Improving Personalized Recommendation in VSS Based on User's Preference

Huageng Li, Tomohiro Murata

***Abstract*— The popularity of Video Sharing Service is growing intense as the rapid development of Internet. The competition among Video Sharing Service is increasing. For avoiding competition, personalized recommendation system must to be applied. In this paper, a novel Video Recommendation algorithm based on extended collaborating filtering algorithm is proposed which provides personalized recommendation service. The system applies extended collaborative filtering algorithm by analyzing external factors which influence the user's personal preference. For external factors, user's personal attribute is given as one of the key factors to make user's preference different; Time and user's recent watching behavior are factors to make user's preference changing. Based on the theory of traditional collaborative filtering, we apply cosine similarity algorithm to formulize external factors for getting a more effective personalized recommendation system for Video Sharing Service.**

***Index Terms*— Video Sharing Service, collaborative filtering, time weight, personal attribute, video attribute**

I. Introduction

VIDEO Sharing Service (VSS) is the video website which provides sharing service. For instance: YouTube [1], Microsoft Box. VSS as the product of rapid development of Internet is growing intense. So that the competition is more and more extreme among VSS service provides. To keep users stay with their website, the improvement of user's loyalty becomes to be a serious problem for VSS provides. For satisfying user's personal requirements, the most effective method is to establish a personalized recommendation system.

The most important part of VSS is the recommendation system which plays an important role for improving user's loyalty. Recommendation system has been used in many websites for recommending variety products, such as: book, music and etc. There are large amounts of recommendation technologies have been developed in the past ten years [2]. Recently, the most common used algorithms are content-based recommendation algorithm and collaborative filtering algorithm [3] [4].

1) *Content-based recommendation algorithm* [4]

This algorithm assume that user's preference (or interest) not change at all, according to user's history data, summarizes the characters of items that user likes and predicts item to user by fitting characters.

2) *Collaborative filtering algorithm*

This algorithm works based on user's preference information to produce a recommend list for target user. It assumes that if users have similar or same rating information for one item, then they will also give the similar rating for other items. Collaborative filtering algorithm firstly searches a number of nearest neighbors for a target user and predicts target user's rating for a specific item on the basis of the rating information that nearest neighbor made, and then to generate recommend item list [5]. Usually, users who have the same age and sex will possibly have similar preference. However, personal's preference will change by time goes, so that the recommendation system will be needed to have a personalized and novel recommendation service.

The main idea that proposed by this paper is to give an extended collaborative filtering algorithm by analyzing and formulizing external factors which influence personal preference different and changing, after that, combining the formulized external factors with traditional collaborative filtering algorithm to make a higher effective personalized recommendation in VSS.

The paper is organized as follows: in Section II, the concept of collaborative filtering and related works on recommendation system is discussed. Section III introduces the extended collaborative filtering by combining the formulized external factors. The experimental result and evaluation are described in Section IV. In the last Section presents the conclusion of this paper.

II. Related Works

A. The theory of Collaborative Filtering

In the real world, people would like to ask their friends or the people who have the similar interest for unknown problems and events, and make their choice according with those people that have similar interest.

Collaborative Filtering (CF) algorithm simulates this process: find out the most similar neighbors for target user based on users' history behavior, moreover, to predict target user's preference for a specific item by the preference information that most similar neighbors made. It is a typical user-based recommendation algorithm.

B. The working procedure of Collaborative Filtering

Basically, collaborative filtering has three steps: creates user's rating matrix, searches for nearest neighbor and

Huageng Li is master student of Graduate School of Information, Production and Systems, WASEDA University, Fukuoka, Japan (e-mail: hgli@ruri.waseda.jp).

Tomohiro Murata is professor of Graduate School of Information, Production and Systems, WASEDA University, Fukuoka, Japan (e-mail: t-murata@waseda.jp).

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

generates prediction according to preference of the most similar neighbors.

TABLE I
RATING INFORMATION TABLE

User \ Item	$Item_1$	$Item_2$	...	$Item_n$
$User_1$	$R_{1,1}$	$R_{1,2}$	...	$R_{1,n}$
$User_2$	$R_{2,1}$	$R_{2,2}$	...	$R_{2.n}$
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
$User_m$	$R_{m,1}$	$R_{m,2}$	...	$R_{m,n}$

1) Create user's rating matrix

Assuming that a system has m users and n items, the information matrix could be expressed by $m \times n$. $R_{m,n}$ stands for the rating information that user m rated for item n. The rating matrix will be created after the rating information table (see TABLE I) is prepared. The rating matrix shows as follow:

$$\begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1n} \\ R_{21} & R_{22} & \cdots & R_{2n} \\ \vdots & \vdots & & \vdots \\ R_{m1} & R_{m2} & \cdots & R_{mn} \end{bmatrix} \quad (1)$$

2) *Search for nearest neighbor set*

This step mainly focuses on similarity calculation among users. Searches for neighbors who have the highest similarity value for target user and sets up a nearest neighbor list (Top-K: number for K depends on experiment result). The most common used approach for calculate the similarity is Pearson's Correlation Coefficient:

$$sim(m,n) = \frac{\sum_{C \in I_{m,n}} (R_{m,c} - \overline{R_m})(R_{n,c} - \overline{R_n})}{\sqrt{\sum_{C \in I_m} \left(R_{m,c} - \overline{R_m}\right)^2 * \sum_{C \in I_n} \left(R_{n,c} - \overline{R_n}\right)^2}} \quad (2)$$

Where $R_{m.c}$ is the rating of item c made by user m; $\overline{R_m}$ is the average rating of user m made for all rated item; $I_{m.n}$ is the item set both rated by user m and n.

3) *Generate prediction*

Recommends target user's rating for specific item based on the preference information of users in nearest neighbor list. The calculation as follows:

$$P_{m,c} = \overline{R_m} + \frac{\sum_{y=1}^{n} sim(m,y) \cdot (R_{y,c} - \overline{R_y})}{\sum_{y=1}^{n} |sim(m,y)|} \quad (3)$$

Where s*im(m,y)* is the similarity of user m and use y; $R_{y,c}$ is the rating user y made for item c; $\overline{R_y}$ is the average rating of user y made for all rated items.

III. EXTENDED COLLABORATIVE FILTERING

A. Time weight for watched video

Time is one of the most important contexts for predicting user's preference. Usually, users have different interests in different time interval and their preferences are successive changing along with time rolling around. That is one part that traditional collaborative filtering algorithm did not concern with before. For example: the parent may interest in information about introduction and admission of university, after the child went to university, the parent may not interest in university information any more.

For example (see TABLE II): there are two videos which watched by user recently, video A and B watched in different time, the user watched video A on yesterday and video B watched in last month, according to the time weight idea that proposed in this paper, the video watched in variety time has different time weight value and most recently watched has higher time weight. So that the time weigh of video A will be higher than video B: *TW(A) > TW(B)*.

TABLE II
IDEA OF TIME WEIGHT FOR VIDEO

Video	Watch time	Time weight
A	Yesterday	TW(A)
B	Last month	TW(B)

In order to improve the sensitivity of prediction for videos, and get the expected result of the implementation, here we propose a weight for videos watched based on time. Since the old or the history behavior has the ability to express user's preference, but not exactly correct, because user's preference is changing no matter stable change or temporary change, the most current videos watched have more meaning for express user's preference. Suppose that there is a user u, we have the data information about his time to enter the web site and the time watching videos, therefore, we can give the time weight for videos that the user watched:

$$TW(u,c) = (1-\alpha) + \alpha \frac{T_{uc}}{T_u} \quad (4)$$

$TW_{(u,c)}$ is the time weight for video c to user u. $T_{u,c}$ is time interval between the time when user u watch the first video in the website to the time when user u watched video c. T_u is time interval between the time when user u watch the first video in the website to the time when user u watched latest video. a is the increasing parameter ($0<\alpha<1$). Here, the value for parameter a is depends on experiment result.

After defined time weight function, use it to improve Pearson's Correlation Coefficient:

$$sim(m,n) = \frac{\sum_{C \in I_{m,n}} (TW(m,c) * R_{m,c} - \overline{R_m})(TW(n,c) * R_{n,c} - \overline{R_n})}{\sqrt{\sum_{C \in I_m} \left(TW(m,c) * R_{m,c} - \overline{R_m}\right)^2 * \sum_{C \in I_n} \left(TW(n,c) * R_{n,c} - \overline{R_n}\right)^2}} \quad (5)$$

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

B. Personal Attribute

Different people have different preference, although they may have similar parts. One of the biggest external factors that influence user's preference different defines as personal attribute in this paper, such as: experience, career, sex or age. After the similarity calculation, the output is a list of users who have the most similar preference with target user. Because personal attribute is one reason to make people's preference different, therefore we use cosine similarity algorithm to approve personal attribute idea, reselect the Top-K neighbor list: get the similarity between target user and users in nearest neighbor list (Top-K) by personal attribute.

$$Psim(m,n) = \frac{P_m \cdot P_n}{\|P_m\| \cdot \|P_n\|} \tag{6}$$

$$P_m = \{a_1, a_2, a_3, a_4, a_5 \cdots a_x\}$$
$$P_n = \{b_1, b_2, b_3, b_4, b_5 \cdots b_x\}$$

Here, P_m is Personal attributes set of user *m*. P_n is Personal attributes set of user *n*. *x* is the number of personal attributes. After the calculation above, a new neighbor list (Top-K') will be generated and users in the list have higher similarity according with target user's personal attributes.

C. Video Attribute

People's preference is not stable for all the time, it may change sometimes, and the change is temporary. For example: personal preference may change by a sad story, after reading a sad story, in the next few hours or even days, the mood will be changed by the story, therefore the preference will prefer to sad video more than funny video. For video sharing service, the information used to get the temporary change of user's preference is the video attribute which recently watched by target user.

After generate recommend video list (Top-N) which is the output of the third step of collaborative filtering algorithm, cosine similarity algorithm is used to reselect the video list, get similarity between videos watched by target user in recent period and videos in recommend item list by video attribute, such as: the language, category or the actor of video.

$$Msim(i,j) = \frac{M_i \cdot M_j}{\|M_i\| \cdot \|M_j\|} \tag{7}$$

$$M_i = \{a_1, a_2, \cdots a_z\}$$
$$M_j = \{b_1, b_2, \cdots b_z\}$$

M_i is video attributes set of video *i*. M_j is video attributes set of video *j*. *z* is number of video attributes. A new recommend video list (Top-N') will be generated after the calculation above.

IV. Experiment and Evaluation Result

For experiment, we use the data set from MovieLens recommender system which collected by the GroupLens Research Project at the University of Minnesota [6]. MovieLens project provides the most widely used common datasets in collaborative filtering research projects. The dataset that we used for the experiment is consists of 1,000,209 ratings for 3900 movies by 6040 users, each user has rated at least 50 movies. There are some statistics for example: movie category, user's age and other personal information that included in the dataset.

Before the evaluation is started, the rating data need to be randomly shuffled, and divided into five sub-datasets, each sub-dataset is divided into two parts, and one part is training set, taking 80% percent of total ratings, another part is testing set, taking 20% percent of total ratings in one sub-dataset. Firstly, the recommendation system with traditional CF algorithm and extended CF algorithm is built up using data of training set, after that, testing set is used to evaluate the recommendation result of traditional CF and extended CF.

A. Evaluation

In this paper, Mean Absolute Error (MAE) is used as the measurement for evaluation [7]. MAE is a statistical accuracy metrics to report prediction experiments and it is widely used in evaluation for CF research. It calculates the average absolute deviation of recommendations from their true user-specified values.

$$MAE = \frac{\sum_{i=1}^{N} |p_i - q_i|}{N} \tag{8}$$

Here *N* is the number of recommended times, and then MAE is defined as the average absolute difference between the *n* pairs. Assume that $p_1, p_2, p_3 \ldots\ldots p_n$ is the prediction of users' ratings and the corresponding actual rating data set of users is $q_1, q_2, q_3 \ldots, q_n$.

The smaller the deviation is the higher accuracy of prediction. The lower the MAE, the more accurate the predictions would be.

B. Value selection of parameter α

In this experiment, the evaluation of parameter α in proposed algorithm is necessary for the first step of experiment. From formula (4) we know that the value of α is from 0 to 1. Changing the value of α can adjust the effect of the time weight. In order to determine the suitable value of α for extended CF algorithm we proposed, at first we need to fix the neighbor number, because the neighbor number also can affect the recommendation effect. In evaluation, the sub-dataset 1 is used. We choose three neighbor numbers randomly from 1 to 100, for each number, we change α value from 0 to 1 to get the value of MAE. The result is shown in following Figure.

As shown in Fig.1, when the value of α is increased from 0.0 to 1.0, the value of MAE is decreased obviously, and get the lowest point when the α value is 0.3. After that, the value of MAE keeps increasing until the α value reaches 1.0. The result shows that, $\alpha = 0.3$ is a better value for the result of our extended CF algorithm. When the different neighbor number is chosen, almost same trend of MAE is shown in result, and, the change of MAE caused by parameter α is independent with change of neighbor number.

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

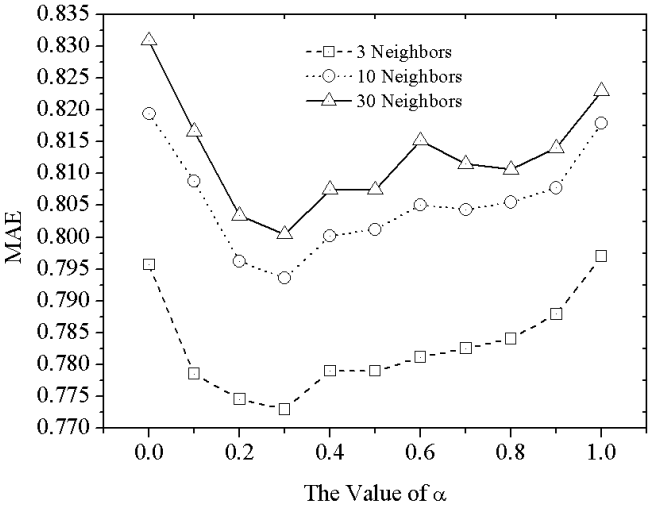

Fig. 1. Effect of parameter α in different value.

C. Selection of Neighbor Number

The nearest neighbor number (TOP-K) is also an important parameter which can affect the recommendation result. Usually, the more neighbor number is used, the more noise is brought in. Therefore we need to find out a better neighbor number to get stable and lower MAE for extended CF algorithm. Since it has been evaluated in previous section that when the α = 0.3, the extended CF can get lowest MAE with fixed neighbor number. In this section, the 0.3 is chosen for parameter α, and different neighbor number is evaluated.

The distribution of nearest neighbor number is investigated: only 17.28% users have more than 2000 neighbors among 6040 users. Therefore, we evaluated the MAE with 1 ~ 2000 neighbors to make sure that MAE trend. Here the value of α used in this evaluation is still 0.3.

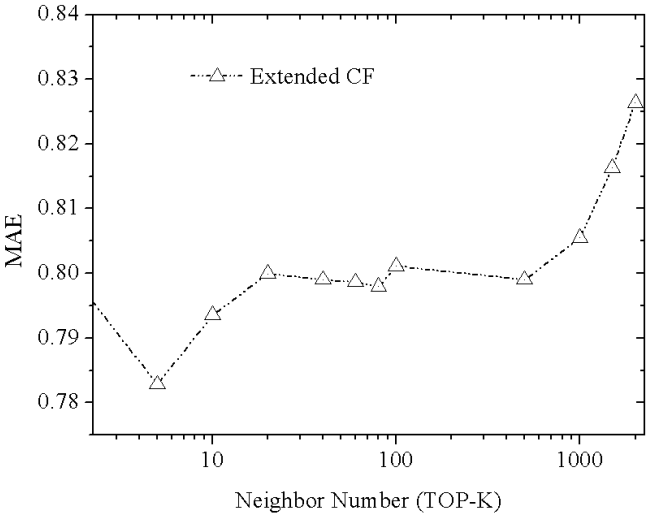

Fig. 2. Selection of suitable number for nearest neighbor, α is 0.3, Neighbor Number is from 1 to 2000.

The result can be seen in Fig.2. Based on experiments with different number of nearest neighbor, we can see that at MAE trend suddenly decreased from 1 to 3 neighbors, MAE reach the lowest point, since MAE trend with 1 to 10 neighbors is very sensitive, the lowest point of 3 neighbors will not be counted. From 20 neighbors to 90 neighbors MAE shows a stable trend between value 0.8 to 0.797, if the neighbor number is increased from 90 neighbors, the MAE keeps the increasing trend from 90 to 2000 neighbors. When the neighbor number exceeds the turn point, more noise is brought in, the MAE will get worse. Therefore, according to the experiment result, the number of TOP-K for nearest neighbor is defined as the range between 20 to 90 neighbors.

D. Comparison of result

In order to show the performance of proposed algorithm to the traditional collaborative filtering algorithm, we compare the result of both algorithms by MAE with suitable nearest neighbor number (20 to 90).

Figure 3 shows the comparison of recommendation result between extended CF algorithm and traditional CF algorithm. The proposed CF algorithm has lower MAE than the traditional CF algorithm in different point of neighbor number from 20 to 90.

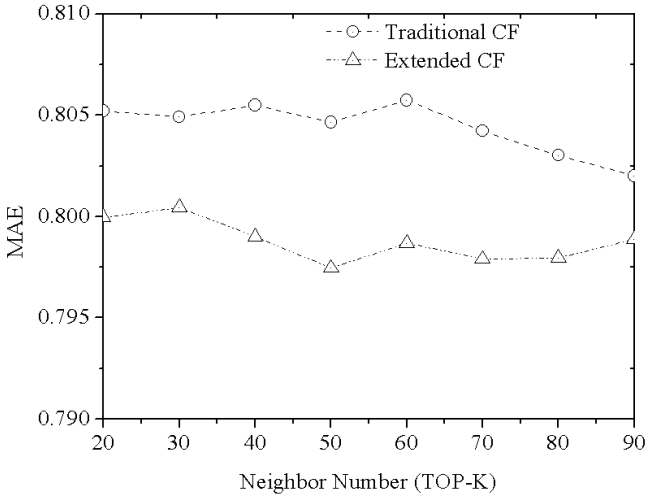

Fig. 3. Comparison between traditional CF and extended CF, α is 0.3, Neighbor Number is from 20 to 90.

By the figure which shown above, we can find out that the recommendation result of proposed extended CF algorithm is lower than the traditional CF algorithm in different nearest neighbor range, which proved the improvement on effectiveness of personalized recommendation that extended CF algorithm we proposed achieved.

V. Conclusion

Personal preference is the key factor for influencing personalized recommendation. As it defined in this paper, the time factor, user's personal attribute and video attribute is the main external factors which effect user's preference changing and different. This paper is proposed an extended collaborative filtering algorithm and combined the external factors for a better personalized recommendation. By the proof of experiment, the extended algorithm has better recommendation result than traditional collaborative filtering algorithm. The future research will focus on the scarcity for improving the recommendation accuracy.

Acknowledgment

We would like to give the gratitude to the GroupLens Research Project for the dataset they provided.

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

REFERENCES

[1] Shumeet Baluja, Rohan Seth, D. Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. Video suggestion and discovery for youtube: taking random walks through the view graph. In Proceeding of the 17th international conference on World Wide Web, WWW' 08, pages 895-904, New York, NY, USA, 2008. ACM.

[2] Linden G, Smith B, York J.Amazon.com recommendations: Item to Item collaborative filtering[J].IEEE Internet Computeing, 2003, 7(1):76-80.

[3] Marko Balabanovic and Yoav Shoham. Fab: content-based, collaborative recommendation. Commun. ACM, 40:66-72, March 1997.

[4] Hua qin-hua, Quyang wei-min, Fuzzy collaborative filtering with multiple agents, Journal of Shanghai University (English Edition), 2007, 11(3):290-295.

[5] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst., 22:5-53, January 2004.

[6] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommendation system s: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17 (6), 734-749.

[7] C. Anderson. The Long Tail. Random House Business, 2006.

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)