A Single-Scaled Hybrid Filtering Method for IPTV Program Recommendation

Kyusik Park, Jongmoo Choi, and Donghee Lee

Abstract— In this paper, a single-scaled hybrid filtering algorithm is proposed to recommend user preferred IPTV-VOD program. A proposed system is implemented with hybrid filtering method that can cooperatively complements the shortcomings of the content-based filtering and collaborative filtering. For a user program preference, a single-scaled measure is designed so that the recommendation performance between content-based filtering and collaborative filtering is easily compared and reflected to final hybrid filtering procedure. In order to provide a high quality of program recommendation, we use not only the user watching history, but also the user program preference and mid-subgenre program preference which are updated weekly as a user preference profile. System performance is evaluated with modified IPTV data from real 24-weeks cable TV watching data provided by Nilson Research Corp. in Korea and it shows quite comparative quality of recommendation.

Keywords— IPTV, Recommendation system, Hybrid filtering, Content-based filtering, Collaborative filtering, Program preference

I. Introduction

IPTV (Internet Protocol TV) is a next-generation media that can provide convergence service of broadcasting and communication using high speed internet. The main role of IPTV is to offer personalized program contents via VOD (Video on Demand) at anytime, anywhere. For this reason, a study of personalized program recommendation algorithm in IPTV environment has been broadly focused recently.

Program recommendation system automates the procedure to predict and recommend program contents for personal tastes based on a user watching history or the watching patterns of other neighborhoods with similar interest.

As a recommendation algorithm, there are two major approaches for general recommendation systems, i.e., the content-based filtering (CBF) approach and the collaborative filtering (CB) approach. In the content-based filtering, the user profiles are first formed by extracting features of items or programs which have been accessed in the past. Based on the user profiles, the system recommends only the items that are highly relevant to the user profiles by computing the similarities between items and the user profiles. Examples of such systems are NewsWeeder [1], Infofinders [2]. In this approach, the feature accuracy for representation of item contents is key issue which dominates the effectiveness of the recommendation. However, content-based recommendation system only able to recommend the items or programs in which a user has indicated his/her interest in past. On the other hand, the collaborative approach computes the similarities between the user profiles. Users of similar profiles are grouped together to share the information in there profiles. The main goal of the collaborative approach is to make the recommendation among the users in the same group. Adopting the collaborative filtering approach, the system has a high possibility to recommend surprising items or programs by the nature of information sharing which cannot be achieved by the content-based filtering approach. But, it has well-known cold start problem. Examples of such system are Ringo [3] and Siteseer [4]. In [5], Wang, et al propose a system architecture of personalized recommendation using collaborative filtering based on web log and this paper also gives an improved k-means algorithm for clustering user transaction.

On the other hand, a hybrid filtering system combines two or more algorithm to get better performance with fewer drawbacks. There are many ways how different filtering system can be combined. Burke [6] identified 53 different possible hybrids and defined hybrid system into seven taxonomies – weighted, switching, mixed, feature combination, cascade, feature augmentation and meta-level. Representative examples of hybrid systems are Fab [7] to recommend web pages and P-Tango [8] to recommend news in an online newspaper. In [9], Charalampos propose a new way of combining neural networks and collaborative filtering, They uses a neural network to recognize implicit patterns between user profiles and items of interest which are then further enhanced by collaborative filtering to personalized suggestions. Ahn [10] presents a hybrid recommender system using a new heuristic similarity measure that focuses on improving performance under cold-start conditions of collaborative filtering. Xia, et al [11] developed an intelligent agent framework that integrates document collection, information retrieval and collaborative filtering recommendation.

In this paper, we propose a new IPTV-VOD program recommendation system based on a hybrid filtering algorithm. In comparison with previous works, the major contributions of this paper can be summarized as follows.

This research was supported by the National IT Industry Promotion Agency (NIPA) under the program of Software Engineering Technologies Development.

(1) As a recommendation algorithm, a mixed type of hybrid filtering that can cooperatively complements the shortcomings of the content-based filtering and collaborative filtering is implemented. For this purpose, a single-scaled measure that can compute user preference is designed to compare recommendation performance between content-based filtering and collaborative filtering, and feed this result into final hybrid filtering procedure.

(2) Not like the most previous mixed type of hybrid filtering works [12]-[14], that recommends a fixed rate of programs from content-based filtering and collaborative filtering, the proposed system does not care whether the recommendation comes from content-based filtering or collaborative filtering. It just simply compares the user preference scores using single-scaled measure to provide effective program recommendations. This kind of recommendation mechanism allows very flexible, stable and adaptive system architecture.

(3) As a user preference profile which is a main key for the success of recommendation algorithm, a user profile is adaptively constructed to reflect the most current user interest on a weekly basis and it gives a weight to consecutive watching for series program. It leads a better personalized recommendation.

(4) Since IPTV watching data has not been released to open public so far, real-24weeks cable TV watching data provided by Nilson Research Corp. in Korea is modified and reconstructed to fit IPTV environment and the proposed system is evaluated with this modified IPTV data.

This paper is organized as following. Section 2 describes overall proposed IPTV recommendation system. A single-scaled hybrid filtering algorithm is fully investigated in section 3. Experimental environments and corresponding system performance are analyzed in section 4. Finally, the paper ends with conclusion in section 5.

II. Proposed IPTV Program Recommendation System

Fig. 1 shows the main architecture for the personalized IPTV-VOD program recommendation agent in proposed system.

As seen in Fig.1, a program agent is mainly composed of Set-Top Box (STB) and the recommendation server. Set-Top Box in fig.1 plays a role of receiving IPTV-VOD program list from the server, providing a list of program recommendations to user through UI (User Interface), and also transmit user watching history to the server. On the other hand, the server consists of user model management engine, recommendation engine, watching history DB, user model DB, and broadcasting program DB. Watching history DB stores individual user watching history received from a Set-Top Box. User model management engine computes mid-subgenre preference and the program preference on a weekly basis and store these results into user model DB. On the other hand, broadcasting program DB stores a current IPTV-VOD programs being telecast which is also updated weekly. As a final recommendation, a hybrid filtering method based on user model DB, mid-subgenre preference and program preference is applied to IPTV-VOD program DB to recommend top-five programs ordered in final program preference.

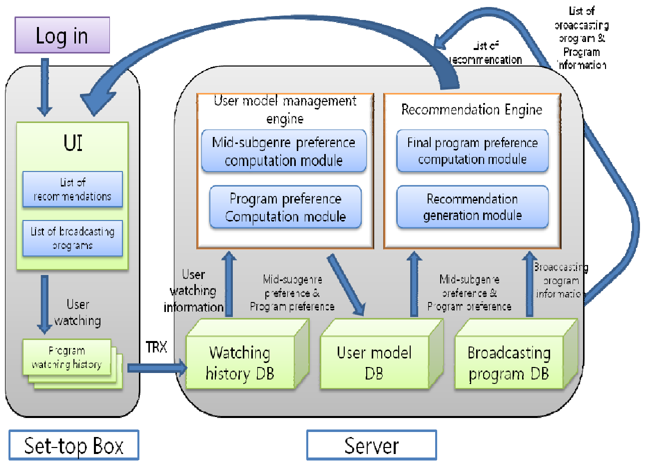

Fig.1 Proposed IPTV-VOD program recommendation agent.

A. *Broadcasting program DB*

Broadcasting program DB stores current IPTV-VOD program as a metadata which is composed of program ID, program name, broadcasting time, program genre, and mid-subgenres. In this paper, we categorized IPTV-VOD program into 13 genres and 60 mid-subgenres.

Table 1 Example of IPTV-VOD program metadata.

Program ID	Program name in Korean	Broad-casting time (min.)	Program genre	Program Mid-subgenre
3145	빵빵 그림책버스	25	20	2001
4235	치로와 친구들	15	20	2002
613	방귀대장 뿡뿡이	15	20	2002
134	주주클럽	50	65	6501
65614	유럽축구 핫! 골!	25	40	4001

B. *Watching history DB and User model DB*

A list of received IPTV-VOD programs from the server and the actual user watching information from the recommendation list are reorganized as a final watching history to be stored in watching history DB as in fig. 2. On the other hand, a user model DB stores program preference and mid-subgenre preference values which is weekly updated in user model management engine.

C. *User model management engine*

User model is a main key to recognize a user watching pattern of the IPTV-VOD programs and the model computes program preference and mid-subgenre preference based on watching history information as in fig.2. A user program preference is calculated using only the most recent two-weeks watching history information. In order to reflect the most current user watching history and to give a weight to consecutive watching for series program, a user program preference and mid-subgenre preference is updated weekly also.

[IPTV-VOD program information]

Program ID	Program name in Korean	Broadcasting time (min.)	Program genre	Mid-subgenre
3145	빵빵그림책버스	25	20	2001
4235	치로와친구들	15	20	2002
613	방귀대장뿡뿡이	15	20	2002
134	주주클럽	50	65	6501
65614	유럽축구핫!골!	25	40	4001

[User information]

User ID	Age	Academic background	Gender
103487801	5	8	1

[Watching history]

Program ID	Beginning time	Ending time	Watching period (in week)
3145	132200	133500	1
4235	133600	135100	1
134	200500	204500	1
613	210000	211100	1

[Watching history information]

User ID	Program ID	Large-subgenre	Mid-subgenre	Watching period (in week)
103487801	3145	20	2001	1
103487801	4235	20	2002	1
103487801	134	65	6501	1
103487801	613	20	2002	1

Fig.2 Example of setting up for final watching history information

1) Computation and update of intermediate program preference

Program preference for user u_i can be calculated by referring a recent two-weeks watching history. It is defined as a normalization of watching frequency p_j regarding program j, $j = 1,2,3,...,N$ where N is a total number of program, by maximum frequency $\max_j(p_j)$. In order to give a weight for watching of series program, we give a weight of $w_j = 1$ when a user watch a series program in two consecutive weeks, otherwise, it gets $w_j = 1/2$. Equation (1) shows program preference score (PPS) $PPS_j^{u_i}$ for user u_i to program j and it has values between 0 and 1.

$$PPS_j^{u_i} = \frac{p_j * w_j}{\max_j(p_j)}, \quad j = 1,2,3,...,N \tag{1}$$

For example, if a total number of program is eight in two-weeks watching history and if the watching frequency is given as in table 2, $\max_j(p_j)$ will be 14 and each program preference can be calculated as follows. For the case of program ID 1, preference score can be computed as $PPS_1^{u_i} = \frac{p_1 * w_1}{\max_j(p_j)} = \frac{14*1}{14} = 1$ and for program ID 2, this will be $PPS_2^{u_i} = \frac{p_2 * w_2}{\max_j(p_j)} = \frac{14*(1/2)}{14} = 0.142$.

Table 2 Example calculation of program preference for user u_i

Program ID	Watching frequency (p_j)	Weight (w_j)	Program preference ($PPS_j^{u_i}$)
1	14	1	1
2	4	1/2	0.142
3	6	1/2	0.214
4	3	1	0.214
5	7	1	0.5
6	8	1	0.571
7	2	1/2	0.071
8	6	1/2	0.428

In order to reflect current user interest, a user program preference is updated on a weekly basis by considering previous and current program preference simultaneously. In other words, updated program preference for user u_i to program j in m^{th} week is given as average of program preference of m^{th} week and $(m-1)^{th}$ week as in (2).

$$PPS_{update_j}^{u_i} = \frac{PPS_j^{u_i}(week_{m-1}) + PPS_j^{u_i}(week_m)}{2} \tag{2}$$

In (2), $PPS_j^{u_i}(week_{m-1}), PPS_j^{u_i}(week_m)$ are program preferences of m^{th} week and $(m-1)^{th}$ week for user u_i to program j and $PPS_{update_j}^{u_i}$ is updated program preference for m^{th} week that reflects program preference of previous week.

2) Computation and update of mid-subgenre preference

Mid-subgenre preference indicates a user interest to IPTV-VOD program subgenre and it is calculated and updated in every week. Mid-subgenre preference for a user u_i is defined as a normalization of watching frequency g_k regarding program subgenre k, $k = 1,2,3,...,L$ where L is a total number of program subgenre, by maximum frequency $\max_k(g_k)$. Equation (3) shows class preference score (CPS) $CPS_k^{u_i}$ for user u_i to program subgenre k and it has values between 0 and 1.

$$CPS_k^{u_i} = \frac{g_k}{\max_k(g_k)}, \quad k = 1,2,3,....,L \tag{3}$$

Mid-subgenre preference is updated every week by taking an average of mid-subgenre preference of m^{th} week and $(m-1)^{th}$ week as in (4).

$$CPS_{update_j}^{u_i} = \frac{CPS_j^{u_i}(week_{m-1}) + CPS_j^{u_i}(week_m)}{2} \tag{4}$$

III. Proposed Recommendation Algorithms for IPTV-VOD Program

The proposed program recommendation algorithm uses a hybrid filtering method that can cooperatively complement the content-based filtering and collaborative filtering. At first, final program preference in each recommendation filtering is calculated as a multiplication of a program preference and mid-subgenre preference as described in section 2.3. Each filtering method then recommends five programs according to the final program preference score. Finally, the proposed hybrid filtering compares the preference score of recommended programs from each filtering and pick out top-ranked five programs.

A. Calculation of final program preference

1) Content-based filtering (CBF)

Content-based filtering recommends the programs in which the user has indicated his/her interest in previous watching history. In this paper, content-based filtering calculates final program preference as a multiplication of program preference and mid-subgenre preference, and it recommends top-five ranked programs ordered in final preference score. Final preference for user u_i to program j in m^{th} week can be calculated as in (5) and it is updated every week.

$$CBF_j^{u_i}(week_m) = PPS_{update_j}^{u_i}(week_m) * CPS_{update_k}^{u_i}(week_m) \tag{5}$$

In (5), $CBF_j^{u_i}(week_m)$ is a final program preference for program j in m^{th} week calculated by content-based filtering. $PPS_{update_j}^{u_i}(week_m)$ is updated program preference for program j in m^{th} week, and $CPS_{update_k}^{u_i}(week_m)$ is updated mid-subgenre preference for program j in mid-subgenre k .

2) Collaborative filtering (CF)

Collaborative filtering recommends programs by referring the other users with similar interest. Program recommendation by collaborative filtering consists of two steps. First step is to group the neighbor users who have similar watching patterns. Then the algorithm calculates final program preference by taking average program preference of neighbor user which is calculated in the same way of (5). Final program recommendation is to pick up top-five ranked programs.

Equation (6) calculates user similarity (US) to group the neighbor users with similar watching patterns and it forms five users in one group by lining up in order of US.

$$US(u_i, u_j) = 1 - \sum_{l=1}^{G} (|\, GR_l^{u_i} - GR_l^{u_j} \,|), \quad i \neq j \tag{6}$$

$$GR_l^{u_i} = \frac{CPS_{update}^{u_i}(week_m)}{\sum_{n=0}^{60} CPS_{update}^{u_i}(week_m)}$$

Here, u_i is a user who will be recommended, u_j is neighbor user, G is a total number of mid-subgenres which is 60 in this paper. $GR_l^{u_i}$ is a mid-subgenre preference ratio for user u_i , so the user similarity (US) will be close to 1 as the mid-subgenre preference of a user u_i and neighbor user u_j is similar. User group is also updated weekly.

Final program preference by collaborative filtering is calculated as an average preferences in remaining four users in a group. Here the preference of each user is also calculated as a multiplication of program preference and mid-subgenre preference as in (5). Equation (7) shows a final program preference for user u_i to program j which is updated in every week.

$$CF_j^{u_i}(week_m) = \frac{\sum_{n=0(n\neq i)}^{3} PPS_{update_j}^{u_n}(week_m) * CPS_{update_k}^{u_n}(week_m) * US_{week_m}(u_i, u_j)}{4} \tag{7}$$

In (7), $CF_j^{u_i}(week_m)$ is a final program preference for user u_i to program j in m^{th} week by collaborative filtering, and $PPS_{update_j}^{u_n}(week_m)$ is updated program preference score of program j for remaining four users excluding a user u_i . And $CPS_{update_k}^{u_n}(week_m)$ is a updated mid-subgenre preference of program j to mid-subgenre k in m^{th} week. $US_{week_m}(u_i, u_j)$ is a user similarity between user u_i and u_j in m^{th} week.

B. IPTV program recommendation procedure of the proposed hybrid filtering method

IPTV-VOD recommendation engine in this paper utilizes a user program preference and mid-subgenre preference to recommend IPTV program. This engine cooperatively complements the content-based filtering and collaborative filtering, so it can not only recommend various kinds of IPTV programs, but also possible to recommend for the new IPTV program.

Content-based filtering calculates final user preference as a multiplication of program preference and mid-subgenre preference and it recommends top-five ranked programs. On the other hand, collaborative filtering first group neighbor users who have similar watching patterns and then choose top-five ranked programs by taking average of final program preferences of remaining four users in a group. As seen in (5) and (7), the final program preference for both content-based filtering and collaborative filtering is scaled between 0 and 1, and so, we can compare recommendation performance between these two filtering methods using a single-scaled measure and reflect this result into final hybrid filtering procedure. In other words, the proposed hybrid filtering algorithm compares ten program preferences submitted from each filtering method, then it ultimately recommends top-five ranked IPTV programs to the users.

Figure 3 shows the flowchart of the proposed hybrid recommendation algorithm and table 3 shows nomenclature for the mathematical symbols used in this paper.

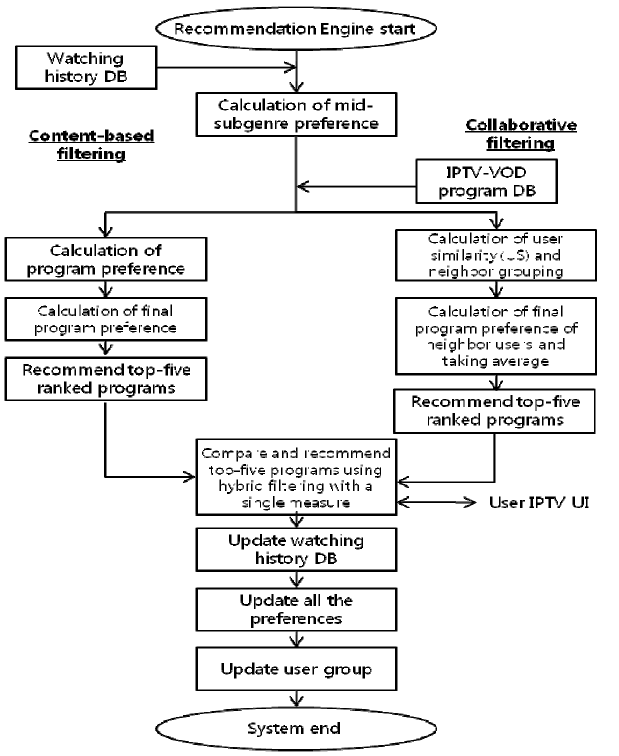

Fig. 3 Flowchart of the proposed recommendation engine

Table 3 Nomenclature for the mathematical symbols

Symbols	Meaning
$PPS_j^{u_i}$	Program preference score (PPS) for user u_i to program j
$PPS_{update_j}^{u_i}$	Updated PPS for user u_i to program j in m^{th} week
$CPS_k^{u_i}$	Class preference score (CPS) for user u_i to program subgenre k
$CPS_{update_j}^{u_i}$	Updated CPS for user u_i to program subgenre k
$CBF_j^{u_i}(week_m)$	Final program preference for program j in m^{th} week by content-based filtering
$US(u_i, u_j)$	User similarity between user u_i and user u_j for $i \neq j$
$GR_l^{u_i}$	Mid-subgenre preference ratio for user u_i to mid-subgenre l
$CF_j^{u_i}(week_m)$	Final program preference for user u_i to program j in m^{th} week by collaborative filtering

IV. Experimental Results

A. Experimental environment

Since IPTV watching data is not opened to public yet, we used cable TV watching data provided by Nilson Research Corp. in Korea. This data was modified and reconstructed to fit IPTV environment to evaluate the system performance. Nilson Research data used in this paper is real 24-weeks watching data from Jan. 1st of 2008 to June 14th of 2008 for terrestrial broadcast and cable TV.

After modification of the Nilson Research data, the first four-week data is used as a training data, and the rest of data is used as a test data for the experiment. Training data is used to calculate the initial program preference and mid-subgenre preference in content-based filtering and also it is used to form initial neighbor group in collaborative filtering. Test data is used to evaluate the system performance.

Followings are major modifications made to Nilson Research data to fit for IPTV-VOD environment. Firstly, to match the characteristics of the IPTV-VOD programs, one-time broadcasting programs such as news, weather, traffic, etc. was removed from the Nilson Research data and only the rest of programs were used to build IPTV-VOD program DB. Secondly, the program genre classification was made by following the program classification system made by KT, one of the well-known IPTV service providers in Korea and as a result, we have a total of 13 program genre with 60 mid-subgenres. Finally, to get accurate statistics of user program preference and mid-subgenre preference, we only consider the programs that a user watches more than 10% of the program to build user model DB.

B. Experimental results and analysis

There are many ways to evaluate the recommendation system and such are Hit ratio, Ranking point average, Percentage of contents, Precision, and Recall, etc. In this paper, we used Precision as a performance evaluation measure and it describes percentage of the actual user watching based on five recommended programs made by the proposed IPTV-VOD recommendation engine. Equation (8) shows the mathematical formula for the Precision measure. Here, $N(=5)$ is a total number of recommended programs by the system and N_k is the number of programs a user actually watched from the recommendation list.

$$Precision = \frac{N_k}{N} \quad (8)$$

The proposed system was evaluated in following five aspects. Firstly, the recommendation performance of content-based filtering and collaborative filtering is independently compared. Secondly, as in the most previous research works, the hybrid system with a fixed ratio of recommendations from the content-based filtering and collaborative filtering was compared. Thirdly, the proposed hybrid recommendation system was evaluated and compared in terms of Precision measure. Fourthly, the system performance with different computation method of final program preference was investigated. Finally, the proposed system was analyzed with different number of program recommendations.

1) Individual recommendation performance

Figure 4 compares individual performance of each filtering method that recommends five programs separately. As seen from the figure, the Precision performance of these two filtering method is very similar to 0.68 in average. But, the system performance is gradually dropped as the week goes, and this is because the number of IPTV-VOD programs to be recommended is accumulated gradually also. And the worst system performance near week 20 is due to the lack or only a small number of users watching history.

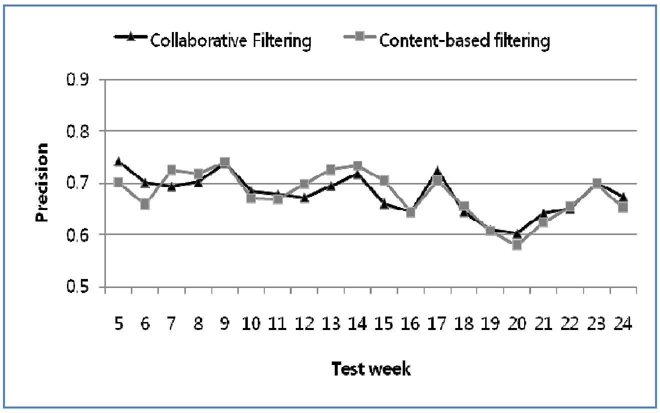

Fig. 4 Performance comparison of content-based filtering and collaborative filtering

2) Performance comparison of hybrid system with a fixed ratio of recommendation

Figure 5 and table 4 compares the system performance when the number of recommendations from the content-based filtering and collaborative filtering has various fixed ratios 4:1, 3:2, 2:3, and 1:4. For example, 4:1 fixed ratio of recommendation means that the system take four recommendations from the content-based filtering and take only one recommendation from the collaborative filtering.

Table 4 System performance with various fixed ratios of recommendation from content-based filtering and collaborative filtering

Recommendation ratio	4:1	3:2	2:3	1:3
Performance	0.705	0.690	0.686	0.681

As seen on the figure and table, the recommendation performance is best when the recommendation ratio is 4:1. It means that the content-based filtering has more influence to overall system performance.

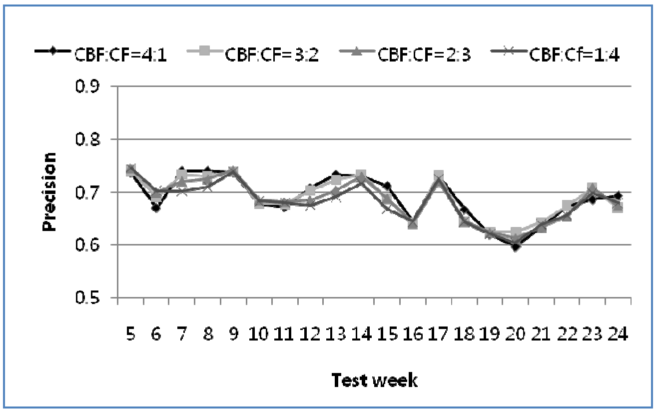

Fig. 5 Performance comparison of fixed ratio hybrid system

As seen on the figure and table, the recommendation performance is best when the recommendation ratio is 4:1. It means that the content-based filtering has more influence to overall system performance.

On the other hand, we notice that the hybrid system in figure 5 shows a little bit of improved performance than the one with individual filtering method in section 4.2.1. It actually explains the efficiency of the hybrid system.

3) Performance comparison of the proposed hybrid recommendation system

Not like the previous hybrid system in section 4.2.2 which recommends a fixed ratio of programs from content-based filtering and collaborative filtering, the proposed system does not care whether the recommendation comes from content-based filtering or collaborative filtering. It just simply compares the final program preference score using a single-scaled measure to provide top-five ranked program recommendations.

Figure 6 compares system performance of the collaborative filtering in section 4.2.1, hybrid system with fixed ratio of 4:1 in section 4.2.2 and the proposed hybrid filtering system.

From the figure, the proposed system outperforms all other methods with an average precision of 0.72 which is 4.3% improvement over the collaborative filtering in section 4.2.1 and 1.7% improvement over the system in section 4.2.2. Such a performance improvement comes from the flexible program recommendation ability with a single-scaled program preference scale.

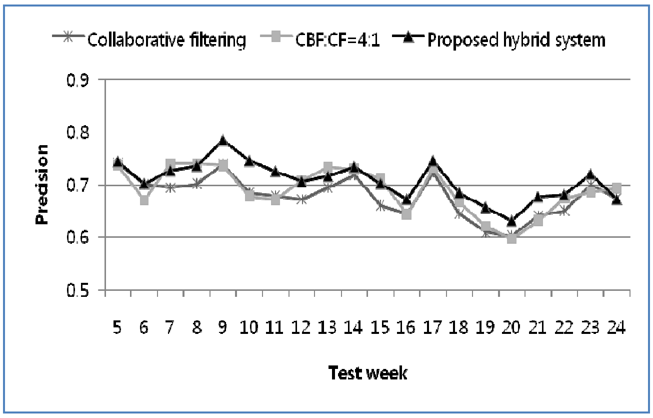

Fig. 6 Performance comparison of the proposed hybrid recommendation system

4) System analysis with different computation method of final program preference

Figure 7 compares system performance when we used different computation method for the final program preference– one as a multiplication of program preference and mid-subgenre preference as we did before and the other as added program genre preference.

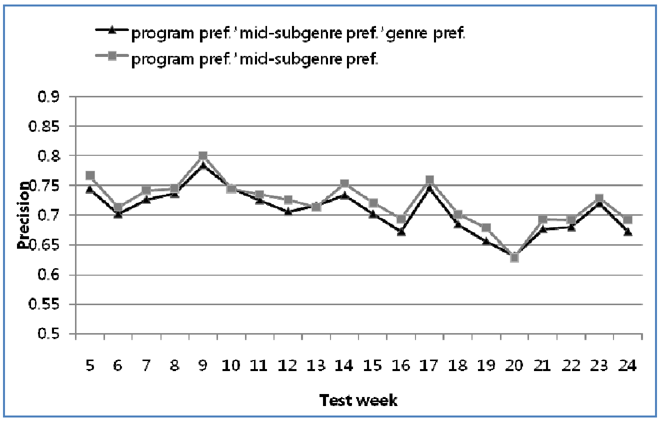

Fig. 7 Performance comparison with different computation method of final program preference

As seen from the figure, the system without program genre preference shows 0.722 of average precision while the one with program genre preference shows 0.709. This tells that the program genre preference has very little effect to the overall system performance.

5) System performance with different number of program recommendations

Figure 8 compares system performance when we vary the number of program recommendations to 5, 10 and 15. From the figure, we see that the system performance is gradually dropped from 0.72 to 0.556 as the number of program recommendation is increased from 5 to 15. For this reason, all the experiments in this paper were performed with an assumption of five recommendations.

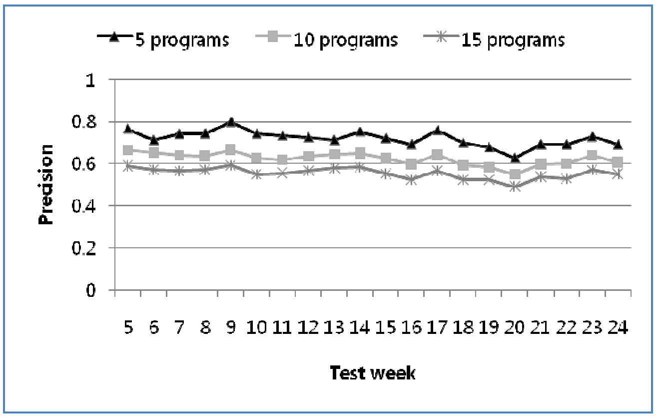

Fig. 8 System performance with different number of program recommendations

V. Conclusions

This paper proposes a new IPTV-VOD program recommendation algorithm with flexible hybrid filtering method. A proposed system works complementally to cooperate the content-based filtering and collaborative filtering. A single-scaled measure is designed to calculate final user program preference and it is used to compare the program preference of each filtering method. Then this comparison result is reflected to final hybrid filtering procedure for IPTV-VOD program recommendation. In order to provide high quality of program recommendation, not only the user watching history, but also the user program preference and mid-subgenre program preference are updated in every week and they are reflected to a final user program preference profile. System performance is evaluated with modified IPTV-VOD data provided by Nilson Research Corp. in Korea and it shows quite comparative quality of recommendation compared to previous research results. The proposed system can provide very flexible and stable program recommendations. For the future research, a further elaboration of the proposed system will be investigated for the adaptive hybrid filtering structure based on the precision performance of each component filtering method. In other words, the contribution of each filtering method can be evaluated at each final program preference score, and this contribution can be reflected to next program recommendation.

References

[1] Lang K., 'Newsweeder: Learning to Filter Netnews', *In Proceedings of International Conference on Machine Learning*, pp. 331–-339, 1955

[2] Krulwich, B. and Burkey, C., 'Learning User Information Interests through Extraction of Semantically Significant Phrases',. *In Proceedings of AAAI Spring Symposium on Machine Learning in Information Access,* 1996

[3] Sharadanand, U. and Maes, P., 'Social Information Filtering: Algorithms for Automating 'Word of Mouth'. *In Proceedings of CHI''95 Conference on Human Factors in Computing Systems*, pp. 210–-217, 1995

[4] Rucker J. and Polanco, M.J., Personalized Navigation *for theWeb. Communications of the ACM*, 40, 73–-89, 1977.

[5] Wang, et al, 'Study on Personalized Recommendation Based on Collaborative Filtering,' *Proceedings of the 3rd WSEAS International Conference on COMPUTER ENGINEERING and APPLICATIONS* (CEA'09).

[6] Burke, R., ‘Hybrid Recommender Systems: Survey and Experiments’, *User Modeling and User-adapted Interaction*, [S.1], v. 12, n.4, pp. 331-370, Nov. 2002
[7] BALABANOVIC, M, SHOHAM, Y, Fab, ‘Content-Based, Collaborative Recommendation’, *Communications of the ACM*, New York, v.40, n. 3, p. 66-72, Mar. 1997
[8] Claypool, M. et al, ‘Combining Content-Based and Collaborative Filters in an Online Newspaper’, *In ACM SIGIR WORKSHOP ON RECOMMENDER SYSTEMS*, Berkley-Cam Proceedings, 1999.
[9] Charalampos, et al, ‘A Recommender System Framework combining Neural Networks & Collaborative Filtering‘, P*roceedings of the 5th WSEAS Int. Conf. on Instrumentation, Measurement, Circuits and Systems*, Hangzhou, China, April 16-18, 2006
[10] Ahn, H.J.,’ A Hybrid Collaborative Filtering Recommender System Using a New Similarity Measure**’**, *Proceedings of the 6th WSEAS International Conference on Applied Computer Science*, Hangzhou, China, April 15-17, 2007
[11] Xia. Z, ‘An Agent Framework for Recommendation**’**, *Proceedings of the 6th WSEAS Int. Conference on TELECOMMUNICATIONS and INFORMATICS*, Dallas, Texas, USA, March 22-24, 2007
[12] Christina Christakou, Andreas Stafylopatis, ‘A Hybrid Movie Recommender System Based on Neural Networks’, *Intelligent Systems Design and Applications*, pp. 500-505, 2005.
[13] Kazuyoshi Yoshii, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno, **‘**Hybrid collaborative and content-based music recommendation using probabilistic model with latent user preferences,’ *ISMIR 2006, 7th International Conference on Music Information Retrieval*, pp 296-301, Victoria, Canada, 8-12 October 2006
[14] Qing Li and Byeong Man Kim, ‘Clustering approach for hybrid recommender system,’ 2003 IEEE / WIC International Conference on Web Intelligence, pp 33-38, Halifax, Canada, 13-17 October 2003

Kyusik Park received B.S, M.S, and PhD degrees in 1986, 1988, and 1994, respectively, all from the Department of Electrical Engineering of NYU-Polytechnic, Brooklyn, NY USA. In 1994, he joined the Semiconductor Division of Samsung Electronics as a staff engineer. He is currently a professor with the Division of Computer Science in Dankook University, Korea. His research interests are digital signal, speech, and audio processing, and digital communication

Jongmoo Choi received the BS degree in oceanography from Seoul National University, Korea, in 1993 and the MS and PhD degrees in computer engineering from Seoul National University in 1995 and 2001, respectively. He is an associate professor in the school of computer science and engineering, Dankook University, Korea. Previously, He was a senior engineer at Ubiquix Company, Korea. He held a visiting faculty position at the University of California, Santa Cruz from 2005 to 2006. His research interests include file systems, mobile memory, and virtualization technology and embedded systems.

Donghee Lee received the MS and PhD degrees in computer engineering, both from Seoul National University, Seoul, Korea, in 1991 and 1998, respectively. He is currently an associate professor in the School of Computer Science, University of Seoul, Korea. He was a senior engineer at Samsung Electronics Company, Korea, in 1998, and an assistant professor in the School of Telecommunication and Computer Engineering, Cheju National University, Korea, from 1999 to 2001. His research interests include operating systems, I/O systems, flash memory, and embedded real-time systems.