A LEARNING RESOURCE RECOMMENDATION METHOD COMBINING USER SEQUENTIAL INTERACTION WITH COLLABORATIVE FILTERING

Wenjuan Niu, Zhendong Niu, Shiping Tang, Zhi Huang, Wei Wang, Yaxin Chen, and Xi Li
School of Computer, Beijing Institute of Technology,
Zhongguancun North Street, Haidian district, Beijing 100081, China
niuwenjuan_2012@163.com, zniu@bit.edu.cn

ABSTRACT
Recommender system can make personalized predictions of resources for users with their learning history automatically. Collaborative filtering is one of the most widely used algorithms in this field. Although various works of collaborative filtering has been researched in e-learning, few of them notice the influence of sequential interactions among users. In this paper, we propose a novel collaborative filtering method by using the sequential interaction information of users. The proposed method consists of four steps: (1) fetching sequential interactive information from comments and replies; (2) computing the interaction influence degree among users with data from step (1); (3) filling the sparse user-item matrix with influence data; and (4) applying the new filled matrix to user-based collaborative filtering to find similar users to recommend. The experiment results on TED dataset show that the proposed method outperforms user-based CF and item-based CF on both precision and recall.

KEY WORDS
E-learning; collaborative filtering; sequential interaction; learning resource recommendation; learning material;

1. Introduction
In recent years, many e-learning systems spring up on the web, such as Cousera, Ted, Khan Academy etc. Learning resources recommendation has become a challenging issue and therefore an important direction of research [13].

Recommender systems have been researched and deployed extensively over the last decade in various application areas, including e-commerce and e-health. Salehi, M et al. categorize the previous works into four groups: data mining, content-based filtering [6], collaborative filtering [7], and hybrid approach [8], which are widely discussed in the literature and in several surveys of the state-of-the-art.

Content-based recommendation algorithms (CB) and collaborative filtering (CF) are two popular types. CF approaches used in e-learning environments focus on the correlations among users having similar interests and can be divided in to two categories: (a) looking for users who share the same rating patterns with the current user and then using the ratings from those like-minded users to calculate predictions for him/her; (b) finding out similar items preferred by users and using the ratings from these similar items to compute a prediction for the current user. A common approach of CF is based on neighborhood models, which is based on similarities among users or items [12].

Li, Y. et al. describe a web log mining approach to recommend learning items for each active user based on the user’s historical learning track. Different from other recommendation strategies, which use CF or SPM separately, their approach combines the two algorithms together and makes some optimizations to adapt them for E-learning environments. Salehi, M et al. proposed a hybrid recommendation approach for learning material based on sequential pattern of the accessed material and the user's preference tree. Their research considers the multidimensional attributes of material, rating of users, and the order and sequential patterns of the user’s accessed material in a unified model. Experimental results show that the approach outperforms other algorithms.

Although existing researches can provide the users with some personalized learning recommendation, they nearly consider the user interaction influence among each other. If user A tags the resources as favorite soon after user B for many times, it’s very likely that A is influenced by B. In addition, the influence between A and B are not balanced. Based on this idea, many researchers [2, 9, 10] propose their Collaborative filtering algorithms based on user relationship mining and achieve good results. How to obtain the relationship among users and quantify the similarities of users is the key step. There are two main types of the existing solutions: one is to use an explicit social relationship, the other computes the similarities of users by implicit tag information to get the relationship among users.

However, few real e-learning systems provide enough social relations and the tag information. In this paper, we find that in most systems users can comment on the resources and reply to others. Our proposed method uses users’ these interaction behavior, digs out the interaction influence among them, and then applies the data to fill the user-item matrix and find the nearest neighbor users. The interaction influence is bidirectional so it can describe the relationship among users accurately. Another advantage of our method is unlimited types and organization of learning platform resources.
In this paper, we discuss how to recommend learning resources using user interaction influence, and demonstrate that it’s useful to consider the user interaction influence by comment and reply. The proposed method consists of four steps: (1) fetching sequential interactive information from comments and replies; (2) computing the interaction influence degree among users with data from step (1); (3) filling the sparse user-item matrix with influence data; and (4) applying the new filled matrix to user-based collaborative filtering to find similar users to recommend.

The rest sections of the paper are organized as follows. Section 2 introduces the recommendation model that uses CCR-UCF (combining sequential interaction influence among users by comments and replies with user-based collaborative filtering) method. In Section 3, experiments are conducted and results are shown for the evaluation of the proposed approach. Finally, conclusion and future work are presented in section 4.

2. Recommendation algorithm

In this paper, we focus on how to recommend items matching people’s interests in TOP-N recommendation task.

In this section, we discuss our proposed method with considering user sequential interaction using comments and replies. Then we combine the sequential interaction with user-based collaborative filtering. Four steps of our proposed method are described in detail.

2.1 Fetching sequential interactive information from comments and replies

Learning resources has many attributes such as subject, title and description and so on. However, we find out that the organization of the learning resources in e-learning systems are commonly divided into two categories: with a syllabus framework and without a syllabus framework. With a syllabus framework, the learning resources belong to some lecture of a course, like in Coursera, while without a syllabus framework, the resources are not a part of a course but classified to some topic, like Ted Talks. On the other hand, the multimedia nature makes it difficult to calculate content similarity between two items. This situation makes it a quite complex problem to recommend resource universally.

Therefore, we focus on collaborative filtering recommendation algorithm which incorporates users’ history and features data and is very suitable to e-learning resources recommendation.

In this paper, three types of information are used for user modeling. (1) Favorite is used to model user preference; (2) Comment is used to model user comments on resources; (3) Reply is used to model users’ replies to others. The users’ sequential interaction consists of comment and reply.

We use $C_{i,j}$ to represent the times that user $i$ gives comments on the same resource after user $j$. $R_{i,j}$ represents the times that user $i$ gives replies to user $j$ directly. $i \in U, j \in U$. $U$ represents all the users in the dataset.

2.2 Computing the interaction influence among users

We define comment influence degree of user $i$ on user $j$ as $Inf_{C(i,j)}$ and reply influence degree of user $i$ on user $j$ as $Inf_{R(i,j)}$. $i \in U, j \in U$.

$$Inf_{C(i,j)} = \frac{C_{i\rightarrow j}}{\sum_{k \in U, k \neq i} C_{i\rightarrow k} + 1}$$

(1)

$$Inf_{R(i,j)} = \frac{R_{i\rightarrow j}}{\sum_{k \in U, k \neq i} R_{i\rightarrow k} + 1}$$

(2)

$Inf_{C(i,j)}$ $(0 \leq Inf_{C(i,j)} \leq 1)$ and $Inf_{R(i,j)}$ $(0 \leq Inf_{R(i,j)} \leq 1)$ constitute the sequential interaction influence degree of user $u_i$ on user $u_j$, which we define as $Inf(u_i, u_j)$.

Parameter $\lambda$ is used as an adjustable parameter to weight the comment influence and $\theta$ weights the reply influence. The definitions of the two parameters are as follow:

$$0 \leq \lambda < \theta \leq 0.5$$

(3)

$$Inf(u_i, u_j) = \lambda \times Inf_{C(i,j)} + \theta \times Inf_{R(i,j)}$$

(4)

We represent user interaction influence information in matrix $Inf(U)$. The influence value is calculated as follows in Eq.4. $U$ is the total member of users in dataset. Matrix $Inf(U)$ is like as follows in Table1:

<table>
<thead>
<tr>
<th>user id</th>
<th>$U_1$</th>
<th>$U_2$</th>
<th>...</th>
<th>$U_u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U_1$</td>
<td>-</td>
<td>0.03</td>
<td>...</td>
<td>0.001</td>
</tr>
<tr>
<td>$U_2$</td>
<td>0.01</td>
<td>-</td>
<td>...</td>
<td>0.11</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$U_u$</td>
<td>0.09</td>
<td>0</td>
<td>...</td>
<td>-</td>
</tr>
</tbody>
</table>

The influence value is not balanced from the matrix’s dissymmetry.

2.3 Filling the sparse user-item matrix with influence data

In some e-learning environment, users can rate the resource from 0 to 5 or other values. However in many learning systems, users can’t rate but can only mark their
favorite items. Given such situation, we assign the value of the favorite resource a user has marked as 1.0, the others that the user doesn’t mark as favorite as 0.

Assume we have $n$ users and $m$ resources. Then we multiply the influence matrix into the user favorite matrix. The updated prediction is calculated as follows using Eq.5:

$$M'(u,i)_{m,n} = Inf(U)_{n,m} \times M(u,i)_{m,n}$$  \hspace{1cm} (5)$$

The main process step (1), step (2) and step (3) are illustrated in Fig.1. After the prediction computation, we obtain the new user-item matrix for generic user-based recommendation.

2.4 User-based Collaborative Filtering

In the process of user-based CF, ratings are represented using a user-item rating matrix as calculated in step (3). The user-based collaborative approach can be described in the following four steps:

1) Getting the User Similarity Matrix: computing the similarities among users to get the similarity matrix of

$$M'(u,i)_{n,m} = Inf(U)_{n,n} \times M(u,i)_{n,m}$$

Figure 1. Process of generating new user-item matrix
users by Cosine similarity.

2) Searching nearest neighborhood: selecting other k users which have highest similarities with the current one.

3) Getting the Users’ New Preference Matrix: computing predictions on an item for a user by computing the user preference and the user similarity.

4) Selecting the top-N items to recommend: making recommendation by selecting the top-N large prediction items to target users and the recommendation is finished.

In this paper, we choose Cosine similarity to calculate similarity between user $i$ and user $j$. Cosine similarity is calculated as Eq.6:

$$sim(i,j) = \cos(\vec{i}, \vec{j}) = \frac{i \cdot j}{\|i\| \times \|j\|} (6)$$

3. Evaluation metrics

Traditionally, collaborative filtering algorithms are evaluated by the accuracy of their predicted ratings. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are two popular evaluation metrics. Recommendation algorithms with low MAE or RMSE are often considered as good ones. However, what the users of recommender systems often want is not to make a prediction for all items, but to find the best ones [16]. This is the top-N recommendation task, where a recommender system is trying to pick the best N items for people.

This task can be evaluated more informatively by using accuracy metrics of precision and recall, which measures how well the system generates a list of recommendations [4].

Precision is defined as the ratio of relevant items selected to number of items selected. Precision represents the probability that a selected item is relevant. Recall is defined as the ratio of relevant items selected to total number of relevant items available. Recall represents the probability that a relevant item will be selected [11]. Precision and recall depend on the separation of relevant and non-relevant items.

In this paper, we use precision and recall. They are calculated as Eq.4 and Eq.5, where $T(u)$ is the set of items in the test set that are interested by user $u$. $R(u)$ is an item set that is the TOP-N recommendation result for user $u$.

$$Precision(N) = \frac{\sum_u |R(u) \cap T(u)|}{\sum_u |R(u)|} (7)$$

$$Recall(N) = \frac{\sum_u |R(u) \cap T(u)|}{\sum_u |T(u)|} (8)$$

4. Experiments

User-based CF and Item-based CF are the two conventional collaborative filtering methods. One recommends items based on user similarity, while the other based on item similarity. In this section, we compare our method with item-based CF [14] and user-based CF [15] on TED dataset with TOP-N task. Then we introduce the evaluation metrics and analyze the experiment results to evaluate the effectiveness of the proposed method.

4.1 Experiment setup

The proposed recommendation approach is evaluated on TED dataset. We use 5-fold cross validation for the evaluation. Starting from the initial data set, five distinct splits of training and test data are generated. For each data split, 80% of the original set is included in the training data and 20% of it is included in the test data. Users’ rating history in the training set is used to generate recommendations according to different algorithms. The test set is then used to evaluate the recommendation results.

4.1.1 Dataset description

In the field of recommender systems, it is a common practice to use public available datasets from different application environments (e.g. MovieLens and Netflix) in order to evaluate recommendation algorithms. These datasets are used as benchmarks to develop new recommendation algorithms in given settings [5]. But in e-learning system, there’re few open datasets. Thanks for authors in paper [4], they explore TED datasets that capture user interactions with tools and resources.

In this paper, we use the TED dataset in 10 Sep 2012, which consists of 12605 users and 1203 talks. The talks have the following data fields: identifier, title, description, speaker name, TED event at which they were given, transcript, publication date, filming date, sum number of views and user comments. Each talk has a variable number of user comments. In addition, three fields were assigned by TED editorial staff: related tags, related themes, and pointers to related talks (generally three per talk). Users have an identifier and a list of talks marked as favorites. In this paper, we will only use the subset of 3,107 users who have made 12 or more ratings.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Talks</th>
<th>Active users</th>
<th>Favorites</th>
<th>comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>1,203</td>
<td>3,107</td>
<td>103,612</td>
<td>112,571</td>
</tr>
</tbody>
</table>

4.1.2 Experiment setting

$\lambda$ and $\theta$ are two important parameters. It is found that when $\lambda=0.01$ and $\theta=0.1$, the proposed method gains the best performance. Therefore, this pair of value is selected as empirical values in our experiments.

4.2 Results
In top-N recommendation task, at most only top-N (N<20) recommended results are considered. In this subsection, we present our experiment results in two aspects. Firstly we focus on sensitivity of similar neighbors. The neighbor sizes vary from 10 to 50 with recommending TOP-5 items. Then we discuss the sensitivity of recommendations. The recommendation sizes vary from 1 to 10.

### 4.2.1 Sensitivity of similar neighbors

The precision and recall are presented in Fig.2 and Fig.3. From Fig.2 we can see our CCR-CF method performs best, user-based CF and item-based CF gain the worse performance. When the number of similar neighbors is greater than 20, precisions of all methods decrease. From Fig.3, we can see that three methods show significant differences. The recall of CCR-UCF increases when the number of recommendations increase, the change of item-based CF is not obvious, and user-based CF decrease.

From subsection 4.2.1, the result is best when the number of neighbors is 20. Thus when we discuss the sensitivity of recommendations, the number of nearest neighbors is set as 20.

The size of similar neighbors plays an important role in recommendation quality. We experiment TOP-5 recommendations and vary size of recommendation from 1 to 10. The corresponding precisions and recalls are presented in Fig.4 and Fig.5. Fig.4 show that precisions of all method change a little. CCR-UCF is around 0.09, user-based CF is around 0.005 and item-based CF is around 0.008. From Fig.5, we can see that the recalls of three methods all increase when we increase size of recommendations. Obviously, CCR-UCF method performs better than the other two methods. That is because the new sequential interaction user-item matrix contains user relationship which is helpful to find neighbors effectively.

![Figure 2. Precision of various top-5 recommendation sizes](image)

![Figure 3. Recall of various top-5 recommendation sizes](image)

![Figure 4. Precision of various similar neighbor sizes](image)

![Figure 5. Recall of various similar neighbor sizes](image)

### 4.2.2 Sensitivity of recommendations
Therefore, we can see that the proposed method CCR-UCF achieves much better than item-based CF and user-based CF. It’s useful and effective to use user sequential interaction information of comment and reply to find similar neighbors.

5. Conclusion

In this paper, we propose a novel learning resource recommendation method CCR-UCF considering user interaction influence, which combines users’ comments and replies with user-based CF together. Experimental results show that our proposed method has better performance than user-based CF and item-based CF. But we ignore the content of comments and replies in proposed method. The content of comments and replies should to be mined in further work. We will further explore algorithms inspired from this method. Also the TED dataset has rich content information to be exploited. We will hybrid the content and user behavior together to make better recommendation.

Acknowledgements

This work is supported by Beijing Institute of Technology - Taipei University of Technology cooperation and exchange programs (No. 307001221404), the National Natural Science Foundation of China (Project No. 61370137) and the 111 Project of Beijing Institute of Technology.

References


