I-038

Personalized Audiovisual-Program Recommendation by Improved Watch-Flow Algorithm

Pakpahan Christina Natalia† Wataru Kameyama‡

1. Introduction

In this information overload era, the importance of recommendation system has been dramatically increased in suggesting people what items they might like. By exploiting the rating given by users, recommendation can be generated. But it requires large number of users' ratings to make it viable. On the other hand, people are generally lazy and rarely rate items, even give partial information about a certain item.

To solve the above mentioned problems, some recommendations are generated by observing user behavior, called implicit rating. In this research, we focus on audiovisual programs and investigate user's watching behavior. Then, we take the user's watching duration into account as user's preference. We also group users into two clusters based on their similarity in terms of watching duration.

2. Recommendation Model

The proposed model is built based on the watch-flow algorithm [1]. The difference can be seen in Fig. 1.

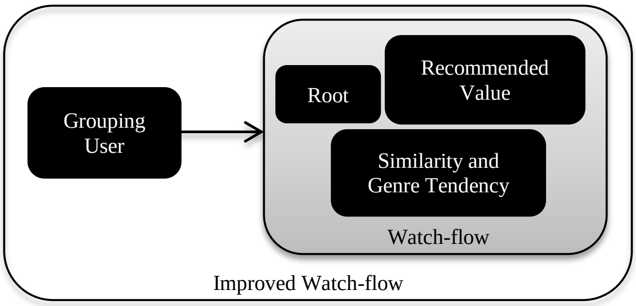

Fig. 1 The Difference of Watch-flow Algorithm and Improved Watch-flow Algorithm

3. Watch-flow Algorithm

There are three main parameters as illustrated in Fig. 1.

(1)Root

Root is the last-watched program by each user. Each user has different root.

(2)Similarity and Genre Tendency

Similarity is divided into user-based similarity and metadata-based similarity. The user-based similarity $ubs_{root,m}$ defines the likelihood that how much time other users spent to watch program *m* after the target users' root. It is shown in Eq. (1), where α is a parameter set to 0.5 according to [2].

$$ubs_{root,m} = \frac{sum_{root,m}}{selected_{root} * (selected_m)^{\alpha}} \tag{1}$$

Metadata-based similarity defines the similarity details of the target user's *root* and *m*. It is shown in Eq. (2), where $intersection_{root,m}$ is the number of the intersection of the metadata elements. Meanwhile the number of metadata elements itself in this case is 6, which is actor1, actor2, actor3, director, production and writer.

$$mbs_{root,m} = \frac{intersection_{root,m}}{\#metadata\_elements} \tag{2}$$

The genre tendency p_m is derived from all users' trending value of genre selection or overall genre distribution g_m, and from the trending value of the target user's genre selection or genre distribution g_s, which is shown in Eq. (3).

$$p_m = g_m + g_s \tag{3}$$

Eq. (4) is used for calculating g_s, where $sel_{root,m}$ is the number of the occurrence of the genre associated with program *m* selected after the target user's root in other users' histories, while $sel_{root,other}$ is the number of the occurrence of other genres selected after the target user's root in other users' histories.

$$g_s = \frac{sel_{root,m}}{sel_{root,other}} \tag{4}$$

(3)Recommended Value

Recommending value is defined as the likelihood estimation that a user will watch program *m* based on the last watched program called *root*. Recommending value r_m is calculated from the genre tendency (p_m) and the similarity between a program m and root ($similarity_{root,m}$) as shown in Eq. (5).

$$r_m = p_m * similarity_{root,m} \tag{5}$$

By applying descending sort to the recommending list, we can get the top N recommendation list for each user.

4. Improved Watch-flow Algorithm

The proposed model works similar with the watch-flow algorithm. However grouping users has been added to the original one as shown in Fig.1, where users are grouped based on their similarities. In this experiment, we use the watching time duration as the parameter of the similarity. By the average of all users' watching durations, users are divided into two groups;

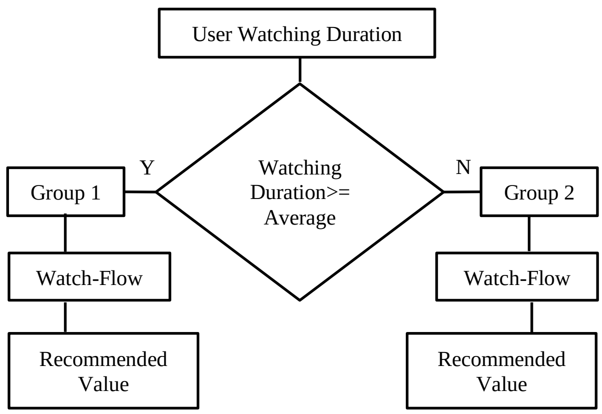

Fig. 2 Illustration of Grouping User in Improved Watch-flow

† GITS, Waseda University
‡ School of Fundamental Sc. and Eng., Waseda University

users who have total watching duration below and above the average. By applying watch-flow algorithm to each group, the recommended value is generated, as shown in Fig. 2.

Another modification is also done in terms of user-based similarity. The proposed algorithm takes user-watching duration into account, which shows the user preferences toward a program. This data is used as an implicit rating to generate the program recommendation. The modified one is shown in Eq. (6).

$$ubs_{root,m} = \frac{sum_{root,m}}{duration_{root} * (duration_m)^{\alpha}} \tag{6}$$

5. Experimental Dataset

In this experiment, we use dataset provided by WOWOW Inc, with the detail as follows.

- Number of programs: 10132
- Number of users: 1544
- Metadata elements of each program: actors, director, production, and writer.
- Total genres: 50

This dataset provides user-logging data information such as the starting time of a Webpage browsing featuring a specific audiovisual program and metadata for every program in the dataset.

We take the specific Webpage watching by a user as a real audiovisual program watching by users.

6. Evaluating Recommendation

The proposed model is evaluated by two metrics; precision that shows the accuracy of the recommendation, and coverage that shows the diversity of the recommendation. The used equations are described as follows.

$$precision = \frac{\#number of\ hits}{K} \tag{7}$$

Where K is the number of generated recommendations of audiovisual programs.

$$coverage = \frac{frequency of\left({feature_{rec}}/{feature_{user}} \right)}{frequency of\ feature_{user}} \tag{8}$$

Where $feature_{rec}$ is the new feature that reflects the relative component of features in the recommending list in user's watching historic features, and $feature_{user}$ is user's watching history feature. The feature in the coverage is decided by the metadata components.

By applying the metrics to both the watch-flow and the improved watch-flow algorithms, the result is shown in Fig. 3 and Fig. 4. k is the number of program recommendations, for example $k=5$ means that the number of generated program recommendations is 5. In this experiment, we use k from 5 to 10.

In the case of the precision shown in Fig. 3, the proposed model works slightly better when k=10, about 2%. But it's also not really bad when k=5 (less than 1%).

In the case of the coverage shown in Fig. 4, the improved watch-flow algorithm gives better diversity of the result. It means that the recommended programs gives more surprise to users and can enlarge the scope of recommendations.

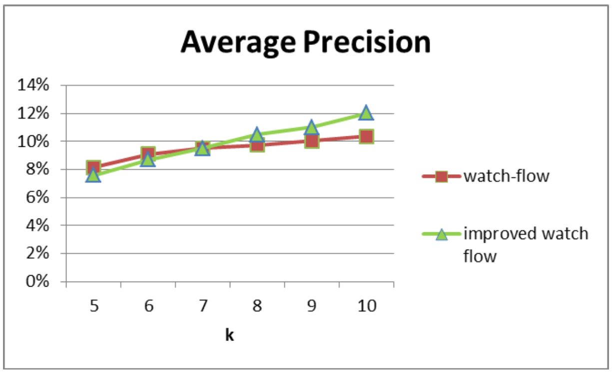

Fig. 3 The Precision Comparison of Watch-flow and Improved Watch-flow

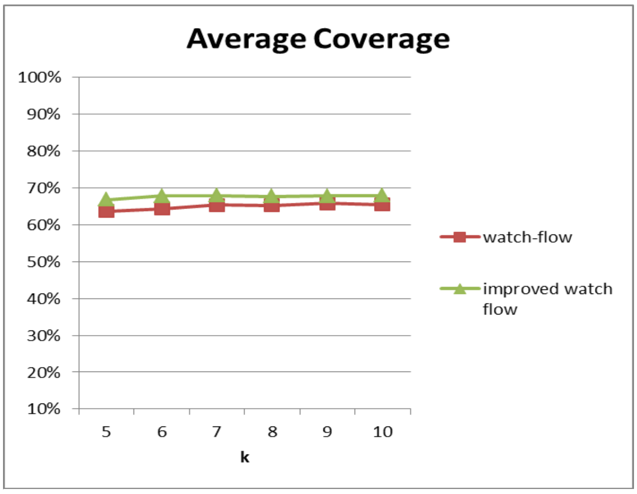

Fig. 4 The Coverage Comparison of Watch-flow and Improved Watch-flow

7. Conclusions and Future Work

Through this experiment, grouping users can be considered to improve performance in recommendation. Therefore, this makes the proposed method viable for future research and development.

For future works, we will investigate more how to group users such as by determining the optimum number of groups and applying similarity algorithm to create it.

Acknowledgement

We would like to thank WOWOW Inc. for providing us with the dataset.

References

[1] D. Q. Hung, P. Sriprasertsuk, W. Kameyama, K. Fukuda, "Utilizing Users' Watching Sequences and TV-programs' Metadata for Personalized TV-program Recommendation", IPSJ SIG Technical Report on AVM, 2012-AVM-77(11), 2012.

[2] G. Karypis: "Evaluation of Item-Based Top-N Recommendation Algorithms", in Proc. of the 10th International Conference on Information and Knowledge Management, pp.247-254 (2001).