

Performance Analysis of Group Recommendation Systems in TV Domains

Noo-ri Kim and Jee-Hyong Lee

Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon, Korea



Abstract

Although researchers have proposed various recommendation systems, most recommendation approaches are for single users and there are only a small number of recommendation approaches for groups. However, TV programs or movies are most often viewed by groups rather than by single users. Most recommendation approaches for groups assume that single users' profiles are known and that group profiles consist of the single users' profiles. However, because it is difficult to obtain group profiles, researchers have only used synthetic or limited datasets. In this paper, we report on various group recommendation approaches to a real large-scale dataset in a TV domain, and evaluate the various group recommendation approaches. In addition, we provide some guidelines for group recommendation systems, focusing on home group users in a TV domain.

Keywords: Group recommendation system, Group modeling, Home group, TV domain

1. Introduction

In various domains, such as movies, music, books, and TV, the amount of content is increasing rapidly. Although this vast content provides a wide range of choices to users, they often have difficulty with choosing or finding what they want to consume due precisely to this huge amount of content. For this reason, researchers have proposed various recommendation systems for providing convenience in choice [1-7] and these recommendation systems are successful in various services, such as movies, music, and e-commerce [8]. Some services are for single users and some for group users. For example, e-commerce services are usually for single users, and movies or TV programs are for single or group users. However, most recommendation approaches are designed for single users and there are only a small number of approaches for group users.

General group recommendation systems need to model the preference of groups for the recommendation. There are two main approaches. The first is the group profile-based approach. If there is a group of users A , B , and C , the system collects the list of items consumed by the group and recommends items based on the profile [9]. Because it is difficult to obtain profiles of groups, the other approach is based on the profiles of individual group members. It is a consensus function-based approach [10-12] that creates the pseudo-preferences of a group by merging the preference of each member in the group using consensus functions.

Because group profile-based approaches are very similar to recommendations for single users, most approaches for single users can be directly applied and do not have many variants. On the other hand, many consensus function-based approaches have been proposed. McCarthy

Received: Dec. 19, 2014
Revised : Dec. 31, 2014
Accepted: Mar. 17, 2015

Correspondence to: Jee-Hyong Lee
(john@skku.edu)
©The Korean Institute of Intelligent Systems

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

and Anagnost [8] proposed a group preference arbitration system. It merged group member preference into a preference and recommended pieces of music that the group would like to listen to. Boratto and Carta [10] proposed how to make synthetic virtual group users by using clustering and then applied several consensus functions to a movie domain. Baltrunas et al. [11] proposed a group recommendation system using rank aggregation. In order to build preferences of group users, it used ranks instead of ratings. This method was applied to a movie domain [11]. Gorla et al. [13] proposed a probabilistic group recommendation via information matching. It made a group user preference model from each member's preference, based on probability. In addition, it applied to a movie domain.

Researchers evaluate and verify group recommendation approaches, however, with synthetic or limited datasets [8-16] because real large-scale datasets of group profiles are rare. Regarding design group recommendation systems, we need to consider group and domain characteristics simultaneously. Group user characteristics may change, depending on domain characteristics [12]. For example, we could ignore minor opinions in a group on casual domains, such as movies or TV programs.

In addition, to provide recommendation in many domains, we need to consider making user profiles use general recommendation methods. The general recommendation methods use the users' profiles, such as rating. In many domains, however, it is difficult to collect the user rating information because of lack of interaction or additional cost. Therefore, we need to consider basing the user profiles on consumed histories.

In this paper, we make users' profiles from users' consumed histories and analyze the effectiveness of group recommendation systems on real large-scale datasets regarding TV domain. In addition, we provide some guides for group recommendation systems, focusing on home group users in TV domains. We have organized the remainder of this paper as follows. Section 2 overviews methods for group modeling. Section 3 describes real-world datasets and pre-processing. Section 4 presents experimental results and Section 5 provides conclusions.

2. Group Modeling

To suggest items to a single user, most recommendation systems adopt a simple approach based on user profiles, such as ratings on items. For example, if a target user has high ratings on the items A , B , and C , the recommendation system suggests similar items to A , B , and C , or items popular to users who like A , B and C .

Unlike the recommendation system for single users, the group recommendation system needs consideration of the group profile and the group members' profile together.

There are two major approaches for group recommendations: the group profile-based approach and the consensus function-based approach. Each approach uses only one type of profiles, group profiles or group member profiles. The following paragraphs provide details on this approach.

2.1 Group Profile-Based Approach

The group user approach makes group profiles from group histories only. In other words, this approach treats a group as a single user. For example, there is a group user g_1 that is composed of u_1 , u_2 , and u_3 . If this group watches the TV contents c_1 , c_2 , c_3 , c_4 , c_5 , and c_6 and rates 3, 4, 2, 5, 1, and 2 together, these ratings are threaded to a group profile, such as $\langle 3, 4, 2, 5, 1, 2 \rangle$. This approach makes the group profile for the group recommendation system. However, if group histories are insufficient to use or cannot be collected, this approach cannot be used.

2.2 Consensus Function-Based Approach

In the case where group profiles are hard to obtain but group members' profiles are available, the consensus function approach can be applicable. These approaches make a group profile from group members' profiles by using consensus functions. The consensus functions merge group members' profiles into a pseudo-group profile and the profile is used to recommend. There are various consensus function strategies, such as least misery, most pleasure, and average. These consensus functions imitate decision making processes in the group [9]. The pseudo-group profiles that are made by each consensus function have difference characteristics, leading to different performances of the group recommendation system. Let us assume that there are group members' profiles as follows:

$$\begin{aligned} \mathbf{u}_1 &= \langle 3, 2, 2, 4, 3, 2 \rangle \\ \mathbf{u}_2 &= \langle 1, 5, 3, 4, 2, 1 \rangle \\ \mathbf{u}_3 &= \langle 2, 3, 3, 5, 2, 3 \rangle \end{aligned} \quad (1)$$

2.2.1 Least misery strategy

The least misery strategy (*Min*) makes a pseudo-group profile by selecting a minimum rating on each item from the group

members' profiles. Therefore, group g_1 is made as follows:

$$Min : g'_1 = \langle 1, 2, 2, 4, 2, 1 \rangle \quad (2)$$

The recommendation system could avoid suggesting TV programs that are similar to c_1 and c_6 , because g_1 's profile shows the lowest ratings on c_1 and c_6 . This strategy leads the recommendation system to avoid dissatisfaction for group members. However, this strategy could ignore satisfaction from other members.

2.2.2 Most pleasure strategy

The most pleasure strategy (*Max*) makes a pseudo-group profile by selecting a maximum rating on each item from the group members' profiles. The following vector shows g_1 's profile:

$$Max : g'_1 = \langle 3, 5, 3, 5, 3, 3 \rangle \quad (3)$$

The recommendation system could suggest TV content, such as TV programs similar to c_2 and c_4 , because g_1 's profile shows the highest ratings on c_2 and c_4 . This strategy supposes that the most pleasure user determines the group users' opinions. However, this strategy could ignore dissatisfaction from other members.

2.2.3 Average strategy

The average strategy (*Avg*) makes a pseudo-group profile by selecting the average rating on each item from group members' profiles. The following vector shows g_1 's profile:

$$Avg : g'_1 = \langle 2, 3.3, 2.7, 4.3, 2.6, 2 \rangle \quad (4)$$

The recommendation system could suggest TV content, such as items similar to c_4 , because g_1 's preference model shows the highest rating on c_4 . This strategy supposes that all group members have the same influence to determine the group's opinion. However, if group members have dissimilar preferences, this strategy is not suitable.

2.3 Consensus Function with Group Profile-Based Approach

In the case where both group profiles and group members' profiles are available, the consensus function with group profile based approaches can be applicable. The approach makes pseudo-group profiles from group profiles and group members'

profiles. This approach is an extension of the previously mentioned consensus function-based approach. Let us assume that there are a group profile and group members' profiles as follows:

$$\begin{aligned} g_1 &= \langle 3, 4, 2, 5, 1, 2 \rangle \\ u_1 &= \langle 3, 2, 2, 4, 3, 2 \rangle \\ u_2 &= \langle 1, 5, 3, 4, 2, 1 \rangle \\ u_3 &= \langle 2, 3, 3, 5, 2, 3 \rangle \end{aligned} \quad (5)$$

Then, g_1 's pseudo-group profiles are as follows:

$$\begin{aligned} Min/G : g'_1 &= \langle 1, 2, 2, 4, 1, 1 \rangle \\ Max/G : g'_1 &= \langle 3, 5, 3, 5, 3, 3 \rangle \\ Avg/G : g'_1 &= \langle 2.3, 3.5, 2.5, 4.5, 2, 2 \rangle \end{aligned} \quad (6)$$

This approach considers group preferences and group members' preferences simultaneously. However, this approach could be suitable when we know the agreement process among the group members and select an appropriate consensus function.

In order to make group profiles, we should consider when we apply the collaborative filtering to predict unknown ratings [12]. To make group profiles, we can apply the collaborative filtering before or after applying the consensus function-based approach and the consensus function with group profile-based approach. If a collaborative filtering is applied first, recommendation lists for each group members are generated and then the recommendation list for the group is generated by a consensus based approach. Therefore, the recommendation list for the group may reflect more each group member's preference. If a consensus function-based approach is applied first, the pseudo-group profiles are made from the group members' profiles by a consensus based approach, and then the recommendation list for the group is generated by applying a collaborative filtering to the pseudo-group profile. Therefore, this approach may reflect more of the group preference, or the consensus between group members.

3. Dataset

In order to analyze the performance of the group recommendation systems, we use the TNmS dataset. TNmS is the company that collects users' watching histories and provides program rating information. To collect the users' watching histories, TNmS uses the devices (peoplometer) that are set on the panel homes. The TNmS dataset consists of users' watching histories from 1,000 homes in Korea between 1/1/2013 and 6/30/2013. The number of total personal users is 3,318, with total TV con-

Table 1. Dataset information

	Single user	Group	Total
No. of history	1,078,254	655,577	1,733,831
No. of user	3,195	5,664	8,859
Average no. of group size	-	2.63	2.63
Average no. of history size	337.48	115.74	195.71
No. of group	2	-	3,278
on group size	3	-	1,742
	4	-	535
	5	-	93
	6	-	14
	7	-	2
Total	-	5,664	5,664

tent at 107,570. This TV content is on four public channels (KBS1, KBS2, MBC, and SBS). If a user watches TV content for less than 15 minutes, it is regarded as an invalid view and ignored.

Table 1 shows information of the dataset. In Table 1, groups have fewer watching histories than single users. Furthermore, about 50% of groups have fewer than 25 histories.

In order to apply the previously mentioned approaches, ratings on the TV program should be made from watching histories. We use the following equation.

$$r_{u,UP_n} = \ln(1 + cnt_{u,UP_n}) \tag{7}$$

In Eq. (7), r_{u,UP_n} is a rating on the unique TV program UP_n by a user u . For example, if a user watches TV programs in a unique TV program repeatedly, the rating on the unique TV programs is higher. The number of TV programs and unique TV programs are 21,480 and 1,958, respectively. In other words, each user has ratings vector that consists of 1,958 dimensions.

4. Experiment

4.1 Experiment Design

We experiment with five types of group recommendation systems. The first is the group profile-based approach (*GP*). The second is the collaborative filtering-consensus approaches (*CF-Con*), which applies collaborative filtering to individual members and makes a consensus of the individual group members. In *CF-Con*, there are three approaches with different consensus

strategies: *CF-Min*, *CF-Max*, and *CF-Avg*. The third is the collaborative filtering-consensus with group profile approach (*CF-Con/G*), which applies collaborative filtering to the individual members and applies a consensus function to the individual members including the group profile. Also, there are three approaches with different consensus strategies: *CF-Min/G*, *CF-Max/G*, and *CF-Avg/G*. The fourth is the consensus-collaborative filtering approach (*Con-CF*), which makes a consensus of the individual members and applies collaborative filtering. There are three approaches with different consensus strategies: *Min-CF*, *Max-CF*, and *Avg-CF*. The last is the consensus with group profile-collaborative filtering approach (*Con/G-CF*), which applies a consensus function to the individual members, including the group profile, and applies collaborative filtering. There are three approaches: *Min/G-CF*, *Max/G-CF*, and *Avg/G-CF*.

We use the user-based collaborative filtering to predict ratings. In collaborative filtering, we use the Pearson correlation coefficient to calculate similarity between users. It is as follows:

$$sim(\mathbf{u}_1, \mathbf{u}_2) = \frac{\sum_i (r_{u_1,i} - \bar{r}_{u_1})(r_{u_2,i} - \bar{r}_{u_2})}{\sqrt{\sum_i (r_{u_1,i} - \bar{r}_{u_1})^2 (r_{u_2,i} - \bar{r}_{u_2})^2}} \tag{8}$$

In Eq. (8), $r_{u_1,i}$ is a rating of TV program i by user u_1 and \bar{r}_{u_1} is an average of ratings by user u_1 . The value of similarity is from -1 to 1. If user u_1 and u_2 have more similar ratings, the similarity is closer to 1. In order to predict the unknown ratings, we used the adjusted weighted aggregation. It is as follows:

$$\tilde{r}_{u,i} = \bar{r}_u + \frac{\sum_{n \in KNN_u} (sim(u, n)(r_{n,i} - \bar{r}_n))}{\sum_{n \in KNN_u} sim(u, n)} \tag{9}$$

In Eq. (9), $\tilde{r}_{u,i}$ is the predicted rating on item i by user u . KNN_u is a set of users similar to the user u . Therefore, $\tilde{r}_{u,i}$ is predicted from the average of user u 's ratings and similar users' ratings. We set the neighborhood size to 50.

The dataset is divided into a training set and a testing set. The training set is first the 80% watching histories and the other 20% of watching histories were used as the testing set. The number of TV programs in the testing set is 133,383.

The recommendation process is as follows. If there is a group user with a test item, the recommendation system finds the existing ratings or predicts the unknown ratings on available TV programs by using the group user's profile or the pseudo-group profile. A TV program with the highest rating is then offered to the group user.

4.2 Metric

In order to analyze performances of the group recommendation systems, we use precision as a metric. It is defined as follows:

$$Precision = \frac{\# \text{ of correctly recommended TV programs}}{\# \text{ of recommended TV programs}} \tag{10}$$

The number of correctly recommended TV programs is the number of programs viewed by users in the recommended TV programs. If the value of the precision is higher, it means that the group recommendation system has a higher performance.

4.3 Experiment Result

In this section, we analyze the experimental results in two aspects. The first is the overall comparison of performance between each group recommendation system. The second is the performance depending on history sizes.

4.3.1 Overall comparison of performance

Table 2 shows the performance of various group recommendation approaches. First, *GP* shows a better performance than traditional consensus function-based approaches: *CF-Min*, *CF-Max*, *CF-Avg*, *Min-CF*, *Max-CF*, and *Avg-CF*. It implies that group user preferences built from group user profiles are more valuable than those built by merging individual group member profiles. However, most *CF-Con/G* approaches, *CF-Max/G*, *CF-Avg/G*, *Max/G-CF*, and *Avg/G-CF*, show better performances than *GP*. That is, the best performances were achieved by using both the preference of groups and the preferences of individual group members, implying that the preferences of individual group members are valuable even though the preferences of groups are available. We cannot find significant differences between *CF-Con* approaches and *Con-CF* approaches. The order of collaborative filtering and consensus function may not make a big difference in recommendation. However, the strategies for consensus show different performances. Max and Avg strategies show a better performance than the Min strategy, and thus are more suitable for home group users in TV domains.

4.3.2 Comparison of performance depending on history size

Figures 1-3 show performances of various group recommendation systems depending on the history sizes of the groups. We do not compare with the experiment results of *Con-CF*

Table 2. Overall performance comparison

Method		Precision
GP		0.319
CF-Con	CF-Min	0.202
	CF-Max	0.281
	CF-Avg	0.285
CF-Con/G	CF-Min/G	0.200
	CF-Max/G	0.362
	CF-Avg/G	0.365
Con-CF	Min-CF	0.217
	Max-CF	0.284
	Avg-CF	0.280
Con/G-CF	Min/G-CF	0.209
	Max/G-CF	3.660
	Avg/G-CF	0.357

GP, group profile-based; CF-Con, collaborative filtering-consensus; Con-CF, consensus-collaborative filtering.

approaches, because their performances are about the same as those of *CF-Con* approaches. The horizontal axis indicates the group history sizes and the vertical axis indicates the precision.

First, Figure 1 shows the performance of *GP* and *CF-Con* approaches. The performance of *GP* increases along with the group history size. It shows that the more histories a group has, the better performance the recommendation system has. In the *CF-Con* approaches, it is difficult to find a strong relation between the performances and history sizes, because they do not directly use group histories.

When a group has more than 300 histories, the performance of *GP* is better than that of *CF-Con* approaches. When a group has fewer than 300 histories, *CF-Con* approaches with Max and Avg strategies show better performances than *GP*. Especially when a group has fewer than 25 histories, the performance of *CF-Max* and *CF-Avg* are almost twice of that of *GP*. Therefore, if a group has fewer than 300 histories, *CF-Con* approaches may be more suitable to the group recommendation systems. In *CF-Con* approaches, Min strategy shows the worst performance. It means that most home group users choose TV programs that give more pleasure to group members.

Second, Figure 2 shows the performance of *GP* and *CF-Con/G* approaches. *CF-Max/G* and *CF-Avg/G* show better performances than *GP* for any history sizes, unlike *CF-Con* approaches. In other words, using group preferences is helpful to improve the performance of *CF-Con* approaches. Because *CF-Con/G* approaches use group preferences, their performance increases as the history size increases.

Last, Figure 3 shows the comparison of the *bests*, that is, the

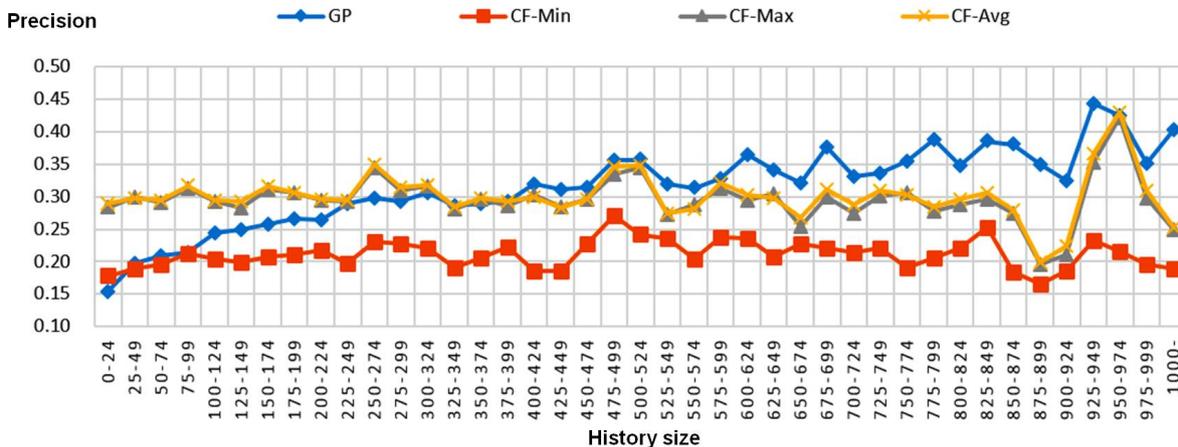


Figure 1. Performance comparison between GP and CF-Con approaches. GP, group profile; CF-Con, collaborative filtering-consensus.

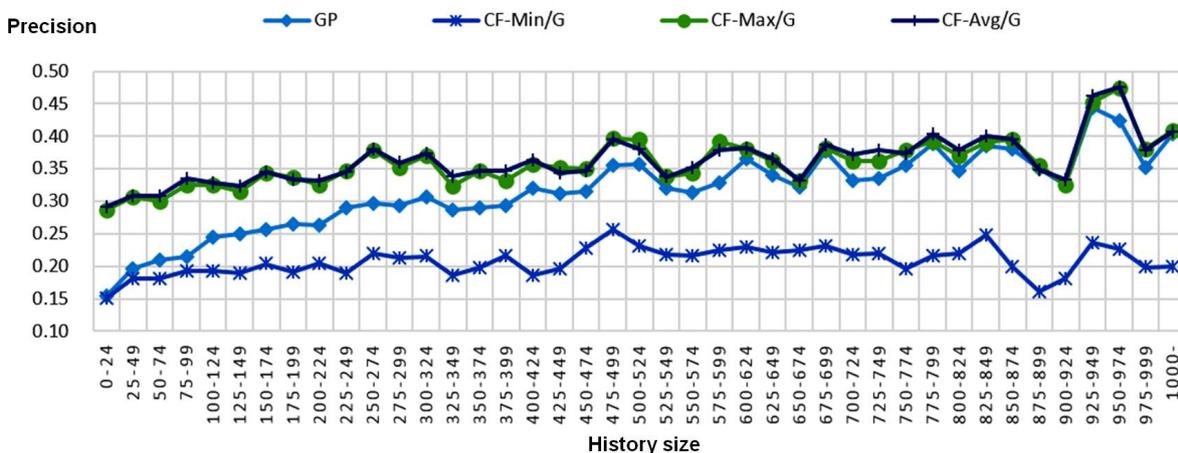


Figure 2. Performance comparison between GP and CF-Con/G approaches. GP, group profile; CF-Con/G, collaborative filtering-consensus with group profile.

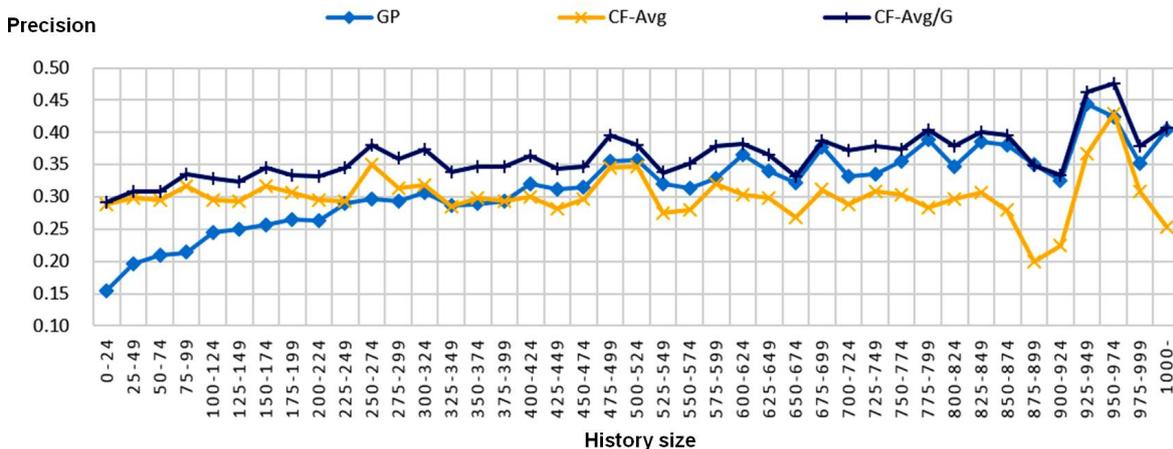


Figure 3. Performance comparison between GP, CF-Avg, and CF-Avg/G approaches. GP, group profile; CF, collaborative filtering.

comparison of *GP*, *CF-Avg*, and *CF-Avg/G*. *CF-Avg* and *CF-Avg/G* show the best performance in *CF-Con* and *CF-Con/G* approaches, respectively. For all the histories, *CF-Avg/G* shows the best performance. The gap between *CF-Avg* and *CF-Avg/G* increases more as the history size increases. In other words, if a group has more histories, *CF-Con/G* approaches can provide better performances.

These results tell us which approach is better depending on available information. If the system cannot use group profiles, *CF-Avg* is better. If the system can use group profiles, however, *CF-Avg/G* is better. In summary, for group recommendation system design, we should consider availability of group profiles.

5. Conclusions

In this study, we evaluated various group recommendation approaches with a real-world dataset, and provided some guidelines for group recommendation systems, focusing on home group users in TV domains.

We evaluated the five types of approaches: *GP*, *CF-Con*, *CF-Con/G*, *Con-CF*, and *Con-CF/G*. The experimental results showed characteristics of each approach.

The performance of the *GP* increases along with the group history size and when a group has more than 300 histories, it is better than the performance of the consensus function-based approaches. In the consensus function with a *GP*, *Max/G* and *Avg/G* show the best performance for all the histories. The order of collaborative filtering and consensus function may not make a big difference on home group users in TV domains. If a recommendation system cannot use group profiles, *Max* and *Avg* are more suitable than other approaches. If the recommendation system can use group profiles or a group has more than sufficient history for recommendation, however, *Max/G* and *Avg/G* are more suitable.

For a better recommendation, we may use profiles of groups with preferences that are similar to those of target groups. In future work, we will extend group recommendation approaches with similar group profiles and more various consensus functions.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

Acknowledgments

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2014M3C4A7030503) and the ICT R&D program of MSIP/IITP (10041244, Smart TV 2.0 Software Platform).

References

- [1] J. Bobadilla, F. Serradilla, and J. Bernal, "A new collaborative filtering metric that improves the behavior of recommender systems," *Knowledge-Based Systems*, vol. 23, no. 6, pp. 520-528, 2010. <http://dx.doi.org/10.1016/j.knosys.2010.03.009>
- [2] H. Shin, M. Lee, and E. Y. Kim, "Personalized digital TV content recommendation with integration of user behavior profiling and multimodal content rating," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 3, pp. 1417-1423, 2009. <http://dx.doi.org/10.1109/TCE.2009.5278008>
- [3] J. Kim, K. Kim, K. H. You, and J. H. Lee, "An approach for music recommendation using content-based analysis and collaborative filtering," *Information*, vol. 15, no. 5, pp. 1985-1996, 2012.
- [4] Y. Chung, H. W. Jung, J. Kim, and J. H. Lee, "Personalized expert-based recommender system: training C-SVM for personalized expert identification," in *Machine Learning and Data Mining in Pattern Recognition*, P. Perner, Ed. Berlin: Springer-Verlag, 2013, pp. 434-441. http://dx.doi.org/10.1007/978-3-642-39712-7_33
- [5] D. M. Kim, K. S. Kim, K. H. Park, J. H. Lee, and K. M. Lee, "A music recommendation system with a dynamic k-means clustering algorithm," in *Proceedings of the 6th International Conference on Machine Learning and Applications (ICMLA 2007)*, Cincinnati, OH, 2007, pp. 399-403. <http://dx.doi.org/10.1109/ICMLA.2007.97>
- [6] S. W. Bang, H. W. Jung, J. Kim, and J. H. Lee, "An auto playlist generation system with one seed song," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 10, no. 1, pp. 19-24, 2010.
- [7] S. I. Lee and S. Y. Lee, "A recommendation system using context-based collaborative filtering," *Journal of Korean*

Institute of Intelligent Systems, vol. 21, no. 2, pp. 224-229, 2011. <http://dx.doi.org/10.5391/IJKIIS.2011.21.2.224>

- [8] J. F. McCarthy and T. D. Anagnost, "MusicFX: an arbiter of group preferences for computer supported collaborative workouts," in *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work (CSCW)*, Seattle, WA, 1998, pp. 363-372. <http://dx.doi.org/10.1145/289444.289511>
- [9] Z. Yu, X. Zhou, Y. Hao, and J. Gu, "TV program recommendation for multiple viewers based on user profile merging," *User Modeling and User-Adapted Interaction*, vol. 16, no. 1, pp. 63-82, 2006. <http://dx.doi.org/10.1007/s11257-006-9005-6>
- [10] L. Boratto and S. Carta, "Modeling the preferences of a group of users detected by clustering: a group recommendation case-study," in *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS 2014)*, Thessaloniki, Greece, 2014, pp. 1-7. <http://dx.doi.org/10.1145/2611040.2611073>
- [11] L. Baltrunas, T. Makcinskas, and F. Ricci, "Group recommendations with rank aggregation and collaborative filtering," in *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys)*, Barcelona, Spain, 2010, pp. 119-126. <http://dx.doi.org/10.1145/1864708.1864733>
- [12] S. Pizzutilo, B. D. Carolis, G. Cozzolongo, and F. Ambruso, "Group modeling in a public space: methods, techniques, experiences," in *Proceedings of the 5th WSEAS International Conference on Applied Informatics and Communications*, Malta, 2005, pp. 175-180.
- [13] J. Gorla, N. Lathia, S. Robertson, and J. Wang, "Probabilistic group recommendation via information matching," in *Proceedings of the 22nd International Conference on World Wide Web*, Rio de Janeiro, Brazil, 2013, pp. 495-504.
- [14] R. Sotelo, Y. Blanco-Fernandez, M. Lopez-Nores, A. Gil-Solla, and J. J. Pazos-Arias, "TV program recommen-

ation for groups based on multidimensional TV-anytime classifications," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 1, pp. 248-256, 2009. <http://dx.doi.org/10.1109/TCE.2009.4814442>

- [15] M. O'Connor, D. Cosley, J. A. Konstan, and J. Riedl, "PolyLens: a recommender system for groups of users," in *Proceedings of the 7th European Conference on Computer Supported Cooperative Work (ECSCW 2001)*, Bonn, Germany, 2001, pp. 199-218. http://dx.doi.org/10.1007/0-306-48019-0_11
- [16] L. Quijano-Snchez, D. Bridge, B. Daz-Agudo, and J. Recio-Garca, "A case-based solution to the cold-start problem in group recommenders," in *Case-Based Reasoning Research and Development*, B. D. Agudo and I. Watson, Eds. Berlin: Springer-Verlag, 2012, pp. 342-356. http://dx.doi.org/10.1007/978-3-642-32986-9_26



Noo-ri Kim received his B.S. in computer engineering from Sungkyunkwan University, Suwon, Korea, in 2013. He is currently pursuing his M.S. in computer engineering at Sungkyunkwan University. His research interests include group recommendation system and machine learning.

E-mail: pd99@skku.edu



Jee-Hyong Lee received his B.S., M.S., and Ph.D. in computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Rep. of Korea, in 1993, 1995, and 1999, respectively. From 2000 to 2002, he was an international fellow at SRI International, USA. He joined Sungkyunkwan University, Suwon, Korea, as a faculty member in 2002. His research interests include fuzzy theory and application, intelligent systems, and machine learning.

E-mail: john@skku.edu