Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference

Power of the Group Neighborhood in Memory-Based Group Recommender Systems

Nadia A Najjar and **David C Wilson**
Univesity of North Carolina at Charlotte
9201 University City Blvd.
Charlotte, NC 28223
{nanajjar, davils}@uncc.edu

Abstract

Recommender Systems play a significant role in helping users identify items worthwhile for them to consume. With the increase of adopting such systems a need for systems that help a group of users identify such items for the whole group to consume together has emerged. Early research has focused on strategies that combine individual preferences to generate group preferences without much consideration of the group context in the recommendation technique. In this paper, we explore neighborhood selection in the group context when applying a neighborhood-based Collaborative Filtering approach to recommendation. We identify several neighborhoods that are related to the group context and investigate their effect on recommendation accuracy when employing a neighborhood-based Collaborative Filtering. We evaluate the performance of such neighborhoods with respect to the group recommendation technique and the group size. We measure performance using a *success@n* measure. Results show improvements on the success rate of recommendations when identifying a neighborhood, for the group as a whole, rather than basing the recommendation on only the individual neighborhoods of the group members.

Introduction

Recommender systems were originally thought of as systems that are geared to help individual users find items of interest as they navigate through the large content space. As these systems have become ubiquitous with the advancement of the web and online, interactive applications a need has emerged for such systems to provide and tailor recommendations to a group of users rather than individuals (Jameson and Smyth 2007; Baltrunas, Makcinskas, and Ricci 2010).

Group recommender systems balance the preferences of individual members holistically across an entire group of users, in order to create a recommendation that is applicable to the group as a whole. Common group recommendation domains involve social and shared-consumption elements, for example: watching a movie together (O'Connor et al. 2001; Goren-Bar and Glinansky 2004; Senot et al. 2010); eating together (McCarthy 2002; Berkovsky and Freyne 2010), or traveling together (McCarthy et al. 2006; Ardissono et al. 2003; Jameson 2004).

Prior work on group recommendation has mainly focused on the modeling of the group. The notion behind this is to reduce the problem to individual recommendation taking advantage of already validated approaches in that context. This had led to two main group recommendation strategies (Jameson and Smyth 2007). The first one is based on merging the individual profiles of the group members into one representative group profile (e.g., (O'Connor et al. 2001)) or *pseudo-user*. In the second approach, the individual recommendation lists, or predictions computed for each group member, are merged into one recommendation list presented to the group (e.g., (Quijano-Sánchez, Recio-García, and Díaz-Agudo 2011; Recio-Garcia et al. 2009)). The aggregation techniques are commonly inspired by Social Choice Theory and center around modeling the achievement of consensus among the group (Masthoff 2004). Variations have also been investigated that consider personalities of, and social interactions among, group members (Gartrell et al. 2010; Recio-Garcia et al. 2009).

One of the widely used recommendation techniques is Collaborative Filtering (CF) which is based on estimating relations among users or items in order to predict unknown preferences (Breese, Heckerman, and Kadie 1998). This has led to a wide focus of exploring group-based recommendations in this context (Quijano-Sanchez et al. ; Berkovsky and Freyne 2010; Baltrunas, Makcinskas, and Ricci 2010). In this paper we focus on group recommendation employing ratings-based user profiles as a foundation for making recommendations. Common methods of CF are based on neighborhood models which estimate relations among users or items in order to predict unknown ratings. The "neighbors" are users that share similar preferences to the user targeted for recommendation on commonly rated items. Sarwar et al. (Sarwar et al. 2000) divided CF based recommendation into three sub-tasks, *representation*, *neighborhood formation*, and *recommendation generation*. Our previous work (Najjar and Wilson 2014) projected these tasks into the group-based recommendation process focusing on group context as an explicit factor in the *neighborhood formation* and the *recommendation generation* subtasks by identifying a neighborhood that is shared among all the group members.

In this paper, we extend this research by examining various *neighborhood formation* models as well as other *recommendation generation* sub-tasks employing a different dataset that includes "real" user judgment as a baseline for evaluation.

Related Work

Early research on group recommendation has investigated the core group models used for aggregation in generating the group recommendations. More recently, a shift of focus on the recommendation technique itself has occurred. Chen et al. (Chen, Cheng, and Chuang 2008) used a Genetic Algorithm (GA) to exploit known preferences of subgroups of the active group and predict possible similarities among group members. These similarities were used to weight member contributions in item predictions. One limitation of their approach is that it is based on having access to some item ratings for the target group as well as subgroups of the target group and individual group members' preference information. They use an item-based CF approach to identify items similar to the item under consideration for prediction. If the group did not provide a rating for these items a user-based CF was used to predict the individual ratings. Subgroup information was exploited using a GA to assign weights in combining the individual users' ratings into a group rating. Then item-based CF was used to calculate the final group rating for the target item.

Berkovsky et al. (Berkovsky and Freyne 2010) investigated a different approach for weighting users, in a group-based recommender using CF, rather than the commonly used user-to-user similarity weighting. They implemented four weighting models (*uniform, heuristic, role-base, family-log*) for aggregating individual data rather than using a similarity. The uniform model weights users uniformly, i.e., weight for every user equals 1. The heuristic model is role-based, where a role refers to a user's function within a family: applicant, partner, or child. A user's weight is defined solely by their role. The role-based model weights users according to the activity of users in the same role across the entire community. The family-log model weights users according to their activity in relation to other family members. They evaluated CF recommendations generated using their approach against real-life recipe ratings provided by families interacting with an experimental eHealth portal. The results showed that the most appropriate family-based recipe recommendation strategy should aggregate individual user models, rather than individual recommendations, and weight individual users according to their observed activity rather than according to predefined preferences.

Similarly, Recio-Garcia et al. (Recio-Garcia et al. 2009) described a group recommender system that takes into account the personality types of the group members as the basis for weighting user contribution using a CF approach. They reported that *Average* and *Least Misery* with personality weighting reflected improvements in the accuracy of the recommendations.

For the neighborhood selection component of a CF system Najjar et al. (Najjar and Wilson 2014) investigated a neighborhood selection approach based on the group as a whole rather than the individual user neighborhoods. Their evaluation explored assigning a higher weight to users that are shared by all the group members in calculating the individual group members' predictions. They evaluated their approach across various group sizes and group similarity levels using simulated groups from the MovieLens 1M[1] dataset. Their results showed an increase in prediction accuracy when combining their approach with the profile aggregation group recommendation strategy for groups of high cohesiveness among the group members. In this research, we build on this work by expanding the neighborhood selection component to identify several neighborhoods in the group context as well as a weighting approach based on frequency rather than a constant. We also base our evaluation on "real" user judgment as the ground truth for the group preference rather than an average of the simulated groups' individual preferences.

The Group Neighborhood

Neighborhood-based CF identifies users that might be more beneficial to the user targeted for recommendation. The basis in this approach is that each person belongs in a larger group of like-minded users with similar histories of preferences. Employing user-to-user similarity, users are identified as neighbors of the active user. The relationships between these neighbors are used as part of the algorithmic calculation of generating recommendations. The individual neighborhoods of the group members are not the only influence for the recommendations in the group-based context. Neighbors that are shared by one or more group members might have more insight to the group context in comparison to neighbors that only appear in one of the group member's neighborhoods. We can consider the relationship between the individual group member and the neighbors of the other group members as well. These neighborhoods can also be exploited and might be beneficial for group recommendations.

In analyzing neighborhoods in the group context, we have identified three different neighborhoods as following and as outlined in Figure 1:

1. User Neighborhood: For any given group member, this contains the users that are identified as the neighbors of that group member. Here, there is no consideration of the group context only the individual neighborhoods.
2. Intersect Neighborhood: This contains the users that appear in every group member's neighborhood. In other words, these are the users that all the group members have in common.
3. Union Neighborhood: This contains the users that appear in any of the group member's neighborhoods. In other words, these are all the neighbors of all of the group members.

The basis for Neighborhood based CF algorithms is to calculate a similarity between users a and b, w_{ab} to identify the

[1] http://www.grouplens.org

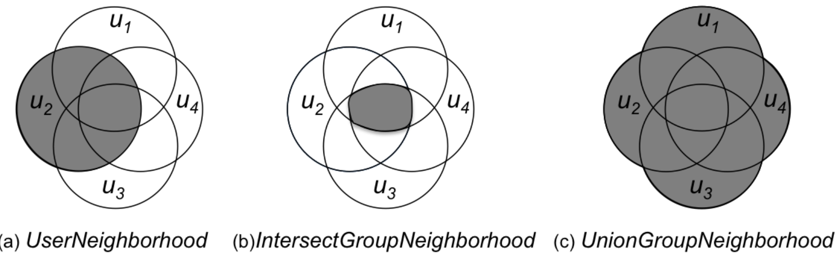

Figure 1: Neighborhoods identified in the group context

top k users that have the highest similarity to the targeted user for recommendation (Breese, Heckerman, and Kadie 1998). We use the Pearson correlation to compute this similarity as defined in (Herlocker, Konstan, and Riedl 2002).

To calculate the prediction of item i for user a, it is defined as:

$$p_{ai} = \overline{r}_a + \sigma_a \frac{\sum_{b=1}^{n} [(r_{bi} - \overline{r}_b) \cdot w_{ab}]/\sigma_b}{\sum_{b=1}^{n} w_{ab}} \tag{1}$$

Revising this equation to fit the context of group-based neighborhood models identified earlier, we first need to identify the neighborhood as shown in Figure 1. Details of the models we define based on these neighborhoods are in the next section. Once the neighborhood is identified for each group member, we use equation 2 to compute the individual predictions.

$$p_{ai} = \overline{r}_a + \sigma_a \frac{\sum_{b=1}^{n} [(\alpha(r_{bi} - \overline{r}_b) \cdot w_{ab})]/\sigma_b}{\sum_{b=1}^{n} w_{ab}} \tag{2}$$

where α = 1 if neighbor $\in$ *User_Neighborhood*
and α = x if neighbor $\in$ *Group_Neighborhood*
The value for x is determined based on the model used. If a weighted model is chosen, then a value greater than 1 would be assigned. If a frequency based model is used then x is determined by the number of times this neighbor appears in the individual group members' neighborhoods (i.e. the number of User Neighborhoods it is in for any group).

Once a prediction is calculated for each group member, an aggregate of these values is used as the prediction for the group. We implement the Average aggregation strategy. This is the basic and most commonly used group aggregation strategy that assumes equal influence among group members (Jameson and Smyth 2007). Let n be the number of users in a group and r_{ji} be the rating of user *j* for item *i*, then the group rating for item *i* is computed as follows:

$$Gr_i = \frac{\sum_{j=1}^{n} r_{ji}}{n} \tag{3}$$

Utilizing Group Neighborhood

Now that we have identified the neighborhoods that can be used as a part of the recommendation generation step, we formalize the following group neighborhood selection models:

- Group User Neighborhood (Baseline): employs only the users that appear in the group member's neighborhood in generating recommendations for that member. Recommendations are generated for each group member and then aggregated to generate a final group recommendation. This is the baseline recommendation with no consideration for the group context.
- Intersect Group Neighborhood (IntersectGN): employs only the users that appear in the intersect neighborhood of the group as the neighborhood for each group member. Recommendations are generated for each group member using this neighborhood and then aggregated to generate a final group recommendation.
- Union Group Neighborhood (UnionGN): employs the users that appear in the union neighborhood as the neighborhood for each group member. Recommendations are generated for each group member using this neighborhood and then aggregated to generate a final group recommendation.
- Group User Weighted Intersect (WtIntGN): employs the individual user neighborhoods as the neighborhood for each group member where neighbors that appear in the intersect neighborhood are assigned a different weight.
- Group User Frequency Intersect (FreqIntGN): employs the user neighbors of a group member, but weighs neighbors that appear in the intersect neighborhood using the number of the individual neighborhoods of the group members in which this neighbor appears. This turns out to be the size of the group.
- User Frequency Union (FreqUnGN): employs the baseline neighbors of a group member but weighs neighbors with the number of neighborhoods in which this neighbor appears.
- Union Frequency Union (UnionFreqUnGN): employs the union neighborhood as the neighborhood for each group member and neighbors are weighted by the number of group member's neighborhoods in which they appear.
- Union weighted union (WtUnGN): employs the union neighborhood as the neighborhood for each group member and neighbors are weighted more if they appear in more than one of the group member's neighborhoods.

We hypothesize that group prediction based on group selection models and frequency are more beneficial for the group than predictions based on the individual members' neighborhoods.

Evaluation Setup

Dataset: In this evaluation, we use a dataset we obtained from (Quijano-Sánchez et al. 2012). A common evaluation approach of group-based systems is to simulate groups from individual datasets (De Pessemier, Dooms, and Martens

2012; Baltrunas, Makcinskas, and Ricci 2010). This dataset utilizes this approach but, rather than basing the group preference on a model of the individual preferences, they employed human experts to evaluate the group's preferences and produce a group decision on which to base and ground the evaluation. Following is a detailed explanation of this dataset.

The baseline dataset used is the MovieLens 1M dataset. Building an effective CF recommendation system requires sufficient data. This data set provides the basis for that. This dataset contains 1 million ratings, on a scale of 1 to 5, for 6040 users and 3952 movies. Each user has at least 20 ratings. The dataset also gives a small amount of demographic information about each user. In particular, they use the user's gender and age range (under 18, 18- 24, 25 -34, and so on).

They created 100 groups from this dataset. Group members were chosen at random from all users, but subject to the following restrictions: 1) In a group, users are distinct (but a user may be in more than one group). 2) In a group, they ensure that all the users are in the same age range. 3) In a group, they ensure that there are at least 15 movies which are co-rated by all members of the group. These 15 movies will be the test items for the group.

They conducted a Facebook poll in which they asked respondents to tell them, for the last five times that they went to the movie theatre as a group, how large the group was. There were 105 respondents that reported the group size for 525 events. They used the frequencies from this distribution to create 100 groups. The break down of the groups is as follows: 50 groups of size 2, 18 of size 3, 16 of size 4, 7 of size 5, 5 of size 6, and 4 where they set the size to be 7.

To establish ground truth to be used as the baseline for the evaluation, they used four human experts who were given all the information about a group's members and the candidate movies (test items), including the actual ratings by the members of the group for the items in their test set. The experts were asked to decide on which of the movies the group would be most likely to settle. Each expert evaluated 50 cases, hence each of the 100 groups was evaluated by two experts (not always the same two). Experts were asked to give an ordered list of three movies from the test set on which they thought the members of the group would agree. They combined the experts' judgements into a single, final ordered list of size three.

Since we are interested in evaluating our recommendation approach for both profile merging and recommendation aggregation, we wanted to ensure that the same training set was used to generate all the predictions for that group. We created a training and testing set for each group based on the test set of the group. We first created the profiles of the pseudo users for each group by merging the individual group members' ratings based on the average aggregation strategy. For each item rated by one or more group member, the rating for the pseudo user would be the average of the ratings based on the number of the group members that rated it. Once the profiles for the pseudo users were created we added the pseudo profile to the original data set to include the new pseudo user. This ensured that the same training set was used to generate predictions for the group across all the evaluated techniques, both merging profiles and merging recommendations.

To create the training set for each group, we started off with the original MovieLens dataset. We then added the profile of the pseudo user of that group to the dataset. We then took out the ratings of the test items identified for that group from each of the group member's profiles and the pseudo user. In other words, the training set for each group is the original MovieLens dataset plus that group's pseudo user profile minus the ratings for the test items for that group, for each of the group members and the pseudo user of that group.

We explored outcomes of prediction accuracy for profile merging and recommendation aggregation using the Average group aggregation strategy. We analyzed the results across the various group sizes. We made a comparison between the baseline nearest neighborhood recommendation technique and the Group Neighborhood Selection techniques as outlined in the previous section. We compared recommendation rankings based on prediction to the item's ranking provided by the experts using the evaluation metric outline in the following section.

Evaluation Metric: To evaluate the performance of the implemented recommendation techniques we compared the recommended list of items to the actual preferences list. A variant of this strategy, *success@n*, was employed in (Quijano-Sánchez et al. 2012) to measure the rate of having at least one recommended item in the top *n* positions of the actual preferences list. For example, given an ordered set of recommended items *recList* of size *n* and an ordered set of the actual preferences *actList* of the same size, *success@3* would return 1 if at least one of the items in the top 3 positions of *recList* appeared in the top 3 positions of *actList*, and 0 otherwise. We used the *success@n=3* metric in this evaluation. For each recommendation technique we measured the *success@3* for each group using each expert's list as the benchmark for evaluation. For an overall success rate we averaged the results across the different group sizes and then averaged the results from each expert.

Recommendation Algorithm Settings: For selecting the neighbors to form the user neighborhood, Herlocker et al. (Herlocker, Konstan, and Riedl 2002) recommend setting a maximum neighborhood size in the range of 20 to 60 users. We set the neighborhood size to 50 users. This would be the maximum size for the individual user neighborhood (Fig.1-a).

For the weighted group neighborhood models, we set the weight for users that appears in more than one user neighborhood among the group members to be 2 ($\alpha = 2$). This assigns a higher weight to those users reflecting a higher influence on the recommendation.

Results and Discussion

Profile Merging: We first analyzed our results for the Profile Merging recommendation technique. Table 1 includes the success rate for the various Group Neighborhood Selection Models we implemented across the different group sizes. For groups of size 2, the WtIntGN, FreqIntGN and the FreqUnGN models outperform the baseline and the other models with a success rate of 87.5%. All models except the IntGN and UnionGN models had an increase in success rate when compared to the Baseline neighborhood approach . For groups of size 3 the FreqUnGN models perform best with a success rate of 88.89%. Similar to groups of size 2, all models, except the IntGN and UnionGN models, had an increase in success rate when compared to the baseline neighborhood approach .

For groups of size 4 all the models returned the same success rate of 100% except, for the IntGN and the FreqUnGN model which had a lower success rate of 87.5%. For Groups of size 5, the UnionGN model outperformed the others with a 100% success rate. The FreqUnion, UnFreqUnGN, and WtUnGN models had a decrease in success rate when compared to the Baseline Neighborhood model. While for groups of size 6, all models except the FreqUnionGN model, performed similarly scoring a 100%. There was no change across the models for groups of size 7 with a success rate of 75%.

Recommendation Aggregation: Next, we analyzed the results for the recommendation aggregation technique. Table 2 includes the success rate for the implemented models across the different group sizes. For groups of size 2, the WtIntGN, FreqIntGN, and the FreqUnGN models outperform the baseline and the other models with a success rate of 81.25%. This result goes along with the results obtained for the profile merging approach for the same group size. The models based on the UnionGN as the individual user's neighborhood had a lower success rate than the baseline and other models.

For groups of size 3, the WtIntGN model performed best with a success rate of 88.89%. Similar to the performance of groups of size 2, the models based on the UnionGN, as the individual user's neighborhood, had a lower success rate than the baseline and other models while the FreqIntGN model performed best for groups of size 4 having a 100% success rate. Again, we noticed that the models based on the UnionGN, as the individual user's neighborhood, had a lower success rate than the other models, and in particular, the Baseline. For groups of size 5, the Baseline, models based on the IntGN and the FrequUnGN model resulted in the same success rate of 100% and the other models had a decline in success rate. There was no change in performance for groups of size 6. For groups of size 7, all models except for the WtUnGN model had a success rate of 75%. That model had a lower success rate of 50%.

Across the Board: We also wanted to examine the performance of the models over all the groups. Table 3 includes the success values for all the groups across the various neighborhood models for both, Profile Merging and Recommendation Aggregation. From these results, we can see that the WtIntGN and FreqIntGN models perform best. With success rates of 88.76% and 86.73% for the Profile Merging and Recommendation Aggregation respectively. Results show similar patterns to the reported results spanning the group sizes. For the Profile Merging approach, all models except the IntGN model had an increase in success rate compared to the Baseline model. For the Recommendation Aggregation approach, models based on the individual user's neighborhood performed better than the models based on intersect, or union, as the individual user's neighborhood. Still, giving special consideration to neighbors that are shared with other members increased the success rate.

We recall from the dataset details that the majority of the groups fall in the size 2 category (50 groups of size 2, 18, 16, 7, 5, 4 for sizes 3, 4, 5, 6, and 7 respectively). We perceive the results obtained for this group size as more effective and indicative of the difference in performance of the evaluated models. For either group recommendation technique, assigning a higher weight to a neighbor, if they are shared with more than one group member, increased the success rate when compared to a baseline neighborhood approach. In the baseline neighborhood approach all neighbors are considered to have the same influence in the group context and neighbors are only weighted by their similarity to the individual group members.

Conclusion

In this paper, we carried out further exploration in the space of neighborhood identification in group-based recommendations when employing a CF recommendation technique. We identified the possible neighborhoods given a group context as well as a weighting scheme incorporating these neighborhoods. We have evaluated our approach using a success metric for a list of recommended items. We reported results for different group sizes (2-7) and group recommendation strategies (profile merging, recommendation aggregation).

For the profile merging strategy, accounting for a higher influence of users that are neighbors of more than one group member resulted in an increase in success rate when generating a list of three recommended items. Similarly, using a neighborhood that includes all the neighbors of the group members as the user neighborhood also increased the success rate. Since the profile merging approach creates a pseudo user that has the preferences for items given by all the group members, this neighborhood selection reflects that. Our results shows a 6% increase in success rate, compared to the baseline neighborhood CF.

Similarly, for the recommendation aggregation approach, accounting for a higher influence of users that are neighbors of more than one group member resulted in an increase in success rate (%4). On the other hand, using a neighborhood that includes all the neighbors of the group members as the user neighborhood decreased the success rate. We attribute this to the fact that this neighborhood might include neighbors that are not similar to the targeted group member resulting in a decrease in the prediction accuracy of the individual predictions which is carried out when aggregating them to finalize the group prediction.

Given that this evaluation is based on "real" user judgment, we believe that our approach has significant grounds in extending these results to real groups, not just synthesized groups. In the future, we plan to investigate this approach with a larger dataset using real groups rather than simulated ones.

	Baseline	IntGN	WtIntGN	FreqIntGN	UnionGN	FreqUnGN	UnFreqUnGN	WtUnGN
Size 2	0.7917	0.6667	**0.875**	**0.875**	0.7917	**0.875**	0.8125	0.8125
Size 3	0.7778	0.7778	0.8333	0.8333	0.7778	**0.8889**	0.8333	0.8333
Size 4	**1**	0.875	**1**	**1**	**1**	0.875	**1**	**1**
Size 5	0.85714	0.85714	0.85714	0.85714	**1**	0.71429	0.71429	0.71429
Size 6	**1**	**1**	**1**	**1**	**1**	0.8	**1**	**1**
Size 7	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**

Table 1: Success@3 with Profile Merging

	Baseline	IntGN	WtIntGN	FreqIntGN	UnionGN	FreqUnGN	UnFreqUnGN	WtUnGN
Size 2	0.77083	0.6875	**0.8125**	**0.8125**	0.625	**0.8125**	0.52083	0.52083
Size 3	0.8333	0.7778	**0.8889**	0.8333	0.66667	0.8333	0.7222	0.7222
Size 4	0.9375	0.875	0.9375	**1**	0.8125	0.9375	0.875	0.8125
Size 5	**1**	**1**	**1**	**1**	0.7143	**1**	0.85714	0.85714
Size 6	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Size 7	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**	**0.75**	0.5

Table 2: Success@3 with Recommendation Aggregation

	Baseline	IntGN	WtIntGN	FreqIntGN	UnionGN	FreqUnGN	UnFreqUnGN	WtUnGN
PM	0.83673	0.75510	**0.88776**	**0.88776**	0.84694	0.85714	0.84694	0.84694
RA	0.83673	0.77551	**0.86735**	**0.86735**	0.69388	0.85714	0.67347	0.65306

Table 3: Success@3 with Profile Merging (PM) and Recommendation Aggregation (RA) for 100 groups

References

Ardissono, L.; Goy, A.; Petrone, G.; Segnan, M.; and Torasso, P. 2003. Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices. *Applied AI*.

Baltrunas, L.; Makcinskas, T.; and Ricci, F. 2010. Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the 4th ACM RecSys Conference*.

Berkovsky, S., and Freyne, J. 2010. Group-based recipe recommendations: analysis of data aggregation strategies. In *Proceedings of the 4th ACM RecSys Conference*.

Breese, J. S.; Heckerman, D.; and Kadie, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proc of the 14th Conference on Uncertainty in AI*, 43–52.

Chen, Y.-L.; Cheng, L.-C.; and Chuang, C.-N. 2008. A group recommendation system with consideration of interactions among group members. *Expert Syst. Appl.* 34.

De Pessemier, T.; Dooms, S.; and Martens, L. 2012. Design and evaluation of a group recommender system. In *Proc of the Sixth ACM RecSys Conference*.

Gartrell, M.; Xing, X.; Lv, Q.; Beach, A.; Han, R.; Mishra, S.; and Seada, K. 2010. Enhancing group recommendation by incorporating social relationship interactions. In *Proc of the 16th ACM International Conference on Supporting Group Work*.

Goren-Bar, D., and Glinansky, O. 2004. Fit-recommending tv programs to family members. *Computers & Graphics* 28.

Herlocker, J.; Konstan, J. A.; and Riedl, J. 2002. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Inf. Retr.* 5.

Jameson, A., and Smyth, B. 2007. Recommendation to groups. In Brusilovsky, P.; Kobsa, A.; and Nejdl, W., eds., *The adaptive web*.

Jameson, A. 2004. More than the sum of its members: challenges for group recommender systems. In *Proceedings of the working conference on Advanced visual interfaces*.

Masthoff, J. 2004. Group modeling selecting a sequence of television items to suit a group of viewers. *User Modeling and User-Adapted Interaction* 14.

McCarthy, K.; Salamó, M.; Coyle, L.; McGinty, L.; Smyth, B.; and Nixon, P. 2006. Cats: A synchronous approach to collaborative group recommendation.

McCarthy, J. F. 2002. Pocket restaurantfinder: A situated recommender system for groups. 1–10.

Najjar, N. A., and Wilson, D. C. 2014. Differential neighborhood selection in memory-based group recommender systems. In *The Twenty-Seventh International Flairs Conference*.

O'Connor, M.; Cosley, D.; Konstan, J. A.; and Riedl, J. 2001. Polylens: a recommender system for groups of users. In *Proc of the 7th conference on European Conference on Computer Supported Cooperative Work*.

Quijano-Sanchez, L.; Recio-Garcia, J. A.; Diaz-Agudo, B.; and Jimenez-Diaz, G. Social factors in group recommender systems. *ACM Trans. Intell. Syst. Technol.* 4(1):8:1–8:30.

Quijano-Sánchez, L.; Bridge, D.; Díaz-Agudo, B.; and Recio-García, J. 2012. Case-based aggregation of preferences for group recommenders. In *Proc of the 20th ICCBR*.

Quijano-Sánchez, L.; Recio-García, J. A.; and Díaz-Agudo, B. 2011. Group recommendation methods for social network environments. *3rd Workshop on Recommender Systems and the Social Web* 5th ACM RecSys Conference.

Recio-Garcia, J. A.; Jimenez-Diaz, G.; Sanchez-Ruiz, A. A.; and Diaz-Agudo, B. 2009. Personality aware recommendations to groups. In *Proceedings of the 3rd ACM RecSys Conference*.

Sarwar, B.; Karypis, G.; Konstan, J.; and Riedl, J. 2000. Analysis of recommendation algorithms for e-commerce. In *Proc of the 2nd ACM Conference on Electronic Commerce*

Senot, C.; Kostadinov, D.; Bouzid, M.; Picault, J.; Aghasaryan, A.; and Bernier, C. 2010. Analysis of strategies for building group profiles. In *User Modeling, Adaptation, and Personalization*, volume 6075 of *Lecture Notes in Computer Science*.