Journal of Intelligent Information Systems
https://doi.org/10.1007/s10844-018-0542-3

A survey on group recommender systems

Sriharsha Dara[1] · C. Ravindranath Chowdary[1] · Chintoo Kumar[1]

Received: 28 June 2018 / Revised: 7 December 2018 / Accepted: 12 December 2018 /
Published online: 03 January 2019

Abstract

Recommender systems are increasingly used in various domains like movies, travel, music, etc. The rise in social activities has increased the usage of recommender systems in general and group recommender systems in particular. A group recommender system is a system that recommends items to a group of users collectively, given their preferences. In addition to the user preferences, using social and behavioural aspects of group members to generate group recommendations will increase the quality of the content recommended in heterogeneous groups. Group recommender systems also address the cold start problem that arises in an individual recommendation system. This paper presents a survey on the state-of-the-art in group recommender systems concerning various domains. We discussed existing systems with respect to their aggregation and user preference models. This organisation is very useful to understand the intricacies with respect to each domain.

Keywords Group recommender systems · Domain wise survey · Aggregation models

1 Introduction

Recommender systems solve the problem of information overload and help users to choose from the options in our day to day life. Recommender systems collect the past preferences of users and generate an appropriate recommendation of items to satisfy the users. The user preference information can be acquired either by explicit ratings or by implicit ratings. Many traditional recommender systems have concentrated only on single-user models. But in real life, there are many situations, where we interact mostly with groups like watching a

✉ C. Ravindranath Chowdary
rchowdary.cse@iitbhu.ac.in

Sriharsha Dara
dara.sriharsha.cse13@iitbhu.ac.in

Chintoo Kumar
chintookr.rs.cse17@itbhu.ac.in

[1] Department of Computer Science and Engineering, Indian Institute of Technology (BHU), Varanasi, 221005, India

movie with the family, having dinner with colleagues, planning a vacation with friends, etc. So, group recommendation is also an equally important problem to be addressed.

The increase in mobile devices and social networking increased the importance of group recommendations in various domains. Incorporating social and behavioural aspects in the group recommendations is an active field of research (Quijano-Sanchez et al. 2010). The factors that influence the group recommender systems are social background, social relations, trust, similarities in interests, etc. These factors can be either directly or indirectly extracted from the social network structure (Kompan and Bielikova 2014). Figure 1 gives a general framework of a Group Recommendation system (Christensen and Schiaffino 2011a).

The issues that arise in recommending items to groups are discussed in Jameson and Smyth (2007), and the issues are organised with respect to the four subtasks of the group recommendation problem. The subtasks are:- 1) Acquiring user preference information 2) Generating recommendations 3) Explaining recommendations and 4) Helping group members to settle on a final recommendation, and the challenges in each subtask are discussed in detail. Different types of preferences are discussed in Felfernig et al. (2018c) and various approaches related to the handling of preferences in group recommendations are analysed. The authors in Felfernig et al. (2018a) discuss the various aggregation strategies and different existing approaches in group recommendation. Evaluation of the group recommendation algorithm is necessary for identifying the usefulness of the algorithm. So, Felfernig et al. (2018b) discusses the various approaches to evaluate group recommendation models. Here, the authors also discuss various metrics used for evaluating group recommendation models and their relevance based on the domain of an item. However, in our review, we discuss existing literature concerning each domain. Within each domain, we compared existing systems with respect to the aggregation and user preference models. This comparison will give better opportunity to analyse the individual systems. This organisation is useful to understand the intricacies with respect to each domain.

Groups can be classified as homogeneous groups and heterogeneous groups based on their composition. A homogeneous group consists of members with similar interests, whereas heterogeneous groups consist of members with diverse interests. Many existing

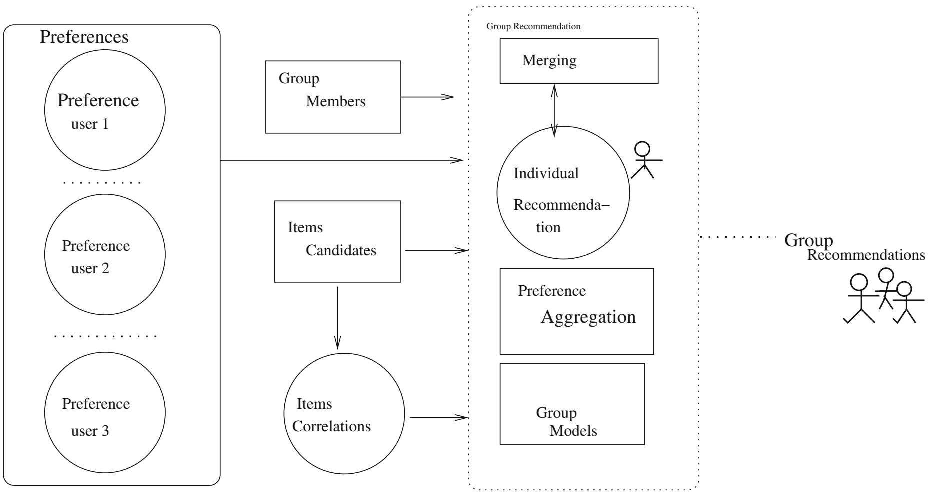

Fig. 1 General framework of Group Recommendations

group recommendation strategies (Kim et al. 2010; Gorla et al. 2013; Boratto et al. 2010) are suitable only for homogeneous groups and do not perform better on heterogeneous groups. The problem with a heterogeneous group is - as the members of a group have differing interests, building a consensus to satisfy all the group members would be a difficult task. Very few works like Recio-Garcia et al. (2009), Sotelo et al. (2009), and Quijano-Sanchez et al. (2010) are on heterogeneous groups.

Most of the existing approaches for group recommender systems can be broadly classified into :

1. Creating a group profile by aggregating the individual member profiles and recommending the items based on the group profile (Kim et al. 2010).
2. By generating the personalised recommendations to the members of the group and then aggregating them into a single group recommendation (Kim and El Saddik 2015).

Group recommender systems can also be classified based on the types of groups to which the system recommends. Groups can be mainly of three types based on the interactions among the members of the group (Boratto and Carta 2010).

1. **Established group:** A group of people who explicitly choose to be a part of the group based on the shared common interests (Kim et al. 2010).
2. **Occasional group:** A group of people who perform some activity occasionally together. All the group members will have a common aim at a particular moment like planning for a vacation (Garcia et al. 2011).
3. **Random group:** A group of people who share an environment at a particular time without any link between them like passengers travelling in public transport (Crossen et al. 2002).

In this paper, we present a state-of-the-art survey of existing group recommender systems in various domains like movies, music, travel, food, etc. Table 1 gives the overview of different group recommender systems designed on multiple domains.

The rest of the paper is organised as follows. Section 2 provides the details of different aggregation models used in group recommendations. Different user data acquisition for group recommendations are discussed in Section 3. Section 4 explains various group recommender systems implemented across the domains. Section 5 discusses the impact of social and behavioural aspects in group recommendations. We gave few future directions and concluded in Section 6.

2 Group aggregation models

The main task in a group recommender system is to design a model that integrates all the preferences of the group members. This integration is done with the help of different aggregation strategies chosen according to the requirements of the group. As mentioned in Section 1, both the approaches of aggregation for group recommender systems, i.e. profile aggregation and recommendation aggregation incorporate aggregation in them. The most commonly used aggregation strategies are: (i) Aggregated Voting (ii) Least Misery.

Table 1 Overview of group recommender systems

Year	Model	Algorithm	Domain
2001	Polylens (O'Connor et al. 2001)	Collaborative filtering	Movies
2009	CMW (Recio-Garcia et al. 2009)	Collaborative filtering	Movies
2010	Rule based Rec (Gartrell et al. 2010)	Rule based, heuristics	Movies
2010	RankAgg (Baltrunas et al. 2010)	Rank aggregation, collaborative filtering	Movies
2010	IGRA (Boratto et al. 2010)	K-means clustering, collaborative filtering	Movies
2010	IterVote (Dery et al. 2010)	Dynamic information gain, expected rating	Movies
2011	e/r RS (Bobadilla et al. 2012)	Collaborative filtering	Movies
2011	HappyMovie (Quijano-Sánchez et al. 2011)	Conflict mode weight	Movies
2012	GroupRem (Pera and Ng 2013)	Aggregation	Movies
2013	InfoMatch (Gorla et al. 2013)	Probabilistic, information matching	Movies
2013	UGSM (Ortega et al. 2013)	Collaborative filtering, aggregation	Movies
2014	COM (Yuan et al. 2014)	Probabilistic	Movies
2014	FlexiFeed model (Basu Roy et al. 2014)	Aggregation	Movies
2014	HybridRec (Quijano-Sánchez et al. 2014)	Hybrid	Movies
2015	LGM (Shi et al. 2015)	Matrix factorization	Movies
2015	CRP (Castro et al. 2015)	Collaborative filtering	Movies
2015	Top-N Rec (Kaššák et al. 2016)	Content-based, collaborative filtering	Movies
2015	POSN (Salehi-Abari and networks 2015)	Aggregation	Movies
2016	NNM (Castro et al. 2017)	Aggregation	Movies
2017	PSIE (Quijano-Sanchez et al. 2017)	Collaborative filtering	Movies
2017	OrderedRec (Agarwal et al. 2017)	Aggregation	Movies
1998	MusicFX (McCarthy and Anagnost 1998)	Arbitration algorithm	Music
2002	Flytrap (Crossen et al. 2002)	Voting	Music
2011	jMusicGroupRecommender (Christensen and Schiaffino 2011b)	Aggregation	Music, Movies

Table 1 (continued)

Year	Model	Algorithm	Domain
2014	SparseRec (Ghazarian and Ali Nematbakhsh 2015)	SVM, Collaborative filtering	Music
2015	MusicRec (Kim and El Saddik 2015)	Aggregation, random walk with restarts	Music
2006	CATS (McCarthy et al. 2006)	Collaborative filtering	Travel
2010	e-Tourism (Garcia et al. 2011)	Aggregation	Travel
2012	ConNeg (Salamó et al. 2012)	Aggregation	Travel
2013	TravelRec (Chen et al. 2013)	Probabilistic bayesian learning	Travel
2017	MobileRec (Nguyen and Ricci 2017)	Dynamic voting	Travel
2009	TV-A (Sotelo et al. 2009)	Aggregation	TV programs
2011	PBRec (Seko et al. 2011)	Power balance map	TV programs
2011	VideoRec (Seko et al. 2011)	Power balance map	TV programs
2012	External expert (Wang et al. 2012)	K-means clustering, context filtering	Video
2002	Pocket restaurant finder (Mccarthy 2002a)	Arbitration algorithm	Restaurants
2008	RestauRec (Park et al. 2008)	Bayesian network, probabilistic approach	Restaurants
2011	IMPC (Guzzi et al. 2011)	Critiquing based	Restaurants
2012	PIT,E-PIT (Liu et al. 2012)	Gibbs sampling	Social networks
2012	SIS (Ye et al. 2012)	Probabilistic	Social networks
2007	(Chen et al. 2008)	Genetic algorithm , collaborative filtering	Synthetic data
2010	P&TRec (Quijano-Sanchez et al. 2010)	Collaborative filtering	Synthetic data, social network
2009	Research assist (Baskin and Krishnamurthi 2009)	Kemeny ordering, aggregation	Research papers
2010	Grec-OC (Kim et al. 2010)	Hybrid approach	Books
2017	BookRec (Ahmad et al. 2017)	Group neighborhood, collaborative filtering	Books
2010	RecipeRec (Berkovsky and Freyne 2010)	Collaborative filtering	Food
2016	CrowdRec (Rakesh et al. 2016)	Probabilistic	Crowd funding

Table 2 Group satisfaction with respect to items

Items	1	2	3	4	5	6
Satisfaction	4	1	1	2	1	1

2.1 Aggregated voting

In Aggregated Voting, the task is to determine the set of items that increase the overall individual satisfaction scores of all the users in the group. The objective is to maximise the overall satisfaction of the group.

Example 1 Suppose there are *n(=5)* users in a group and *m(=6)* items and with the group budget of *k(=2)*, and with user preferences:

user 1's list is {1, 4}
user 2's list is {3, 1}
user 3's list is {2, 5}
user 4's list is {1, 4}
user 5's list is {1, 6}

In this case, considering all possible combinations of size k(=2), we have ${}^{m}C_k$=15 possible recommendation sets. This strategy considers the preferences of group members with an objective to maximize the overall group satisfaction. Let us consider; if an item from a group member's preference appears in the final recommendation, it increases the group satisfaction by 1. Table 2 gives the group satisfaction for the items. Item 1 has satisfaction 4 as users 1, 2, 4 and 5 have item 1 in their preference lists. Similarly, item 2 has satisfaction 1 as item 2 is in the preference list of user 3 alone.

Aggregated Voting picks the top k(=2) items having the highest satisfaction values. In this example, item set {1, 4} maximises the group satisfaction, and hence it is the final recommendation of Aggregated Voting.

2.2 Least misery

In the Least Misery method, the task is to recommend items such that they maximise the minimum individual satisfaction score among the users in a group. As the objective is to maximise the minimum satisfaction of users, it performs poorly when the overall group satisfaction is considered. As the initial satisfaction of every user is 0 for an empty recommendation, the Least Misery method selects the item providing the highest group satisfaction on the first iteration. It updates the individual user satisfaction in every iteration and picks the least satisfied user. In the next $k - 1$ iterations, the algorithm looks over the preferences of the least satisfied user which are still not added in the recommendation set and adds the item that maximises the group satisfaction. For the data in Example 1, in the first iteration, it selects item 1 and adds it to the recommendation set as it increases the group satisfaction the most (from Table 2). After the first iteration, the individual satisfaction of users is shown in Table 3.

In the second iteration, as the 3rd user has the least satisfaction, considering the preferences of the 3rd user, the algorithm adds the item which increases the group satisfaction the most (from Table 2). In this case, as both item 2 and item 5 adds the same satisfaction value to the group, any item can be included in the final recommendation. So, the least misery

Table 3 Individual user satisfaction

User	1	2	3	4	5
Satisfaction	1	1	0	1	1

algorithm recommends {1, 5} (or {1, 2}) ensuring that every user is minimally dissatisfied. In the Least Misery method, the task is to recommend items such that they maximise the minimum individual satisfaction score among the users in the group.

2.3 Brief description on rank aggregation strategies

Let us take three users (u_1, u_2 and u_3) and let users provide ratings for the items ($i_1, i_2, i_3 .....i_8$). Many rank aggregation strategies play a pivotal role in group recommendations. Least misery is given in Table 4. For an item, the least rating is chosen as its group rating. The ranked list of items is chosen in the descending order of their group ratings.

Additive utilitarian is a strategy where individual ratings are summed for each item. It is also known as an average strategy because resulting group ranking gives the same result as taking the average of individual ratings. In Multiplicative utilitarian, individual ratings are multiplied for each item. Multiplicative aggregation model generates the group recommendations by calculating the geometric mean of user preferences (Christensen and Schiaffino 2011b). Tables 5 and 6 illustrate Additive utilitarian and Multiplicative utilitarian, respectively. In Most Pleasure strategy, group rating is determined by the highest rating as opposed to Least Misery.

Plurality Voting is a group recommendation technique which allows each user to vote for only one item and the item with the highest number of votes is recommended to the group (Salehi-Abari and networks 2015). Copeland method (Colomer 2013) generates the recommendations by taking the pairwise preferences of users. Fairness strategy uses negotiation of user preferences and generates group recommendations (Kaššák et al. 2016). Approval voting assigns a point to all the items a user likes. Suppose that each user votes for all the items with a rating above a certain threshold (let's assume 5). Group rating for an item in Approval voting is the sum of individual votes. Table 7 illustrates ranked list of items for the group by Approval voting strategy based on the ratings of each user above a certain threshold. Table 7 is based on the ratings given in Table 4.

2.4 Systems developed using aggregation strategies

- Pocket Restaurant Finder (Mccarthy 2002b) recommends a restaurant to a group of people by averaging the individual preferences of the group members on different types of features (location, cost, etc.).

Table 4 The ranked list of the items for the group using Least Misery strategy would be (i_5-i_6, i_3-i_4, i_1, i_8, i_7, i_2)

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	8	10	7	10	9	8	10	6
u_2	7	10	6	9	8	10	9	4
u_3	5	1	8	6	9	10	3	5
Group rating	5	1	6	6	8	8	3	4

Table 5 Additive utilitarian: The ranked list of the items for the group would be (i_6, i_5, i_4, i_7, i_2-i_3, i_1, i_8)

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	8	10	7	10	9	8	10	6
u_2	7	10	6	9	8	10	9	4
u_3	5	1	8	6	9	10	3	5
Group rating	20	21	21	25	26	28	22	15

- (Masthoff 2011) showed, Multiplicative Utilitarian works best when selecting a sequence of television items to suit a group of viewers.
- Polylens (O'Connor et al. 2001) uses Least Misery aggregation strategy as it assumes that people tend to watch movies in small groups and the small group tends to be happy when its least satisfied member is happy. They propose an algorithm to merge the users' recommendation lists and to sort the merged list; they use Least Misery principle.
- LGM (Shi et al. 2015) uses average aggregation method that takes the average of all the preferences of users and produces top-k recommendations. Here, the average aggregation strategy is used to compare the results obtained from users' profile aggregation and users' recommendation aggregation by varying the group size n.
- MusicFX (McCarthy and Anagnost 1998) uses a variant of average without misery algorithm. After preprocessing of user preferences, average without misery strategy is applied to those preferences. Average without misery is a combination of average aggregation strategy and least misery strategy.
- The role of ordering in group recommender systems is studied in Agarwal et al. (2017). Greedy aggregation, least misery and least misery with priority strategies are discussed in this work. During a tie, among the least satisfied members, the least misery with priority picks the item preferred by the members with the highest priority. Also, Hungarian aggregated method is proposed, which generates recommendations in order.
- CATS (McCarthy et al. 2006) is a hybrid aggregation strategy that uses aggregated voting to generate their initial recommendations and then uses least misery strategy to reduce the misery of the group.
- Plurality voting strategy was implemented and tested by TV domain.
- Let's Browse (Lieberman et al. 1999) evaluates if the page considered by the system matches with the user profile above a page threshold using Approval voting and recommends the page with the highest score.
- (Masthoff 2011) showed Borda count is a strategy that generates more satisfaction when selecting a sequence of television items to suit a group of viewers. TravelWithFriends (De Pessemier et al. 2015) uses it to rank the top-5 travel destinations to recommend to a group.

Table 6 Multiplicative Utilitarian: The ranked list of the items for the group would be($i_6, i_5, i_4, i_3, i_1, i_7, i_8, i_2$)

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	8	10	7	10	9	8	10	6
u_2	7	10	6	9	8	10	9	4
u_3	5	1	8	6	9	10	3	5
Group rating	280	100	336	540	648	800	270	120

Table 7 Group choice following the Approval voting strategy. The ranked list of the items for the group would be(i_3-i_4-i_5-i_6, i_1-i_7, i_8)

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	1	1	1	1	1	1	1	1
u_2	1	1	1	1	1	1	1	
u_3			1	1	1	1		
Group rating	2	2	3	3	3	3	2	1

3 Acquiring user preferences for group recommender systems

The quality of recommendations is highly dependent on the quality of data used as preferences. The data acquisition in group recommendations can be broadly classified into three types:

3.1 Explicit data acquisition

In this type of data acquisition, users are given a questionnaire, and their preferences for the items are collected on a rating scale. These ratings are based on direct preference collection. The preferences may be collected either on individual items (Castro et al. 2017) or by collection on a group of items like preferences of genres (Shi et al. 2015), artists, etc. In this type of data acquisition, the major issue is data sparsity. As all the members of the group cannot specify ratings for all the available items, data sparsity problem should be addressed in the construction of a recommendation model using explicit preferences. Kemeny-ordering of items (Baskin and Krishnamurthi 2009), Expected Rating algorithm (Dery et al. 2010) are used to address this issue. Probabilistic approaches are used in Ye et al. (2012) and Gorla et al. (2013) to fill the missing values in the user-item rating matrix. Nearest neighbour approach along with genetic algorithm (Chen et al. 2008) can also be used to predict the missing ratings.

3.2 Implicit data acquisition

In this type of data acquisition model, a user need not specify his preferences directly. The data is collected indirectly by monitoring the user's behaviour like time spent on a website, her activity history, social relations among the group members, gender, age, etc. Social relationship information (Quijano-Sanchez et al. 2017; Kim and El Saddik 2015; Quijano-Sánchez et al. 2011) is used in the recommendation generation, as these models are implemented on social networking platforms. SIS model (Ye et al. 2012) uses music access history as well as location history to generate music recommendations to groups. Flytrap (Crossen et al. 2002) uses users' playlist, activity history. Social tagging information can also be used to generate recommendations (Kim and El Saddik 2015). Many social factors can be explored using the implicit data about the members of the group. LGM (Shi et al. 2015) extracts implicit factors from matrix factorisation which in turn improved the quality of recommendations.

3.3 Negative feedback data acquisition

Negative preferences are equally important along with the positive ones (Kompan and Bielikova 2014). Negative preferences are more important when the group recommendation is designed mainly to avoid some items that are disliked by the group members. Users give their preferences of items that they don't like and wish they do not appear in the final recommendation. Negative preference systems enhance the influence of users with minority opinion (Chao et al. 2005). The negative user preferences may be directly on data items or maybe on a class of items like movie genres, music length, type of food, etc. Very few works (Basu Roy et al. 2014; Quijano-Sánchez et al. 2014; Salamó et al. 2012) have considered negative feedback by the group members. The negative feedback can be effective in situations when a user dislikes only some of the items in the item set, and thus this information can be used to increase the satisfaction of a group.

4 Domain wise notes on group recommender systems

The following subsections describe the state-of-the-art systems available in group recommendations with respect to domains

4.1 Group recommender systems for movies

HappyMovie (Quijano-Sánchez et al. 2011) is a system that recommends movies to a group of users on Facebook. This system exploits the social and behavioural information about the users to improve the recommendations. In general, too many questionnaires have to be filled to get the social and behavioural information about the user. But, as the system is implemented on a social networking platform, the data is deduced easily. While social and trust profiles of a user are calculated automatically from the user's Facebook profile, personality profile is created through a questionnaire, and also the user needs to rate some movies. Later, collaborative filtering is applied to recommend items based on her profile.

Personalised explanations (Quijano-Sanchez et al. 2017) to the group recommendations are generated so that the users could build confidence and trust on the recommendations. These explanations are generated based on the relationships of the user with the other group members. The personalised explanations increase the users' likelihood to accept the recommendations and thus increase the efficiency of the system. HappyMovie (Quijano-Sánchez et al. 2011) system uses social factors and generate explanations to increase both individual and group satisfaction. This system has an advantage of using the social relationships and other information of the user directly as it is implemented on social networking platform itself.

As traditional group recommender systems generate recommendations to pre-defined groups, LGM (Shi et al. 2015) and (Boratto et al. 2010) worked on group recommendation strategies by detecting the groups in a fully automated fashion. These two approaches also address the data sparsity problem in user-item ratings. Both these models use the k-means clustering algorithm to detect groups.

Latent Group Model (LGM) (Shi et al. 2015) is a novel group recommendation model in which, the users who share the same interests of latent factors like a genre in movies are formed as a group. For every group, the latent factor profiles of all the users in a group are aggregated into a group profile, and matrix factorisation is used for the recommendation to these groups. Group profiles are generated from user latent factor matrix by aggregation.

Matrix factorisation is employed to detect groups of users who share similar latent factor preferences. The recommendation to a group is calculated from the group's profile and the component vector containing latent factors based on the user preferences.

The approach used in Boratto et al. (2010) generates the individual recommendations first and then uses the k-means clustering algorithm to identify groups among the users. The proposed baseline group prediction algorithm applies the k-means clustering algorithm on the user-item preference matrix (M). As a result, user groups having members who share similar preferences are formed. An improved group prediction algorithm is also proposed, which additionally predicts the unknown user preference ratings in the matrix M and applies the k-means clustering algorithm on the finally obtained entirely filled matrix M. Experimental results show that the improved group prediction algorithm performs better than the baseline algorithm. After predicting the groups among the users, a Collaborative Filtering based user-item approach is used for generating the group recommendations.

Probabilistic approaches to group recommendations are proposed in Gorla et al. (2013) and Yuan et al. (2014) and these models estimate the user's probability to like a movie. Group relevance (Gorla et al. 2013) is defined as the combination of item's relevance for each user both as a member of the group and as an individual. As it leads to better modelling of group members, higher quality group recommendations are produced. In the data, the items that are not rated by users are estimated using the probabilistic model. In the final recommendation, only the user-item ratings that are above a threshold are considered. Two matrices E (user-item feature matrix) and F (item-user feature matrix) are calculated using the data. Least misery strategy is used to calculate the relevance vector r using the matrices E and F. Item recommendations to individual users are calculated first, and later they are merged using probabilistic relevance framework with group-specific features.

COM (COnsensus Model) (Yuan et al. 2014) addresses group recommendation with a novel probabilistic model to model group activities. The authors' view on the recommendation is: the recommendation decisions are influenced by the people who have expertise in the topics related to the group, and different users have different influences in decision making in a group. So, COM incorporates the selection history and the personal ratings of the users. While aggregating the preferences of users over the items, weights are assigned to different group members. This model also considers the change in user's behaviour as an individual and as a member of a group. For the probabilistic parameter estimation, two-step Gibb's sampling method is used.

Both the models proposed in Baltrunas et al. (2010) and Kaššák et al. (2016) use hybrid approaches to generate group recommendations and both these systems use collaborative filtering. While Rank aggregation is used in Baltrunas et al. (2010) to filter the recommendations after collaborative filtering. Top-N Rec (Kaššák et al. 2016) generates recommendations in parallel using content-based approach and collaborative filtering. The results are merged to produce a useful recommendation.

The groups in Baltrunas et al. (2010) are formed randomly as well as using similarity criterion. Individual user rating prediction for an item is made by collaborative filtering based matrix factorisation approach, and then items are ranked based on rank aggregation method and the top-ranked items are recommended to the user group. Similarity among the users is calculated using Pearson's Correlation, while the approach in Kim and El Saddik (2015) uses cosine similarity to find the similarity among the members of a group. The authors also disprove the intuition that the effectiveness of group recommendations decrease as the group size increases. Their results show that recommendations for the groups having a group size of more than eight are significantly effective. It also reveals the fact that, producing group recommendations in the order of item rankings also increase the satisfaction of the users from the recommendations.

Top-N Rec (Kaššák et al. 2016) works on generating recommendations of Top-N items to the groups which are important mostly in the fields of entertainment, travel, etc. This model generates recommendations to the users using content-based and collaborative filtering approaches individually to each user. Later, aggregation strategies are used separately to generate group recommendations. Finally, items recommended by collaborative filtering are reordered based on the content-based results. This model is based on an intuition that items recommended using both the methods could be considered more appropriate than that of items recommended by individual techniques. When the proposed model is evaluated on a standard dataset, it is found that it performs better than the content-based, collaborative filtering and many traditional group recommendation techniques.

The authors in Agarwal et al. (2017) introduce order in the group recommendations. In the preferences given by the users, they also give their preference items in the order they would like to view the recommendation list. The satisfaction of a user is proportional to the similarity of her preferences with the produced recommendation. A new parameter called user satisfaction with the order is introduced using which the satisfaction of each user is calculated from the recommendation. To reduce the computational complexity of Aggregated method, an approximated algorithm called Greedy Aggregated Method (GrAM) is proposed with a little compromise in accuracy. Also, the authors propose Hungarian Aggregated method. As an extension to the Least Misery approach, Least Misery with priority is proposed which increases the overall group satisfaction with a minor increase in the computation time. Experimental results show that introducing order in the group recommendation increased the group satisfaction. Among the consensus functions proposed, the Hungarian Aggregated model gave more group satisfaction over the other existing models.

Table 8 gives various aggregation models and types of user preferences used in various group recommender systems in the movie domain.

4.2 Group recommender systems for music

MusicFX (McCarthy and Anagnost 1998) is an expert music recommender system for shared environments like fitness centres or gyms. It contains the database of preferences of all the members of the gym. MusicFX provides two different interfaces for gym members and staff. Gym members are allowed to update their preferences and can provide feedback anonymously. Gym staff are allowed to monitor the system and change the parameters that alter the recommendation generation strategies. As the group that stays in the gym varies dynamically, there is a provision in the system that maintains the list of users present in the gym at any point of time with the help of authorisation system present at the entrance of the gym. So, the recommendations get changed dynamically based on the current group in the gym. According to the feedback given by the users, many users are highly satisfied with the recommendations produced by MusicFX. This expert recommender system has been extended to other domains like restaurants (Mccarthy 2002a).

Flytrap (Crossen et al. 2002) is an intelligent music recommendation system which plays songs that satisfy the users surrounding the system. When installed on a users computer, Flytrap keeps track of the music a user plays on his computer and uploads the respective data to the Flytrap's database. This system gets the meta-data of the music viz., genre, artist, etc., and stores it as the preference of the user. This method is unlike users explicitly giving their preferences in many traditional group recommender systems like McCarthy and Anagnost (1998). This system also avoids the repetition of the same genres and artists continuously.

Table 8 Group recommender systems for movies

System	Aggregation model used	User preferences
Polylens (O'Connor et al. 2001)	Least misery	Explicit, implicit
HappyMovie (Quijano-Sánchez et al. 2011)	Fairness	Implicit
LGM (Shi et al. 2015)	Aggregated voting	Explicit
IGRA (Boratto et al. 2010)	Least misery	Explicit
InfoMatch (Gorla et al. 2013)	Aggregated voting, least misery	Explicit
COM (Yuan et al. 2014)	Geometric mean	Explicit, implicit
FlexiFeed (Basu Roy et al. 2014)	Aggregated voting, least misery	Negative feedback, explicit
HybridRec (Quijano-Sánchez et al. 2014)	Least misery, most pleasure, aggregated voting	Implicit, Negative feedback
OrderedRec (Agarwal et al. 2017)	Least misery, aggregated voting	Explicit
POSN (Salehi-Abari and networks 2015)	Plurality voting, borda score	Implicit, explicit
NNM (Castro et al. 2017)	Least misery, aggregated voting	Explicit

When a user's Radio Frequency ID (RFID) card comes into the predefined proximity of the main system that is playing music, it automatically identifies the entry of the user and updates his presence. Thus, the recommendations are produced dynamically.

The authors in Christensen and Schiaffino (2011b) propose two expert systems, jMusicGroupRecommender, jMoviesGroupRecommender and both of these systems generates group recommendations by merging the recommendations of individual users, aggregating individual user rating, and finally generating the group preference. A merging technique, a technique to generate group preference, four aggregation techniques, and a hybrid technique that includes merging and aggregation are proposed in this work. Each technique is evaluated based on the individual satisfaction of all the group members. Each song contains some attributes viz., artist, album, language, genre, etc. These attributes are used to predict the undefined preferences of the members of a group. Among all the proposed algorithms in this paper, it is reported that the hybrid technique is found to have high individual satisfactions among the users of a group. In contrast to the automatic consensus selection (Gartrell et al. 2010), this model gives the users a choice to select the consensus at the beginning.

The system proposed in Kim and El Saddik (2015) is a group recommendation system that recommends music to groups in the social media system. Here, a novel graph-based stochastic approach is proposed which analyses link-structure in a probabilistic way. This model uses the tags on the social media system for the recommendation of items to the groups. This model supports two types of group recommendations - (i) aggregation of individual preferences and computing the ranking (ii) Calculating individual rankings first and then combining the obtained rankings. This method employs Random Walks with Restarts (RWR) which has been used widely in the fields of information retrieval, web search ranking, etc. Similarities between a pair of users or between a pair of items are computed using cosine similarity. The proposed model outperforms many baseline methods when evaluated using a standard dataset.

Traditional group recommendation techniques fail to produce better recommendations when the user-item matrix is sparse, and the preferences of the user are unknown. The data sparsity problem is handled (Ghazarian and Ali Nematbakhsh 2015; Baskin and Krishnamurthi 2009; Dery et al. 2010) in group recommender systems in different ways. The model in Ghazarian and Ali Nematbakhsh (2015) employs item-item similarity to fill the user-item rating matrix, while Kemeny-ordering and expected rating algorithms are used in Baskin and Krishnamurthi (2009) and Dery et al. (2010) respectively to rank the items.

The model in Ghazarian and Ali Nematbakhsh (2015) calculates the item-item similarity and the computed similarities are used to enhance the existing memory based recommendation techniques. *Similarity_degree* is a decimal attribute which lies between 0.0 and 1.0 which is an extra attribute added in this model to every item-item pair. Initially, the SVM learning technique employed in this model is compared with other regression models and is found that SVM outperforms other models. The proposed model outperforms the latent factor model, which is in general considered as the best method to use among memory-based models.

Table 9 gives us the information regarding various aggregation models and user data models used in group music recommender systems.

4.3 Group recommender systems for travel

CATS (Collaborative Advisory Travel System) (McCarthy et al. 2006) is a travel recommender system that recommends places to travel for a group of users. The system is implemented on an interface device called DiamondTouch. The users could be able

Table 9 Group recommender systems for music

System	Aggregation model used	User preferences
MusicFX (McCarthy and Anagnost 1998)	Aggregated voting	Implicit
Flytrap (Crossen et al. 2002)	Aggregated voting	Implicit
jMusicRecommender (Christensen and Schiaffino 2011b)	Multiplicative, aggregated voting, least misery, fairness	Negative feedback, implicit
SparseRec (Ghazarian and Ali Nematbakhsh 2015)	Least misery, aggregated voting	Implicit
MusicRec (Kim and El Saddik 2015)	Aggregated voting	Implicit

to give their preferences both as a member of a group as well as a global user. The interface designed allows users to form groups of maximum size *4*. The system is designed in a way that users can dynamically change their preferences and the recommendation items get changed accordingly. The Group User Model (GUM) is generated based on their preferences, and collaborative filtering is applied synchronously whenever there is an update in the user preferences in a group. The extension to the designed model (Salamó et al. 2012) adds the feature of consensus negotiation to the system.

The authors in Garcia et al. (2011) work on a model that recommends tourist places to the users both for individual and for groups. Individual recommendations are extended to groups by applying various intersection and aggregation strategies. In addition to the preferences of users, this expert recommendation system also uses demographic data of the users and their past trip history to generate better recommendations. In this model, the recommendations are produced according to Generalistic Recommender System Kernel (GRSK), a domain-independent recommender system. The two main tasks of GRSK are the analysis of user profiles to generate a user preference model and to generate recommendations based on the user preference models. User profiles are generated using a hybrid ranking technique. As GRSK has been designed in a very generalised manner, the same framework can be extended to other domains. Experimental results show that the system outperforms traditional group recommender systems.

In group recommender systems, solving user preference conflicts is a major challenge to be handled. So the system proposed in Salamó et al. (2012) mainly concentrates on building a consensus aggregating the preferences of users in group recommendations as an additional feature to the model proposed in CATS (McCarthy et al. 2006). As an extension to CATS recommender system, consensus negotiation is added to the model to produce better recommendations. Nine consensus negotiation strategies based on statistical, content and collaborative ideas are evaluated and analysed using the live-user preference data of ski-holiday group recommender system. These strategies are analysed by varying group size and on different types of user groups. The benefits of each consensus strategy are analysed which helps in choosing a consensus to be followed in generating group recommendations based on the group requirements and the data available. As not much attention was given to consensus negotiation, this work acted as a starting step to tackle this problem.

The authors in Chen et al. (2013) propose a model that generates travel recommendations to both individuals and groups by mining peoples' attributes from data like GPS logs, geo-tagged photos etc. This model also uses data from the community-contributed photos by giving tags to the groups. User profiles are prepared based on various factors like age, gender, race, etc., and group profiles are prepared based on the type of group, i.e., friends, family, couple, etc. Profiling of photos is done in two ways: profiling by location and profiling by time. Then a recommendation model based on probabilistic Bayesian learning is applied considering all the deduced factors and group recommendations are thus generated. The experimental results have shown that the prediction of a travel group from the photos improves the recommendation accuracy to a great extent. This system can also be extended by including some more features like travel duration and season of travel, etc. The model proposed in Park et al. (2008) uses a similar learning model, i.e. it uses Bayesian networks to recommend restaurants to user groups.

Table 10 gives us the information regarding group travel recommender systems.

Table 10 Group recommender systems for travel

System	Aggregation models used	User preferences
CATS (McCarthy et al. 2006)	Aggregated voting	Implicit, explicit
e-Tourism (Garcia et al. 2011)	Aggregated voting	Explicit
ConNeg (Salamó et al. 2012)	Least misery, aggregated voting, most pleasure, multiplicative, borda count	Explicit
TravelRec (Chen et al. 2013)	–	Implicit
MobileRec (Nguyen and Ricci 2017)	Weighted average aggregation	Explicit

4.4 Group recommender systems for TV programs

The group recommender system proposed in Sotelo et al. (2009) recommends TV programs to a group of users. The authors mainly consider the need of selecting the consensus strategy based on the composition of the group. In addition to the user preferences, this system also requires TV content meta-data. The proposed model is mainly based on profile aggregation strategies, in which each content profile contains four attributes, $Intention$, $Format$, $Content$, $IntendedAudience$. Each user profile contains four vectors each containing these attributes based on her preferences. For every pair of users, the correlation factor between their profiles is calculated. Based on these correlation factors of all the user pairs in a group, it is classified as either homogeneous or heterogeneous. An appropriate consensus is then selected based on the type of composition of the group.

A video recommender system to the known groups with the help of viewing history and the preferences of the users in a group is proposed in Seko et al. (2011). The system initially estimates the recommendations based on the watch history and genre ratings of the users. Then, the content is judged based on the preferences and changes are made accordingly. Here, preference is used in a different sense; it means the tendency of power balance among the members of a group. A Power Balance Map (PBM) is constructed between each pair of users, and the new content is matched with the watched content based on the PBM and calculates the parameter used to estimate the *similarity level* among them. This *similarity level* parameter is calculated with all the viewing history of users in a group and is summed up to *Useful Level* S_c. If the *Useful Level* value exceeds the threshold, then the new item can be considered in the recommendation. Experimental results show that the proposed algorithm outperforms the baseline algorithms.

In the model proposed in Seko et al. (2011), $Groupality$ is introduced as the entity or entities that characterise groups and the relationship between members of a group. In addition to the commonly used user preference ratings, the behavioural history of users is also used for the generation of recommendation. The recommendation is generated using the power balance map that consists both the user preferences and the behavioural history of the group. Firstly, *Genre Frequency*(GF) is calculated based on the behavioural history of a group, and then power balance map is created using individual ratings, and then *Behavioral tendency Score* (BS) is calculated as the average of *Genre Frequency* for each item. The final recommendation is calculated using similarities between *Normalised groupality Vector* (NV) formed by merging normalised user ratings and the Behavioral tendency Score. Experimental results showed that including behavioural tendencies in the power balance map increased the precision of the generated group recommendations. The approach in Seko et al. (2011) also uses Power Balance Map to generate video recommendations to a group.

Table 11 gives various aggregation models and types of user preferences used in group recommender systems for TV programs.

Table 11 Group recommender systems for TV programs

System	Aggregation model used	User preferences
TV-A (Sotelo et al. 2009)	Average without misery	Implicit
PBRec (Seko et al. 2011)	–	Implicit, explicit
VideoRec (Seko et al. 2011)	–	Implicit, explicit

4.5 Other group recommender systems

Pocket Restaurant Finder (Mccarthy 2002a) is an extension to the idea used in $MusicFX$ (McCarthy and Anagnost 1998) implemented in the domain of restaurants. Here, along with culinary preferences of users, it should also address the location preferences of the users of a group like how far that group is willing to travel, travel time to the location, etc. When a group visits a new place and wishes to find a restaurant nearby, this system finds a restaurant based on the preferences of the users and the previous public ratings of the restaurants in the specified location. The system also takes care of the privacy of users in a group by not letting know the preferences of one user to the other.

A model that recommends restaurants for a group of people in mobile environments using probabilistic multi-criteria decision making is proposed in Park et al. (2008). As mobile environments would be quite dynamic and uncertain, to make the system reliable, this model uses Bayesian networks to model individual preferences. This model also employs multi-criteria decision-making method AHP (Analytical Hierarchy Process) for modelling the group preference from the individual users' preferences. The whole process is divided into four steps: collection of context-log data, preference modelling using the Bayesian network, Group modelling with AHP and generating recommendations. Context log data may include location information extracted using GPS, climate information like temperature, weather by using the web, and some more useful information like time, season, etc. Experimental results reveal that this model outperforms rule-based models. The model proposed in Chen et al. (2013) uses a similar approach for travel recommendations to groups.

To handle the data sparsity problem in group recommender systems, the authors in Baskin and Krishnamurthi (2009) present an aggregation algorithm that aggregates the users' relative preferences based on the Kemeny-optimal ordering of items. This ordering is also used to classify items as good or bad items for the recommendation. As optimal Kemeny ordering is NP-hard, this model uses a near optimal local search algorithm to reduce the complexity. It employs a modified version of the variable neighbourhood search called Variable Neighbourhood with the Cascade heuristic (VNS-CD) to order the items and is found to produce better results than the conventional BESTFIT algorithm. This model is implemented to suggest articles to a group of students in a library.

The main factor that can define the success of managing an online community is the retention of members in a group. Many group recommendation techniques are based on aggregation methods or collaborative filtering based methods which leave some part of the group unsatisfied with the results. So, to handle such issues, a novel Group Recommendation system for Online Communities (GRec-OC) (Kim et al. 2010) is proposed that works in two phases. In the first phase, a typical collaborative filtering method that has been used by most group recommendation systems is used. In the second phase, the items in the recommendation set obtained from the first phase, which are irrelevant are removed to improve the satisfaction of users in a group. The proposed technique is evaluated on a book recommendation system. Experimental results have shown that the percentage of dissatisfied users by the recommendation obtained is low for the proposed model.

The authors in (Berkovsky and Freyne 2010) design a collaborative filtering based group recommendation algorithm that also involves aggregation of group data. This work particularly works on recipe recommendation problem to family groups, friends, etc. From the analysis of different recommendation aggregation techniques, it is found that the best group recommendation results are achieved when individual user models are aggregated into group models. The aggregation of individual members should be done in a weighted manner, such

Table 12 Other group recommender systems

System	Aggregation model used	User preferences	Domain
ExternalExpert (Wang et al. 2012)	–	Implicit	Video
PocketRestaurantFinder (Mccarthy 2002a)	Aggregated voting	Explicit	Restaurants
SIS (Ye et al. 2012)	Least misery, aggregated voting	Implicit	Social networks
P&TRec (Quijano-Sanchez et al. 2010)	Aggregated voting	Explicit	Social networks
ResearchAssist (Baskin and Krishnamurthi 2009)	Aggregated voting, kemeny ordering	Explicit	Research papers
BookRec (Ahmad et al. 2017)	Least misery, aggregated voting, most pleasure	Explicit, implicit	Books
Grec-OC (Kim et al. 2010)	Aggregated voting	Explicit, implicit	Books
RecipeRec (Sotelo et al. 2009)	Least misery, most pleasure, aggregated voting	Negative feedback, explicit	Food

that the weights are given following the interactions of an individual with the other members within the group. It is found that *Switching Hybridisation* model increases the accuracy of the system. The systems proposed in Yuan et al. (2014) and Recio-Garcia et al. (2009) also aggregate the user models by assigning weights to the users.

Table 12 gives us the details regarding aggregation models and user preferences used in various other domains.

5 Impact of social and behavioural aspects on group recommendations

As the group decision sometimes get more influenced by some members of the group, incorporating social and behavioural factors in the group recommendations (Quijano-Sanchez et al. 2010) along with user preferences will increase the quality of recommendations produced. Social factors like personality, tie strength, satisfaction, etc., will give additional personalised explanations (Quijano-Sanchez et al. 2017) to produce good recommendations. Negotiation based approaches have been evolved in which people interact and negotiate the preferences and finally settle down to some item set based on the agreement of all the group members. However, such methods only work for groups having smaller size and fail for larger groups.

The model proposed in Gartrell et al. (2010) uses social factors like social descriptor, expertise descriptor, dissimilarity descriptor and designs a rule-based system that incorporates the deduced social factors for a group recommendation. Cognitive modelling of user preferences along with social interactions among the group members increase the recommendation quality (Quijano-Sánchez et al. 2014). In group recommendations for heterogeneous groups, as there will not be many social relations among the group members, incorporating behavioural factors along with the preferences will lead to better recommendations. The model proposed in Recio-Garcia et al. (2009) uses weighted user collaborative filtering along with group personality composition to generate better recommendations than traditional methods. Group behavioural history and power balance mapping to calculate behavioural tendency score which is used along with user preferences for producing recommendations is discussed in Seko et al. (2011). External followee information in a microblogging platform is also used to correlate the interests of the group members based on that followee information (Wang et al. 2012).

6 Conclusions and future work

Recommender Systems are useful tools as they assist a user in making decisions and choosing among the large set of items. Recommender systems use the preferences and characteristics of a user while generating a suggestion to her. Due to the increase in social activities and mobile devices, there have been many instances where many activities are done in groups, and thus increased the need for the development of group recommender systems. This paper presents the state-of-the-art in group recommendation systems. We reviewed various aggregation models and different types of preference acquisitions concerning various popular domains. The inclusion of social aspects in group recommender systems increases the recommendation quality. In heterogeneous groups with no much social interaction among the group members, including behavioural aspects along with user preferences leads to better recommendations. Group recommendations can be made either by profile aggregation or by recommendation aggregation.

As we know that in many instances, groups are not readily formed, there are very few works that have focused on the formation of groups among the available members. So, there is much scope in the study of the formation of groups based on their interests and social factors in the group recommendations. Impact of including group-specific features in recommendations for heterogeneous groups can be explored. A study to consider internal and external social relations of group members may be carried out for better group recommendations. The group recommendation models fail to generate better recommendations in case of data sparsity on user-item preferences. There is much scope for future research in group recommendations in various data sparsity situations.

Most of the existing models use either rule-based recommendation strategies or Machine Learning techniques to generate group recommendations. We strongly believe that incorporating both rule-based and machine learning techniques in group recommendation may lead to better results. Many of the past works evaluated their proposed models for the quality of group recommendations but not by the complexity of the model. So, there is a need to perform a qualitative study on the complexity analysis and study the group recommendation models, which is very much required to incorporate these algorithms in real time scenarios. A good future direction may be explored for the role of flexible size preferences and order in group recommendations. Currently, the existing algorithms are designed to generate the recommendation to the group taking a fixed size preference list from all the users in the group. Considering order in preference list of users is relatively a new direction to explore further as it is a very recent major exploration (Agarwal et al. 2017). Dimensionality reduction and graph-based techniques can be explored in the context of group recommendations. Dimensionality reduction helps to represent multidimensional data in two-dimensional space. It gives a compact representation of users and items and also contains the latent features to recommend items accurately. Neighbourhood based approaches can be further explored in the context of group recommendations.

References

Agarwal, A., Chakraborty, M., Ravindranath Chowdary, C. (2017). Does order matter? effect of order in group recommendation. *Expert Systems with Applications*, *82*(Supplement C), 115–127.

Ahmad, H.S., Nurjanah, D., Rismala, R. (2017). A combination of individual model on memory-based group recommender system to the books domain. In *5Th international conference on information and communication technology (ICoIC7)* (pp. 1–6).

Baltrunas, L., Makcinskas, T., Ricci, F. (2010). Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the Fourth ACM conference on recommender systems, RecSys '10* (pp. 119–126). New York: ACM.

Baskin, J.P., & Krishnamurthi, S. (2009). Preference aggregation in group recommender systems for committee decision-making. In *Proceedings of the Third ACM conference on recommender systems, RecSys '09* (pp. 337–340). New York: ACM.

Berkovsky, S., & Freyne, J. (2010). Group-based recipe recommendations: analysis of data aggregation strategies. In *Proceedings of the Fourth ACM conference on recommender systems, RecSys '10* (pp. 111–118). New York: ACM.

Bobadilla, J., Ortega, F., Hernando, A., Bernal, J. (2012). Generalization of recommender systems Collaborative filtering extended to groups of users and restricted to groups of items. *Expert Systems with Applications*, *39*(1), 172–186.

Boratto, L., & Carta, S. (2010). State-of-the-art in group recommendation and new approaches for automatic identification of groups. *Information Retrieval and Mining in Distributed Environments*, *324*, 1–20.

Boratto, L., Carta, S., Satta, M. (2010). Groups identification and individual recommendations in group recommendation algorithms. In *PRSAT@ recsys* (pp. 27–34).

Castro, J., Quesada, F.J., Palomares, I., Martínez, L. (2015). A consensus-driven group recommender system. *International Journal of Intelligence Systems*, *30*(8), 887–906.

Castro, J., Yera, R., Martínez, L. (2017). An empirical study of natural noise management in group recommendation systems. *Decision Support Systems*, *94*(Supplement C), 1–11.

Chao, D.L., Balthrop, J., Forrest, S. (2005). Adaptive radio: achieving consensus using negative preferences. In *Proceedings of the 2005 international ACM SIGGROUP conference on supporting group work, GROUP '05* (pp. 120–123). New York: ACM.

Chen, Y.Y., Cheng, A.J., Hsu, W.H. (2013). Travel recommendation by mining people attributes and travel group types from community-contributed photos. *IEEE Transactions on Multimedia*, *15*(6), 1283–1295.

Chen, Y.-L., Cheng, L.-C., Chuang, C.-N. (2008). A group recommendation system with consideration of interactions among group members. *Expert Systems with Applications*, *34*(3), 2082–2090.

Christensen, I.A., & Schiaffino, S. (2011). Entertainment recommender systems for group of users. *Expert systems with applications*, *38*(11), 14127–14135.

Christensen, I.A., & Schiaffino, S. (2011). Entertainment recommender systems for group of users. *Expert Systems with Applications*, *38*(11), 14127–14135.

Colomer, J.M. (2013). Ramon llull: from 'ars electionis' to social choice theory. *Social Choice and Welfare*, *40*(2), 317–328.

Crossen, A., Budzik, J., Hammond, K.J. (2002). Flytrap: intelligent group music recommendation. In *Proceedings of the 7th international conference on intelligent user interfaces, IUI '02* (pp. 184–185). New York: ACM.

De Pessemier, T., Dhondt, J., Vanhecke, K., Martens, L. (2015). Travelwithfriends: a hybrid group recommender system for travel destinations. In *Workshop on tourism recommender systems (touRS15), in conjunction with the 9th ACM conference on recommender systems (recsys 2015)* (pp. 51–60).

Felfernig, A., Boratto, L., Stettinger, M., Tkalčič, M. (2018). Algorithms for group recommendation. Group Recommender Systems : An Introduction pp 27–58.

Felfernig, A., Boratto, L., Stettinger, M., Tkalčič, M. (2018). Evaluating group recommender systems. Group Recommender Systems : An Introduction pp 59–71.

Felfernig, A., Boratto, L., Stettinger, M., Tkalčič, M. (2018). Handling preferences. Group recommender systems : an introduction pp 91–103.

Garcia, I., Sebastia, L., Onaindia, E. (2011). On the design of individual and group recommender systems for tourism. *Expert Systems with Applications*, *38*(6), 7683–7692.

Gartrell, M., Xing, X., Lv, Q., Beach, A., Han, R., Mishra, S., Seada, K. (2010). Enhancing group recommendation by incorporating social relationship interactions. In *Proceedings of the 16th ACM international conference on supporting group work, GROUP '10* (pp. 97–106). New York: ACM.

Ghazarian, S., & Ali Nematbakhsh, M. (2015). Enhancing memory-based collaborative filtering for group recommender systems. *Expert Systems with Applications*, *42*(7), 3801–3812.

Gorla, J., Lathia, N., Robertson, S., Wang, J. (2013). Probabilistic group recommendation via information matching. In *Proceedings of the 22Nd international conference on world wide Web, WWW '13* (pp. 495–504). New York: ACM.

Guzzi, F., Ricci, F., Burke, R. (2011). Interactive multi-party critiquing for group recommendation. In *Proceedings of the Fifth ACM conference on recommender systems, RecSys '11* (pp. 265–268). New York: ACM.

Jameson, A., & Smyth, B. (2007). Recommendation to groups. In *The adaptive Web: methods and strategies of Web personalization* (pp. 596–627).

Kaššák, O., Kompan, M., Bieliková, M. (2016). Personalized hybrid recommendation for group of users: top-n multimedia recommender. *Information Processing & Management*, *52*(3), 459–477.

Kim, H.-N., & El Saddik, A. (2015). A stochastic approach to group recommendations in social media systems. *Information Systems*, *50*(Supplement C), 76–93.

Kim, J.K., Kim, H.K., Oh, H.Y., Ryu, Y.U. (2010). A group recommendation system for online communities. *International Journal of Information Management*, *30*(3), 212–219.

Kompan, M., & Bielikova, M. (2014). Group recommendations: survey and perspectives. *Computing and Informatics*, *33*(2), 446–476.

Lieberman, H., Dyke, N.V., Vivacqua, A. (1999). Let's browse: a collaborative browsing agent. *Knowledge-Based Systems*, *12*(8), 427–431.

Liu, X., Tian, Y., Ye, M., Lee, W.-C. (2012). Exploring personal impact for group recommendation. In *Proceedings of the 21st ACM international conference on information and knowledge management, CIKM '12* (pp. 674–683). New York: ACM.

Masthoff, J. (2011). Group recommender systems: combining individual models. In *Recommender systems handbook* (pp. 677–702). Berlin: Springer.

Mccarthy, J.F. (2002). Pocket restaurant finder: a situated recommender systems for groups. In *Proceeding of workshop on mobile ad-hoc communication at the 2002 ACM conference on human factors in computer systems*.

Mccarthy, J.F. (2002). Pocket restaurantfinder: a situated recommender system for groups. In *Workshop on mobile ad-hoc communication at the 2002 ACM conference on human factors in computer systems*.

McCarthy, J.F., & Anagnost, T.D. (1998). Musicfx: An arbiter of group preferences for computer supported collaborative workouts.

McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., Nixon, P. (2006). CATS A Synchronous approach to collaborative group recommendation. In *Proceedings of the Nineteenth international Florida artificial intelligence research society conference, Melbourne Beach, Florida, USA, May 11–13, 2006* (pp. 86–91).

Dery, L.N., Kalech, M., Rokach, L., Shapira, B. (2010). Iterative voting under uncertainty for group recommender systems. In *Proceedings of the Fourth ACM conference on recommender systems, RecSys '10* (pp. 265–268). New York: ACM.

Nguyen, T.N., & Ricci, F. (2017). Proceedings of the symposium on applied computing, SAC '17 (pp. 1685–1692). New York: ACM.

O'Connor, M.J., Cosley, D., Konstan, J.A., Riedl, J. (2001). Polylens: a recommender system for groups of users. In *ECSCW 2001: Proceedings of the Seventh European conference on computer supported cooperative work 16–20 September 2001, Bonn, Germany* (pp. 199–218). Dordrecht: Springer.

Ortega, F., Bobadilla, J., Hernando, A., Gutiérrez, A. (2013). Incorporating group recommendations to recommender systems: alternatives and performance. *Information Processing & Management*, *49*(4), 895–901.

Park, M.-H., Park, H.-S., Cho, S.-B. (2008). Restaurant recommendation for group of people in mobile environments using probabilistic multi-criteria decision making. In *Proceedings of the 8th Asia-Pacific conference on computer-human interaction, APCHI '08* (pp. 114–122). Berlin: Springer.

Pera, M.S., & Ng, Y.-K. (2013). A group recommender for movies based on content similarity and popularity. *Information Processing and Management*, *49*(3), 673–687.

Quijano-Sanchez, L., Recio-Garcia, J.A., Diaz-Agudo, B. (2010). Personality and social trust in group recommendations. In *2010 22Nd IEEE international conference on tools with artificial intelligence*, (Vol. 2 pp. 121–126).

Quijano-Sánchez, L., Díaz-agudo, B., Recio-garcía, J.A. (2014). Development of a group recommender application in a social network. *Knowledge-Based Systems*, *71*(1), 72–85.

Quijano-Sánchez, L., Recio-garcía, J.A., Díaz-agudo, B., Jiménez-díaz. G. (2011). Happy movie: a group recommender application in facebook. In *Proceedings of the twenty fourth international Florida artificial intelligence research society conference, FLAIRS 11* (pp. 419–420). Florida: AAAI Press.

Quijano-Sanchez, L., Sauer, C., Recio-Garcia, J.A., Diaz-Agudo, B. (2017). Make it personal: a social explanation system applied to group recommendations. *Expert Systems with Applications*, *76*(Supplement C), 36–48.

Rakesh, V., Lee, W.-C., Reddy, C.K. (2016). Probabilistic group recommendation model for crowdfunding domains. In *Proceedings of the Ninth ACM international conference on Web search and data mining, WSDM '16* (pp. 257–266). New York: ACM.

Recio-Garcia, J.A., Jimenez-Diaz, G., Sanchez-Ruiz, A.A., Diaz-Agudo, B. (2009). Proceedings of the Third ACM conference on recommender systems, RecSys '09 (pp. 325–328). New York: ACM.

Basu Roy, S., Thirumuruganathan, S., Amer-Yahia, S., Das, G., Yu, C. (2014). Exploiting group recommendation functions for flexible preferences. In *30Th IEEE international conference on data engineering* (pp. 412–423).

Salamó, M., Mccarthy, K., Smyth, B. (2012). Generating recommendations for consensus negotiation in group personalization services. *Personal and Ubiquitous Computing*, *16*(5), 597–610.

Salehi-Abari, A., & networks, C.B. (2015). Preference-oriented social group recommendation and inference. In *Proceedings of the 9th ACM conference on recommender systems, RecSys '15* (pp. 35–42). New York: ACM.

Seko, S., Motegi, M., Yagi, T., Muto, S. (2011). Video content recommendation for group based on viewing history and viewer preference. In *2011 IEEE International conference on consumer electronics (ICCE)* (pp. 351–352).

Seko, S., Yagi, T., Motegi, M., Muto, S. (2011). Group recommendation using feature space representing behavioral tendency and power balance among members. In *Proceedings of the Fifth ACM conference on recommender systems, RecSys '11* (pp. 101–108). New York: ACM.

Shi, J., Wu, B., Lin, X. (2015). A latent group model for group recommendation. In *2015 IEEE International conference on mobile services* (pp. 233–238).

Sotelo, R., Blanco, Y., Lopez, M., Gil, A., Pazos, J. (2009). Tv program recommendiation for groups based on multidimensional tv-anytime classifications. In *2009 Digest of technical papers international conference on consumer electronics* (pp. 1–2).

Wang, X.iaoyan., Sun, L.ifeng., Wang, Z.hi., Da, M.eng. (2012). Group recommendation using external followee for social tv. In *IEEE International conference on multimedia and expo (ICME) 2012* (pp. 37–42). Piscataway: IEEE.

Ye, M., Liu, X., Lee, W.-C. (2012). Exploring social influence for recommendation: A generative model approach. In *Proceedings of the 35th international ACM SIGIR conference on research and development in information retrieval, SIGIR '12* (pp. 671–680). New York: ACM.

Yuan, Q., Cong, G., Lin, C.-Y. (2014). Com: A generative model for group recommendation. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '14* (pp. 163–172). New York: ACM.