3: **while** $\mathcal{L}$ is not coveraged **do**
4:      Calculate gradients according to Eq. (18)
5:      $b_{i1} \leftarrow b_{i1} - \eta \frac{\partial \mathcal{L}}{\partial b_{i1}}$
6:      $b_{i2} \leftarrow b_{i2} - \eta \frac{\partial \mathcal{L}}{\partial b_{i2}}$
7:      $\mathbf{p}_u \leftarrow \mathbf{p}_u - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{p}_u}$
8:      $\mathbf{q}_{i1} \leftarrow \mathbf{q}_{i1} - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{q}_{i1}}$
9:      $\mathbf{p}_c \leftarrow \mathbf{p}_c - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{p}_c}$
10:     $\mathbf{q}_{i2} \leftarrow \mathbf{q}_{i2} - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{q}_{i2}}$
11: **end while**
12: **return** Θ

Table 3
Statistics of datasets.

	Ciao	Epinions
# Users	2960	5155
# Items	4394	3432
# Ratings	86,990	164,994
Rating sparsity	6.69×10^{-3}	9.33×10^{-3}
# Trust links	56,988	133,605

Once the parameters Θ are optimized, each user's predictions are generated using Eq. (3) and the group's predictions are generated using Eq. (1). Note that, in this paper, the average aggregation method has been used as the aggregation function $\Phi(\cdot)$. The list of recommendations for the whole group is then arranged using Eq. (2).

3.6. Complexity analysis

Most of the computation complexity lies in optimizing the loss function and calculating the corresponding gradients. The time to compute the loss function $\mathcal{L}$ is $O(2d\,|\mathcal{R}|)$, where d is the dimension of feature space and $|\mathcal{R}|$ is the number of observed records. The value for $|\mathcal{R}|$ should be much smaller than the matrix cardinality because of the sparsity. Further, the time to compute the gradients in Eq. (18) also needs to be considered. The complexity of $\frac{\partial \mathcal{L}}{\partial \mathbf{p}_u}$ is $O(d\,|\mathcal{R}| + d\,|\mathcal{T}|)$, and the other gradients have the same complexity, i.e., $O(d\,|\mathcal{R}|)$. Because $|\mathcal{T}|$ is usually much smaller than $|\mathcal{R}|$, the overall computational complexity for each iteration is $O(2d\,|\mathcal{R}|)$. It follows that the computational time of our model is linear with respect to the number of observed records in the rating matrix and, therefore, has the potential to be used with large-scale datasets.

4. Experimental results

In this section, we introduce datasets and evaluation metrics at first, followed by the descriptions of baselines and parameter settings. We then compare our method with all the baselines and finally analyze the impact of parameters.

4.1. Datasets and metrics

We select two public available datasets [46] collected from product review websites: Ciao [47] and Epinions [48] to evaluate the proposed method. These two datasets have been widely used in social recommendation [45,49]. Table 3 lists the statistics for these two datasets. A five-fold cross-validation [50] is utilized in

our experiments. Specifically, we randomly split each dataset into five folds. In each iteration, four folds are used as the training set, with the remaining fold treated as the testing set.

Our experiments focus on occasional groups where the members have no explicitly shared preference relevance [8]. Occasional groups are practical because they appear in many application scenarios, such as recommendations for tour groups. Most previous studies have paid attention to small groups that usually contain less than 20 members [5,6,51]. In these situations, opinions are relatively easy to reach a consensus. However, it is essential to assess the feasibility of methods in large groups. Therefore, we randomly form groups of different sizes from 10 to 50 with an interval of 10 and test our proposed method on each group size.

Following previous studies [3,6], four common evaluation metrics are utilized in our experiments to evaluate the proposed method and baselines, including precision [6], recall [3], F1 [43], and mean reciprocal rank [52].

According to precision and recall, when the recommendation length is L, then $Pre@L$ and $Rec@L$ are defined as:

$$Pre@L = \frac{1}{m}\left(\sum_{u=1}^{m} \frac{D_u(L)}{L}\right) \qquad (19)$$

$$Rec@L = \frac{1}{m}\left(\sum_{u=1}^{m} \frac{D_u(L)}{C_u(L)}\right) \qquad (20)$$

where $D_u(L)$ denotes the number of recommended items collected by user u and $C_u(L)$ means the number of items collected in the test set.

F1 is a comprehensive metric, defined as:

$$F1@L = \frac{2Pre@L \times Rec@L}{Pre@L + Rec@L} \qquad (21)$$

Mean Reciprocal Rank (MRR) is given by:

$$MRR = \frac{1}{m}\left(\sum_{u=1}^{m} \sum_{i \in C(u)} \frac{1}{pos_i^u}\right) \qquad (22)$$

where pos_i^u is the recommendation position of item i. A larger MRR value means better performance.

4.2. Baselines and parameter settings

To demonstrate the improvements made by the proposed method, we compare its performance with several representative baselines. These baselines span both similarity-based and matrix factorization-based group recommendation models:

User-based CF with the averaging strategy (UCF-AVG): UCF-AVG employs the user-based CF method [53] to make a predicted score of each item for every user in the group. Then the average aggregation function is used to generate a group recommendation score for each item. Cosine measurement is used to estimate the similarity between users.

User-based CF with the least-misery strategy (UCF-LM): Similar to UCF-AVG, UCF-LM also adopts the user-based CF method to compute estimated scores for items. However, we take the lowest predicted score as the group recommendation score for every item across all users in the least-misery strategy.

Matrix factorization with the averaging strategy (MF-AVG): MF-AVG employs the popular MF model [18] to produce individual recommendations. Then, we generate group recommendations by the average aggregation function.

Matrix factorization with the least-misery strategy (MF-LM): MF-LM also uses the MF model to make predictions for each group member, but this model employs the least-misery strategy instead

Table 4
The parameter settings of MF-based methods.

Method	Parameter	Ciao	Epinions	Descriptions
MF-AVG	η	0.01	0.01	Learning rate
MF-LM	λ	0.01	0.01	Avoiding over-fitting
AF	η	0.01	0.01	Learning rate
	λ	0.01	0.001	Avoiding over-fitting
BF	η	0.001	0.001	Learning rate
	λ	0.01	0.01	Avoiding over-fitting
TruGRC	η	0.001	0.001	Learning rate
	λ	0.01	0.001	Avoiding over-fitting
	λ_α	1	1	Controlling item-related regularization
	λ_β	0.01	0.1	Controlling trust-related regularization

Table 5
The comparisons between TruGRC and all baselines in Ciao with the group sizes of 10 and 20.

Group size = 10	Pre@5	Pre@10	Rec@5	Rec@10	F1@5	F1@10	MRR
UCF-AVG	0.0078	0.0080	0.0064	0.0133	0.0070	0.0100	0.0439
UCF-LM	0.0073	0.0076	0.0062	0.0126	0.0067	0.0095	0.0407
MF-AVG	0.0135	0.0099	0.0115	0.0163	0.0124	0.0123	0.0565
MF-LM	0.0105	0.0088	0.0084	0.0158	0.0094	0.0113	0.0500
AF	0.0135	0.0104	0.0113	0.0173	0.0123	0.0130	0.0573
BF	0.0157	0.0110	0.0125	0.0176	0.0139	0.0135	0.0593
TruGRC	**0.0160**	**0.0121**	**0.0137**	**0.0199**	**0.0147**	**0.0150**	**0.0620**
Group size = 20	Pre@5	Pre@10	Rec@5	Rec@10	F1@5	F1@10	MRR
UCF-AVG	0.0057	0.0055	0.0050	0.0094	0.0054	0.0069	0.0323
UCF-LM	0.0047	0.0050	0.0042	0.0089	0.0044	0.0089	0.0296
MF-AVG	0.0229	0.0182	0.0213	0.0340	0.0221	0.0237	0.0841
MF-LM	0.0180	0.0139	0.0165	0.0250	0.0172	0.0179	0.0742
AF	0.0224	0.0169	0.0205	0.0305	0.0214	0.0218	0.0814
BF	0.0244	0.0184	0.0213	0.0329	0.0228	0.0236	0.0881
TruGRC	**0.0255**	**0.0194**	**0.0233**	**0.0356**	**0.0244**	**0.0251**	**0.0940**

of the averaging strategy when aggregating the group recommendations.

After factorization approach (AF) [6]: AF computes a group latent feature vector by combining all the user-specific vectors of group members. Then, group recommendations are made for all items through the group vector dot products with every item vector.

Before factorization approach (BF) [6]: BF models a group of users by building a virtual user that represents the item preferences of the users in the group. The recommendations for the virtual user are then used as the group recommendations through the MF model.

TruGRC: This is our proposed MF-based group recommendation method as demonstrated in Fig. 1. TruGRC method incorporates trust information into group recommendation tasks and models a virtual coordinator to make a balance of the preferences between each user and the entire group.

To propose an equitable comparison, the latent features are set to the same dimension $d = 10$ for all MF-based methods. In addition, we adopt the cross validation to determine the optimal parameter values of each MF-based method in Ciao and Epinions datasets reported in Table 4 where MF-AVG and MF-LM share the same parameter settings, because they are both based on the same MF model [18].

4.3. Performance evaluation

Tables 5 and 6 report the comparisons of $Pre@5$, $Pre@10$, $Rec@5$, $Rec@10$, $F1@5$, $F1@10$ and MRR for the proposed TruGRC and all baseline methods with the group sizes of 10 and 20. According to the results, TruGRC demonstrates the best performance according to most metrics. Specifically, in Ciao, compared to BF, AF, MF-AVG and MF-LM, the proposed TruGRC shows an improvement of
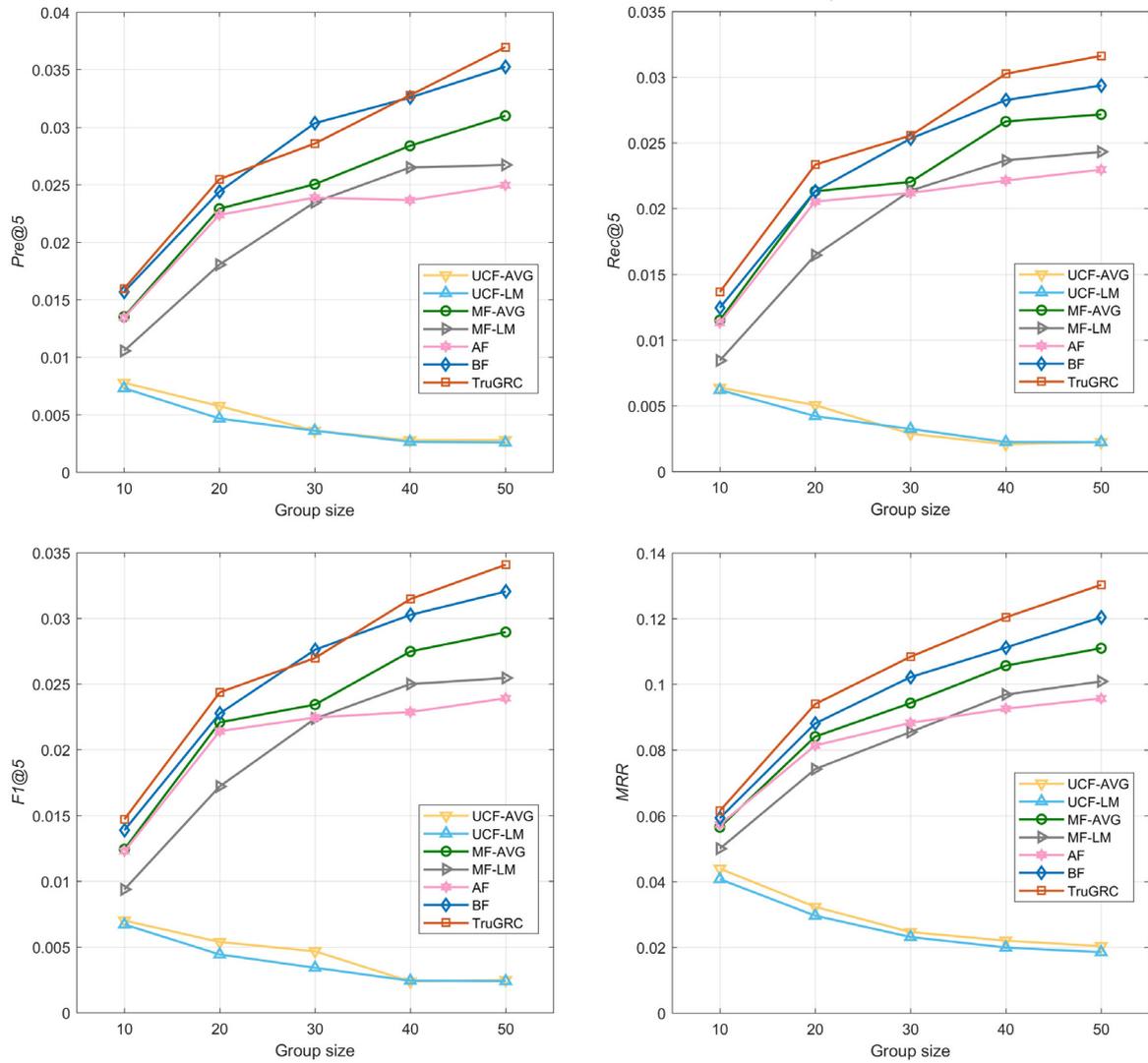
Fig. 2. The comparisons on *Pre*@5, *Rec*@5, *F*1@5 and *MRR* with different group sizes in Ciao.

Table 6
The comparisons between TruGRC and all baselines in Epinions when the group sizes are 10 and 20.

Group size = 10	Pre@5	Pre@10	Rec@5	Rec@10	F1@5	F1@10	MRR
UCF-AVG	0.0075	**0.0076**	0.0062	0.0123	0.0068	0.0093	0.0378
UCF-LM	0.0078	**0.0076**	0.0062	0.0120	0.0069	0.0093	0.0381
MF-AVG	**0.0092**	0.0070	**0.0081**	0.0124	**0.0087**	0.0090	0.0433
MF-LM	0.0073	0.0061	0.0067	0.0100	0.0070	0.0076	0.0393
AF	0.0084	0.0072	0.0074	0.0125	0.0078	0.0092	0.0426
BF	0.0084	0.0069	0.0076	0.0114	0.0080	0.0086	0.0408
TruGRC	0.0091	**0.0076**	0.0081	**0.0133**	0.0086	**0.0097**	**0.0440**
Group size = 20	Pre@5	Pre@10	Rec@5	Rec@10	F1@5	F1@10	MRR
UCF-AVG	0.0045	0.0060	0.0036	0.0104	0.0040	0.0076	0.0330
UCF-LM	0.0052	0.0064	0.0042	0.0104	0.0046	0.0079	0.0331
MF-AVG	0.0123	0.0103	0.0099	0.0180	0.0110	0.0131	0.0564
MF-LM	0.0105	0.0089	0.0090	0.0149	0.0097	0.0112	0.0499
AF	0.0110	0.0097	0.0090	0.0163	0.0099	0.0122	0.0541
BF	**0.0128**	0.0102	0.0107	0.0169	**0.0117**	0.0127	0.0576
TruGRC	**0.0128**	**0.0107**	**0.0108**	**0.0182**	**0.0117**	**0.0135**	**0.0584**

10%, 16%, 22% and 38% on *Pre*@10 and 13%, 15%, 22% and 26% on *Rec*@10 with a group size of 10. In addition, TruGRC also enhances *F*1@10 with 11% and 22% over BF and MF-AVG which are the most competitive methods in baselines. *MRR* is a comprehensive metric that tests the accuracy of the whole recommendation list. TruGRC

improves upon BF and MF-AVG in this metric by 5% and 10%, respectively. Similar improvements are observed with the group size of 20 in Ciao. In Epinions, MF-AVG has the best performance with the group size of 10 on *Pre*@5 and *F*1@5 because consistency is easy to achieve with the average aggregation in a small group size and a short recommendation list. Beyond these two metrics, TruGRC obtains the best overall results in comparison to the other baselines. With the group size of 20, BF surpasses MF-AVG in most metrics, which means that BF is more suitable for larger groups. Even so, TruGRC shows further improvements over BF at 5%, 8% and 6% on *Pre*@10, *Rec*@10 and *F*1@10. Fewer improvements are found with Epinions than Ciao because the Epinions dataset has more users, which leads to a higher probability of preference conflicts in occasional groups.

We also evaluate TruGRC with other group sizes, ranging from 10 to 50. Fig. 2 reports the results for all four metrics with all methods in Ciao. It is interesting to note that for UCF-AVG and UCF-LM, the performance declines as the group size increases, which means these kinds of methods are not suitable for scenarios with large groups. By contrast, the MF-based methods indicate much better performance in large group scenarios. The difference in accuracy for each MF-based method is not obvious, when the group size is small. However, these differences become more distinct as the group size grows. Although TruGRC yields to BF on *Pre*@5 and *F*1@5
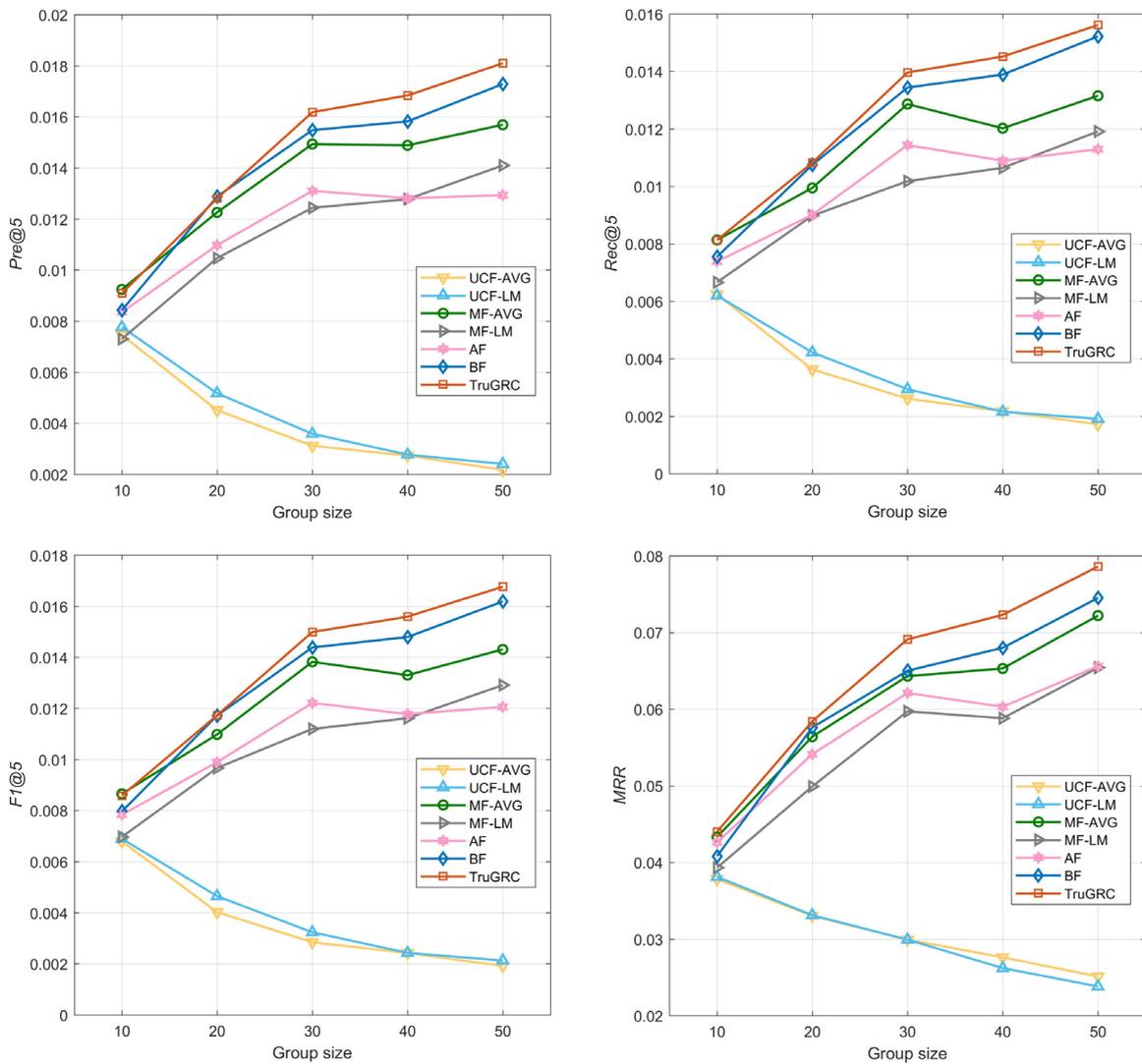
Fig. 3. The comparisons on *Pre*@5, *Rec*@5, *F*1@5 and *MRR* with different group sizes in Epinions.

with the group size of 30, it produces the best performance for most other metrics at different group sizes. In terms of *MRR*, TruGRC consistently demonstrates the best results. Specifically, it shows an improvement of 8% and 14% over BF and MF-AVG with the group size of 40 and an improvement of 8% and 17% with the group size of 50. Similar results are apparent in Fig. 3. Compared to BF and MF-AVG, TruGRC enhances *MRR* by 6% and 9% with the group size of 40 and 50, respectively. These results support TruGRC's ability to address conflicts and generate consensus recommendations with large groups.

The length of a recommendation list plays an important role in ranking-based recommendation tasks. To assess list length, we fix the group size at 20 and vary the recommendation length from 5 to 50 in steps of 5 and test each method. Figs. 4 and 5 report the comparisons on precision, recall, and F1 in both Ciao and Epinions. From these two figures, we observe that the accuracy of BF declines with the recommendation length grows, which means BF prioritizes the top items in the recommendation list. This means BF has a poor ability to generate long recommendation lists. In comparison, AF's performance overtakes the other baselines as the recommendation list becomes longer. Overall, the differences between each method's performance are reduced as the recommendation length increases. However, TruGRC still shows the better performance than AF, with a respective improvement in terms of *Pre*@50, *Rec*@50 and *F*1@50 by 7%, 6% and 7% in Ciao, and 2%, 3%

and 2% in Epinions. Based on these combined experimental results, TruGRC demonstrates superior accuracy with both different group sizes and different recommendation list lengths.

4.4. The impact of parameters

TruGRC contains three parameters, i.e., λ, λ_α and λ_β. To determine these values for each dataset, we verify them in terms of *MRR*, and tune each parameter in the range $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$ while fixing the other two parameters. The results are reported in Figs. 6 and 7. The parameter λ avoids over-fitting, which should be very small in general, e.g. 0.001 or 0.01 [26,54]. The two figures show that TruGRC produces the best performance when $\lambda = 0.01$ in Ciao and $\lambda = 0.001$ in Epinions, which is reasonable.

The parameter λ_α controls the importance of the item-related and virtual coordinator-related regularization terms. The results clearly indicate that a proper value, i.e., $\lambda_\alpha = 1$, can improve the recommendation performance in these two datasets and also demonstrate that the regularization terms in Eqs. (8) and (10) are very helpful to our model. The parameter λ_β regulates the trust-related regularization terms. The optimal values are 0.01 and 0.1 for Ciao and Epinions, respectively. When tuning this parameter, the performance changes significantly in Epinions compared to Ciao because the Epinions dataset contains more observed trust
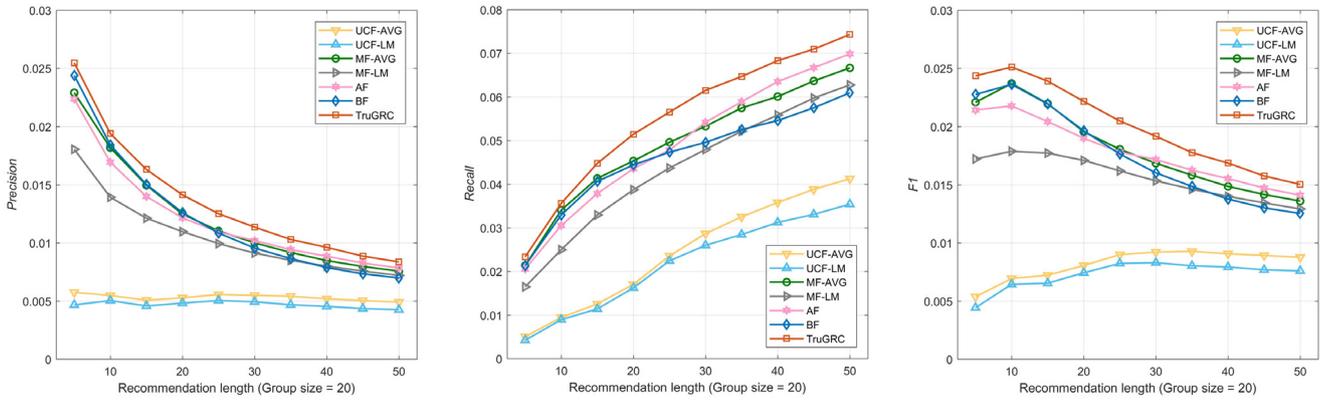
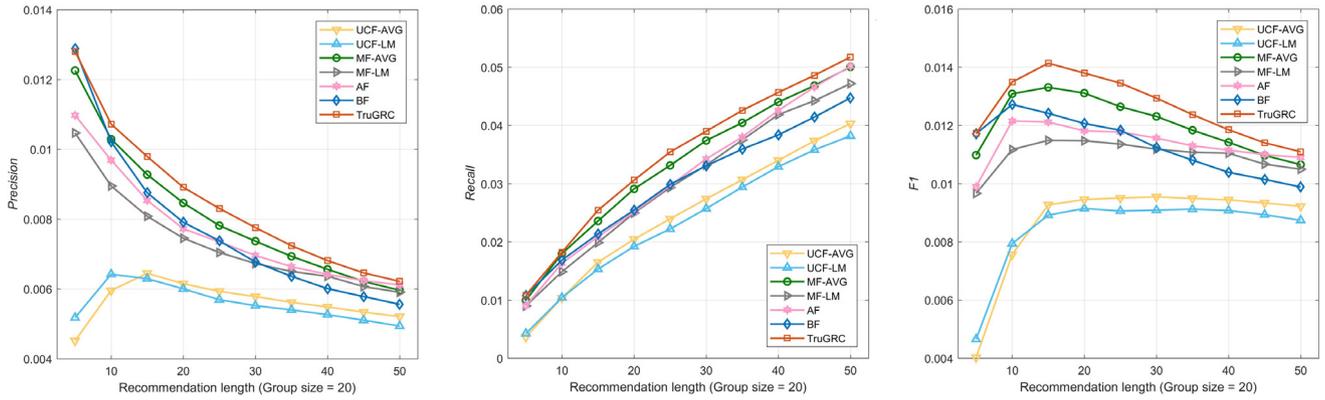**Fig. 4.** The comparisons on *Pre*@5, *Rec*@5 and *F*1@5 with different recommendation length in Ciao.

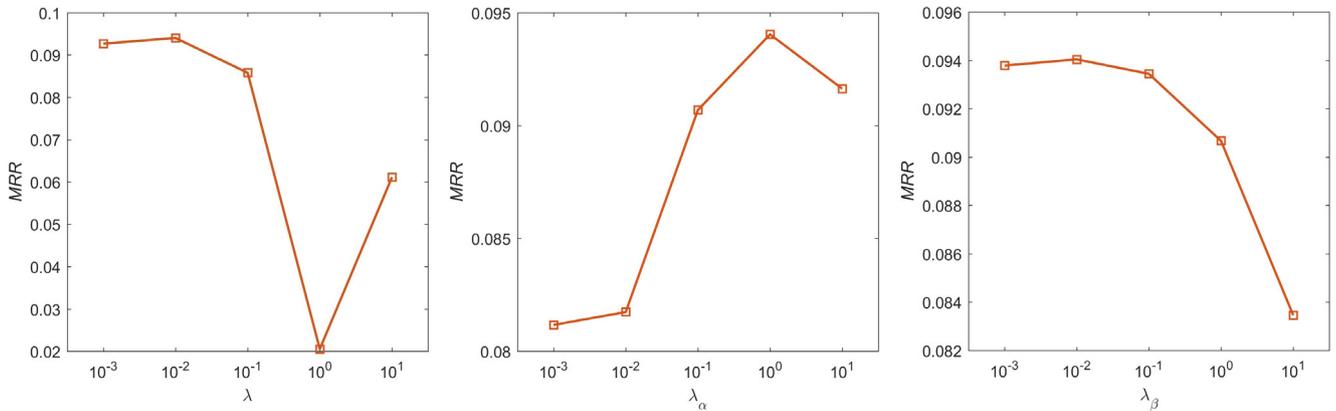**Fig. 5.** The comparisons on *Pre*@5, *Rec*@5 and *F*1@5 with different recommendation length in Epinions.

Fig. 6. The impact of parameters λ, λ_α and λ_β on *MRR* in Ciao.

links, which means personal influence can be calculated more accurately in this dataset.

5. System architecture and potential applications

Aiming to illustrate how to make group recommendations, we design an architecture of the proposed trust-aware group recommender system shown in Fig. 8. There are three components in our designed system, including a system interface, a data server and a group recommender engine.

In the system interface, users in each group can interact with the system through web-based interfaces, e.g., websites. Each user's behaviors on websites indicate user's personalized preferences, such as rating items and building friendship with other users. A data collector extracts user preference data and historical browsing information from web-based interfaces, and then a data cleanser is arranged to find out useful data and transform it into structural data, e.g., XML.

The data server is responsible for storing user preference data and divides data into three categories that include rating data, social data and group data. Note that, the data server may obtain user preference data from the system interface more than once in order to achieve comprehensive user preferences.

User preference data is regarded as input for the group recommender engine from the data server. A data filter is deployed to eliminate negative feedback in rating data, because we only consider positive feedback in the proposed TruGRC method. Rating data is transformed to rating vectors by an data converter, and
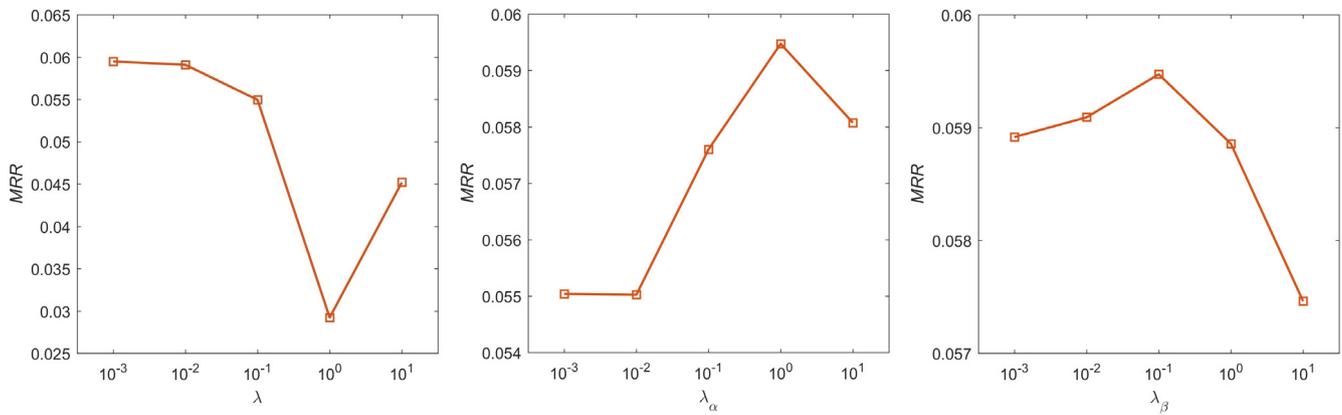
Fig. 7. The impact of parameters λ, λ_α and λ_β on *MRR* in Epinions.
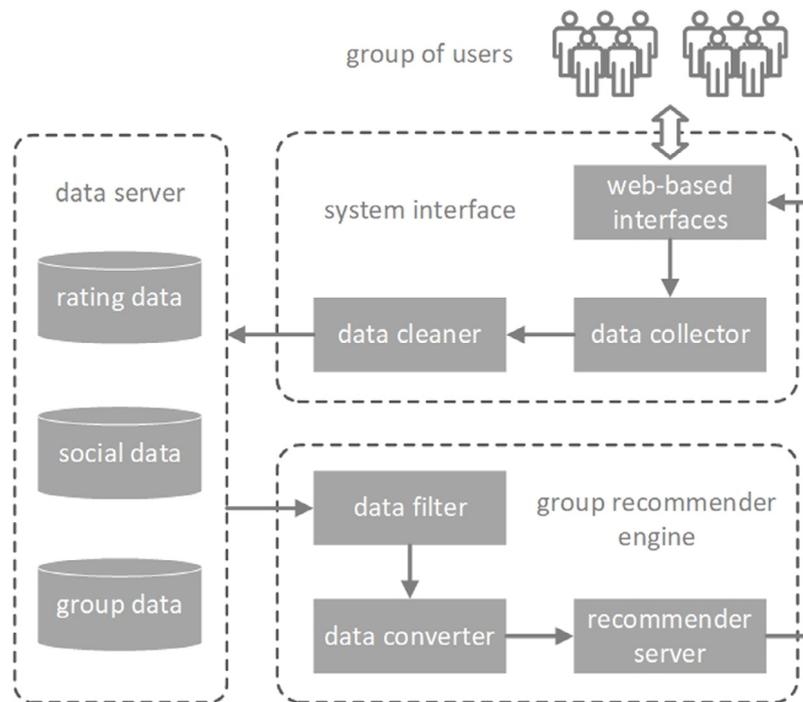

Fig. 8. The architecture of our designed trust-aware group recommender system. This system includes three components, i.e., a system interface, a data server and a group recommender engine.

then transfers into the recommender server. The proposed TruGRC method is applied in a recommender server which models overall group preferences and makes group recommendations with integrated rating, social and group data. Finally, group recommendation results are reported to web-based interfaces where users can view them.

The trust-aware group recommender system can be used to generate recommendations in several group activity scenarios . For example:

- Tourism recommendation is a typical application scenario of group recommender systems. All members in a touring party only can select a series of unique destinations.
- Restaurant recommendation. A group of users only can choose one restaurant for dining, which accords with group recommendation situations.
- Movie recommendation. If some friends want to see a movie together, these friends are regarded as a specific group and a group recommender system can be used to make recommendations for them.

6. Conclusions and future work

Group recommendation is a significant issue in many domains based on group activities. This paper proposes a trust-aware group recommendation method, called TruGRC, that integrates benefits from both the result and the profile aggregations to increase the satisfaction rate of group recommendation. A virtual coordinator provides a global view of the overall preferences for a group of users. This coordinator interacts with each group member to relieve conflicting personal preferences within the group. The explicit trust relations within social networks are leveraged to calculate the personal influence of group members, which is then used to model the impact between each group member and the virtual coordinator. With this information, the virtual coordinator can easily generate group preferences using the average aggregation method. The results from experiments on two benchmark datasets indicate TruGRC outperforms the baselines in terms of most metrics with a range of different group sizes. Specifically, compared to MF-AVG and BF, TruGRC enhances *MRR* by 10% and 5% in Ciao and 2% and 8% in Epinions when the group size is 10.

In future work, we will further investigate model-based group recommender systems by introducing other effective machine learning models, e.g., transfer learning [55,56] and deep learning [57,58]. Additionally, we will look to integrating other different kinds of auxiliary content with a view to developing practical applications.
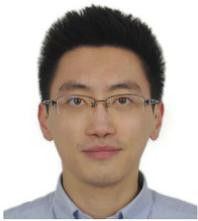
Acknowledgments

References

[1] J. Lu, D. Wu, M. Mao, W. Wang, G. Zhang, Recommender system application developments: a survey, Decis. Support Syst. 74 (2015) 12–32.

[2] L. Ren, W. Wang, An svm-based collaborative filtering approach for top-n web services recommendation, Future Gener. Comput. Syst. 78 (2018) 531–543.

[3] K. Ji, Z. Chen, R. Sun, K. Ma, Z. Yuan, G. Xu, Gist: A generative model with individual and subgroup-based topics for group recommendation, Expert Syst. Appl. 94 (2018) 81–93.

[4] J. Guo, Y. Zhu, A. Li, Q. Wang, W. Han, A social influence approach for group user modeling in group recommendation systems, IEEE Intell. Syst. 31 (5) (2016) 40–48.

[5] J. Castro, R. Yera, L. Martínez, An empirical study of natural noise management in group recommendation systems, Decis. Support Syst. 94 (2017) 1–11.

[6] F. Ortega, A. Hernando, J. Bobadilla, J.H. Kang, Recommending items to group of users using matrix factorization based collaborative filtering, Inform. Sci. 345 (2016) 313–324.

[7] W. Wang, G. Zhang, J. Lu, Member contribution-based group recommender system, Decis. Support Syst. 87 (2016) 80–93.

[8] J. Castro, J. Lu, G. Zhang, Y. Dong, L. Martínez, Opinion dynamics-based group recommender systems, IEEE Trans. Syst. Mand Cybern.: Syst. PP (99) (2017) 1–13, http://dx.doi.org/10.1109/TSMC.2017.2695158.

[9] J. Wang, Y. Jiang, J. Sun, Y. Liu, X. Liu, Group recommendation based on a bidirectional tensor factorization model, World Wide Web 21 (4) (2018) 961–984.

[10] V.R. Kagita, A.K. Pujari, V. Padmanabhan, Virtual user approach for group recommender systems using precedence relations, Inform. Sci. 294 (2015) 15–30.

[11] L. Baltrunas, T. Makcinskas, F. Ricci, Group recommendations with rank aggregation and collaborative filtering, in: Proceedings of the 4th ACM Conference on Recommender Systems, 2010, pp. 119–126.

[12] J. Castro, F.J. Quesada, I. Palomares, L. Martínez, A consensus-driven group recommender system, Int. J. Intell. Syst. 30 (8) (2015) 887–906.

[13] Y.-D. Seo, Y.-G. Kim, E. Lee, K.-S. Seol, D.-K. Baik, An enhanced aggregation method considering deviations for a group recommendation, Expert Syst. Appl. 93 (2018) 299–312.

[14] H. Ma, D. Zhou, C. Liu, M.R. Lyu, I. King, Recommender systems with social regularization, in: Proceedings of the 4th ACM International Conference on Web Search and Data Mining, 2011, pp. 287–296.

[15] L. Quijano-Sanchez, J.A. Recio-Garcia, B. Diaz-Agudo, G. Jimenez-Diaz, Social factors in group recommender systems, ACM Trans. Intell. Syst. Technol. 4 (1) (2013) 8.

[16] K.W. Thomas, Thomas-kilmann conflict mode, TKI Profile and Interpretive Report (2008) 1–11..

[17] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, in: Proceedings of the 10th International Conference on World Wide Web, 2001, pp. 285–295.

[18] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 42–49.

[19] X. Cai, J. Han, W. Li, R. Zhang, S. Pan, L. Yang, A three-layered mutually reinforced model for personalized citation recommendation, IEEE Trans. Neural Netw. Learn. Syst. PP (99) (2018) 1–12, http://dx.doi.org/10.1109/TNNLS.2018.2817245.

[20] B.K. Patra, R. Launonen, V. Ollikainen, S. Nandi, A new similarity measure using bhattacharyya coefficient for collaborative filtering in sparse data, Knowl.-Based Syst. 82 (2015) 163–177.

[21] H. Qiu, Y. Liu, G. Guo, Z. Sun, J. Zhang, H.T. Nguyen, Bprh: Bayesian personalized ranking for heterogeneous implicit feedback, Inform. Sci. 453 (2018) 80–98.

[22] J. Wang, A.P. De Vries, M.J. Reinders, Unifying user-based and item-based collaborative filtering approaches by similarity fusion, in: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2006, pp. 501–508.

[23] H. Ma, I. King, M.R. Lyu, Effective missing data prediction for collaborative filtering, in: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2007, pp. 39–46.

[24] Y. Koren, Factor in the neighbors: Scalable and accurate collaborative filtering, ACM Trans. Knowl. Discov. Data 4 (1) (2010) 1.

[25] J. Liu, Y. Jiang, Z. Li, X. Zhang, H. Lu, Domain-sensitive recommendation with user-item subgroup analysis, IEEE Trans. Knowl. Data Eng. 28 (4) (2016) 939–950.

[26] G. Guo, J. Zhang, N. Yorke-Smith, A novel recommendation model regularized with user trust and item ratings, IEEE Trans. Knowl. Data Eng. 28 (7) (2016) 1607–1620.

[27] Y. Zhang, M. Chen, D. Huang, D. Wu, Y. Li, idoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization, Future Gener. Comput. Syst. 66 (2017) 30–35.

[28] A. Anagnostopoulos, R. Atassi, L. Becchetti, A. Fazzone, F. Silvestri, Tour recommendation for groups, Data Min. Knowl. Discov. 31 (5) (2017) 1157–1188.

[29] Z. Yu, X. Zhou, Y. Hao, J. Gu, Tv program recommendation for multiple viewers based on user profile merging, User Model. User-Adapt. Interact. 16 (1) (2006) 63–82.

[30] P. Dwivedi, K.K. Bharadwaj, e-learning recommender system for a group of learners based on the unified learner profile approach, Expert Syst. 32 (2) (2015) 264–276.

[31] S. Amer-Yahia, S.B. Roy, A. Chawlat, G. Das, C. Yu, Group recommendation: Semantics and efficiency, Proc. VLDB Endowment 2 (1) (2009) 754–765.

[32] L. Boratto, S. Carta, G. Fenu, Discovery and representation of the preferences of automatically detected groups: Exploiting the link between group modeling and clustering, Future Gener. Comput. Syst. 64 (2016) 165–174.

[33] L. Quijano-Sanchez, C. Sauer, J.A. Recio-Garcia, B. Diaz-Agudo, Make it personal: a social explanation system applied to group recommendations, Expert Syst. Appl. 76 (2017) 36–48.

[34] J. Castro, R. Yera, L. Martinez, A fuzzy approach for natural noise management in group recommender systems, Expert Syst. Appl. 94 (2018) 237–249.

[35] R. Forsati, M. Mahdavi, M. Shamsfard, M. Sarwat, Matrix factorization with explicit trust and distrust side information for improved social recommendation, ACM Trans. Inf. Syst. 32 (4) (2014) 17.

[36] A. Kalaï, C.A. Zayani, I. Amous, W. Abdelghani, F. Sèdes, Social collaborative service recommendation approach based on user's trust and domain-specific expertise, Future Gener. Comput. Syst. 80 (2018) 355–367.

[37] X. Ma, J. Ma, H. Li, Q. Jiang, S. Gao, Armor: A trust-based privacy-preserving framework for decentralized friend recommendation in online social networks, Future Gener. Comput. Syst. 79 (2018) 82–94.

[38] G. Guo, H. Qiu, Z. Tan, Y. Liu, J. Ma, X. Wang, Resolving data sparsity by multi-type auxiliary implicit feedback for recommender systems, Knowl.-Based Syst. 138 (2017) 202–207.

[39] N. Mirbakhsh, C.X. Ling, Improving top-n recommendation for cold-start users via cross-domain information, ACM Trans. Knowl. Discov. Data 9 (4) (2015) 33.

[40] H. Ma, An experimental study on implicit social recommendation, in: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2013, pp. 73–82.

[41] H. Liu, F. Xia, Z. Chen, N.Y. Asabere, J. Ma, R. Huang, Trucom: Exploiting domain-specific trust networks for multicategory item recommendation, IEEE Syst. J. 11 (1) (2017) 295–304.

[42] M. Salamó, K. McCarthy, B. Smyth, Generating recommendations for consensus negotiation in group personalization services, Pers. Ubiquitous Comput. 16 (5) (2012) 597–610.

[43] X. Wang, Y. Liu, G. Zhang, F. Xiong, J. Lu, Diffusion-based recommendation with trust relations on tripartite graphs, J. Stat. Mech. Theory Exp. 2017 (8) (2017) 083405.

[44] L. Lü, D. Chen, X.-L. Ren, Q.-M. Zhang, Y.-C. Zhang, T. Zhou, Vital nodes identification in complex networks, Phys. Rep. 650 (2016) 1–63.

[45] F. Xiong, X. Wang, S. Pan, H. Yang, H. Wang, C. Zhang, Social recommendation with evolutionary opinion dynamics, IEEE Trans. Syst. Man. Cybern.: Syst. PP (99) (2018) 1–13, http://dx.doi.org/10.1109/TSMC.2018.2854000.

[46] https://www.cse.msu.edu/~tangjili/trust.html.

[47] http://www.ciao.co.uk.

[48] http://www.epinions.com.

[49] J. Tang, S. Wang, X. Hu, D. Yin, Y. Bi, Y. Chang, H. Liu, Recommendation with social dimensions, in: Proceedings of the 30th AAAI Conference on Artificial Intelligence, 2016, pp. 251–257.

[50] X. Wang, Y. Liu, F. Xiong, Improved personalized recommendation based on a similarity network, Physica A 456 (2016) 271–280.

[51] O. Kaššák, M. Kompan, M. Bieliková, Personalized hybrid recommendation for group of users: Top-n multimedia recommender, Inf. Process. Manage. 52 (3) (2016) 459–477.

[52] X. Wang, Y. Liu, G. Zhang, Y. Zhang, H. Chen, J. Lu, Mixed similarity diffusion for recommendation on bipartite networks, IEEE Access 5 (2017) 21029–21038.

[53] J.L. Herlocker, J.A. Konstan, A. Borchers, J. Riedl, An algorithmic framework for performing collaborative filtering, in: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1999, pp. 230–237.

[54] W. Pan, Z. Ming, Collaborative recommendation with multiclass preference context, IEEE Intell. Syst. 32 (2) (2017) 45–51.

[55] W. Pan, Q. Yang, Y. Duan, B. Tan, Z. Ming, Transfer learning for behavior ranking, ACM Trans. Intell. Syst. Technol. 8 (5) (2017) 65.

[56] F. Liu, J. Lu, G. Zhang, Unsupervised heterogeneous domain adaptation via shared fuzzy equivalence relations, IEEE Trans. Fuzzy Syst. (99) (2018) 1–14, http://dx.doi.org/10.1109/TFUZZ.2018.2836364.

[57] S. Deng, L. Huang, G. Xu, X. Wu, Z. Wu, On deep learning for trust-aware recommendations in social networks, IEEE Trans. Neural Netw. Learn. Syst. 28 (5) (2017) 1164–1177.

[58] L. Zhang, T. Luo, F. Zhang, Y. Wu, A recommendation model based on deep neural network, IEEE Access 6 (2018) 9454–9463.

Jie Lu received the Ph.D. degree in information systems from the Curtin University of Technology, Perth, Australia, in 2000. She is currently a Distinguished Professor and the Associate Dean in Research Excellence of the Faculty of Engineering and Information Technology, and the Director of the Centre for Artificial Intelligence, at the University of Technology Sydney. She has published 10 research books and over 400 papers. Her main research interests include decision making modeling, decision support system tools, uncertain information processing, recommender systems, and e-Government and e-Service intelligence. Prof. Lu was a recipient of 8 Australian Research Council Discovery Grants. She serves as an Editor-in-Chief for Knowledge-Based Systems (Elsevier) and for International Journal of Computational Intelligence Systems (Atlantis), and an Associate Editor for IEEE Transactions on Fuzzy Systems. She is a Fellow of IEEE and a Fellow of IFSA.

Ximeng Wang received the B.E. degree in communication engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2011 and the M.E. degree in software engineering from Beijing Jiaotong University, Beijing, China, in 2013. He is currently pursuing the Ph.D. degree in Beijing Jiaotong University. From 2017, he has been a joint training Ph.D. student with the University of Technology Sydney. His current research interests include recommender systems, network embedding and data mining.

Yun Liu received the Ph.D. degree in communication and information systems from Beijing Jiaotong University, Beijing, China, in 2004. She is currently a Professor and the Director of the Key Laboratory of Communication & Information Systems, Beijing Municipal Commission of Education, at Beijing Jiaotong University. She has published 7 monographs and over 300 papers, and holds 9 patents and 5 software copyrights. Her current research interests include information networks, web mining and Internet security. Prof. Liu was a recipient of 3 National High Technology Research and Development Programs of China (863 Program), 3 National Natural Science Foundations of China and several other research grants. She serves as an Editor for Fundamenta Informaticae (IOS Press). She is a Fellow of IET.

Fei Xiong received the B.E. degree in communication engineering and the Ph.D. degree in communication and information systems from Beijing Jiaotong University, Beijing, China, in 2007 and 2013, respectively. He is currently an Associate Professor with the School of Electronic and Information Engineering, Beijing Jiaotong University. From 2011 to 2012, he was a visiting scholar at Carnegie Mellon University. He has published over 60 papers in refereed journals and conference proceedings. He was a recipient of National Natural Science Foundations of China and several other research grants. His current research interests include the areas of web mining, complex networks and complex systems.

Guangquan Zhang received his Ph.D. degree in applied mathematics from the Curtin University of Technology, Perth, Australia, in 2001. He is currently an Associate Professor with the Centre for Artificial Intelligence, Faculty of Engineering and Information Technology, University of Technology Sydney. He has published 4 monographs, 5 reference books, and over 350 papers in refereed journals and conference proceedings. Dr. Zhang was a recipient of 7 Australian Research Council Discovery Grants and several other research grants. His main research interests include multi-objective and group decision making, and decision support system tools.