Exploiting Various Implicit Feedback for Collaborative Filtering

Byoungju Yang, Sangkeun Lee, Sungchan Park, and Sang-goo Lee
School of Computer Science and Engineering
Seoul National University
Seoul, Korea
{bjyang, liza183, baksalchan, sglee}@europa.snu.ac.kr

ABSTRACT

So far, many researchers have worked on recommender systems using users' implicit feedback, since it is difficult to collect explicit item preferences in most applications. Existing researches generally use a pseudo-rating matrix by adding up the number of item consumption; however, this naïve approach may not capture user preferences correctly in that many other important user activities are ignored. In this paper, we show that users' diverse implicit feedbacks can be significantly used to improve recommendation accuracy. We classify various users' behaviors (*e.g., search item, skip, add to playlist, etc.*) into positive or negative feedback groups and construct more accurate pseudo-rating matrix. Our preliminary experimental result shows significant potential of our approach. Also, we bring out a question to the previous approaches, aggregating item usage count into ratings.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Search and Retrieval—*Information filtering, Retrieval Models*

General Terms

Algorithms, Experimentation

Keywords

Implicit feedback, User behavior, Recommender system, Rating function

1. INTRODUCTION

Recommender systems have been widely studied due to its importance, especially in commercial area. Among various recommendation methods, collaborative filtering (CF) is acknowledged as one of the most successful approaches. CF methods aim to estimate unknown ratings using prior known ratings given by users. The most straightforward approach to collect ratings is to ask users to provide their preference explicitly, but it could be burdens to both users and systems, because the number of items is generally huge. To resolve this problem, recently, many of recent collaborative filtering approaches[3, 4, 5] take an approach to collect users' preference implicitly. In this case, a pseudo-rating matrix is constructed by analyzing users' item consumption log (*e.g. item usage log, purchase log*). They assume that the number of item consumption reflects users' preferences on items. For example, if a user likes a song, he or she will probably repeatedly listen to the song. This assumption seems reasonable in most cases, but there can be other important activities such as skipping songs, adding to playlist, deleting from the playlist and so on. These activities may be also very useful for inferring user's preferences. Most existing approaches do not consider various user activities and use a pseudo-rating matrix by adding up the number of item consumption. This may have a potential risk that it can misinterpret the user's preference. To solve the problem, we categorize various user activities into positive and negative feedback groups and propose a novel method for generating more accurate pseudo-rating matrix by weighted sum of factors. To test our approach, we have implemented a prototype online music listening service based on YouTube APIs and collected 315 users' activities from Dec. 5th 2011 to Feb. 1st 2012. We empirically tested our approach using the collected implicit feedback datasets with various weight settings. The experimental result shows that assigning different weights to diverse user activities for creating a pseudo-matrix could largely affect the recommendation accuracy, and the best cases outperformed the existing approach.

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

2. PROBLEM FORMULATION

According to the conventional usage [2, 4], we can define a set of items as $S=\{s_1, s_2, \ldots, s_l\}$ and a set of users as $U=\{u_1, u_2, \ldots, u_m\}$. Also, a set of user behavior log, which stores user's varied actions for items, is defined as $L=\{l_1, l_2, \ldots, l_n\}$. Each usage log l is composed of item s_i, user u_j, and user's behavior b_k. For instance, Table 1 shows an example of user behavior log. Also, the value of the unknown rating $p_{i,j}$ for item s_i, user u_j is computed by adding up similar user's weighted rating value on item s_i, $p_{i,j} = \bar{r}_{u_j} + \alpha \sum_k sim(u_j, u_k) \cdot (r_{k,j} - \bar{r}_{u_k})$ having α as a normalizing factor. The problem of exploiting implicit feedback in recommender system is defined as follows: Find ad

Table 1: Example of User Behavior Dataset

User	Music	Action
Erik	Poker Face	Search
Erik	Poker Face	Add to list
Sergey	Heal the world	Play
Sergey	Heal the world	Skip

Table 2: User Behavior Category

Type	Behavior
Positive	Explicit Play(with his or her intention) Implicit Play(without intention) Play Completion(play without skip) Search and add an item to playlist Register a new song
Negative	Explicit Skip(by clicking ckip button) Implicit Skip(by playing other song) Delete an item form playlist

rating function which derives the predicted rating $p_{i,j}$ with minimum error.

3. EXPLOITING IMPLICIT FEEDBACK

3.1 User Behavior Analysis

The website [1], which we implemented, provides users with music recommendations and their own playlist. The user not only searches a song and adds it to the playlist, but registers a new song to the system. We have categorized users' behaviors into positive or negative feedback by merging segmented clickstreams into human actions. Table 2 shows summarized user behaviors which exhibit his or her preference of the music.

3.2 Rating Function

Our approach introduces a new rating function exploiting user's weighted behavior: $r_{i,j} = \sum_k f(s_i, u_j, b_k)$ with item s_i, user u_j, and behavior b_k. Function, $f()$ returns weights of the user's behavior b_k in order to calculate the rating value $r_{i,j}$. The problem is to find the optimized function, which can incorporate users' implicit actions into collaborative filtering.

4. EXPERIMENTS

We evaluate our method on the log datasets from our own website which provides 20,514 songs to the users. 32,568 action logs of 315 users are collected during 2 months. 1,158 songs are explicitly marked as their 'like' songs. The actions and rating functions are used to predict users' preferences. The predicted items are compared with the explicitly liked songs using *HR@topK* measurement [4]. Through a number of trials, we could find sets of parameters which predict users' preference better than the previous approaches. Table 3 shows parameters used in experiments, which 'B.' means baseline method adding up consumption numbers and other methods 'T' are our approach with various parameters. Figure 1 presents our approach outperforms the baseline method, especially for recommending top-5 items. For further investigation into the effect of parameters on the results, we use linear regression by using weighting parameters as regressors and averaged HitRatios as regressand. We could find very interesting results from regression coefficients that are shown at the last column of Table 3. 'Play Completion' has matchless influence on the results, while other parameters have negligible values within the margin of error. Although we cannot conclude that 'Play Completion' is the most significant behavior as users' implicit rating, it provides an insight that adding up the number of consumption can be defective approach.

Table 3: Weighting parameters for Experiments

Behavior	B.	T1	T2	T3	T4	T5	T6	Coef.
Exp. play	1	10	10	10	3	1	5	-0.02
Imp. play	1	-3	5	5	2	1	1	-0.06
Complete	0	1	5	10	10	10	10	0.30
Search	0	5	3	10	2	5	5	0.01
Register	0	5	3	10	2	5	5	0.01
Exp. skip	0	-2	-3	-2	-1	-10	-5	-0.06
Imp. skip	0	-1	-2	-2	-1	-10	-5	-0.01
Delete	0	-2	-5	-10	-2	-10	-10	0.06

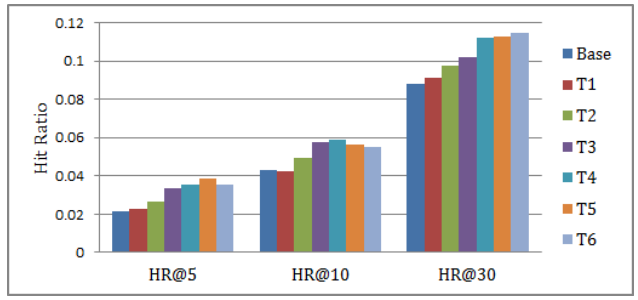

Figure 1: HR(Hit Ratio)@topK analysis with various weighting parameters

5. FUTURE WORK

In future work, we plan to improve our work on three ways. First, systemized behavior analysis will be done both equally and hierarchically. Second, we plan to use a large amount of log datasets to strengthen our approach. Finally, we will adopt some optimization techniques to find the best weighting parameters for various recommendation metrics.

6. CONCLUSION

In this paper, we proposed a novel method for recommender system by exploiting various implicit feedbacks. We categorized users' activities into positive or negative feedbacks by merging segmented clickstreams into human actions. A new rating function using implicit feedback is proposed with weighted behaviors. The experimental results show that our approach performs better than some baseline methods and completion of the play has more effect on the performance than other user behaviors.

Acknowledgement. This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 20110030812).

7. REFERENCES

[1] Dj tube. `http://djtube.co.kr`.

[2] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17:734–749, June 2005.

[3] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 263–272, Washington, DC, USA, 2008. IEEE Computer Society.

[4] D. Lee, S. E. Park, M. Kahng, S. Lee, and S.-g. Lee. Exploiting contextual information from event logs for personalized recommendation. *ICIS*, pages 121–139, 2010.

[5] J. Wang, A. P. de Vries, and M. J. T. Reinders. A user-item relevance model for log-based collaborative filtering. In *ECIR*, pages 37–48, 2006.