ECAI 2010
H. Coelho et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-606-5-607

Recommendations Over Domain Specific User Graphs

Makoto Nakatsuji[1], **Yasuhiro Fujiwara**[2], **Akimichi Tanaka**[3], **Tadasu Uchiyama**[4] and **Toru Ishida**[5]

Abstract.

Content providers want to make recommendations across multiple interrelated domains such as music and movies. However, existing collaborative filtering methods fail to accurately identify items that may be interesting to the user but that lie in domains that the user has not accessed before. This is mainly because of the paucity of user transactions across multiple item domains. Our method is based on the observation that users who share similar items or who share social connections, can provide recommendation chains (sequences of transitively associated edges) to items in other domains. It first builds domain-specific-user graphs (DSUGs) whose nodes, users, are linked by weighted edges that reflect user similarity. It then connects the DSUGs via the users who rated items in several domains or via the users who share social connections, to create a cross-domain-user graph (CDUG). It performs Random Walk with Restarts on the CDUG to extract user nodes that are related to the starting user node on the CDUG even though they are not present in the DSUG of the starting user node. It then adds items possessed by those users to the recommendations of the starting node user. Furthermore, to extract many more user nodes, we employ a taxonomy-based similarity measure that states that users are similar if they share the same items and/or same classes. Thus we can set many suitable routes from the starting user node to other user nodes in the CDUG. An evaluation using rating datasets in two interrelated domains and social connection histories of users as extracted from a blog portal, indicates that our method identifies potentially interesting items in other domains with higher accuracy than is possible with existing CF methods.

1 Introduction

Recently, many content providers offer items across multiple interrelated domains. For example, Amazon.com[6] offers items in several domains such as music, movies and fashions. To support such services, effective recommendations over multiple domains are essential. Most content providers adopt methods based on collaborative filtering (CF), which is a broad term for the process of recommending items to the active user, the one who is receiving the recommendation, based on the intuition that users who access the same items with the active user tend to have similar interests with the active user. They measure the similarity of users from just the co-rating behaviors against items. As a result, they are apt to recommend the types of items that have already been accessed often by the active user. For example, if the user highly rates a horror movie (as an item), the typical CF methods recommend items that were made by the same director, performed by the same actors, or included in the same genre, horror. The paucity of users' transactions across multiple item domains ensures that existing CF methods will fail to recommend items in domains that the user has not yet accessed. Even though the user has interests in many domains, the interests of just a few domains will be strongly emphasized. Thus, the recommendations created by existing CF methods for him will be biased toward those few domains.

Though cross-domain recommendation is becoming more important, few studies have addressed it. Bin and his co-workers analyze users who take similar rating behaviors against items across several item domains[9, 8]. Their method shares the knowledge that is learned by using the rating datasets from multiple item domains even when the users and items of these datasets do not overlap. Their evaluation, which used rating datasets of movie domains and those of a book domain, shows that their method outperforms the method that uses knowledge learned by using individual rating datasets in terms of the accuracy of prediction results in each domain. However, their method does not aim to predict user preference against items in the domains that the user has not accessed before.

In this paper, we try to identify items of interest in domains that the active user has not accessed before. It will a key tool for the content providers that want to offer items across multiple interrelated domains, especially when they have a large rating datasets in some domains while for some other domains they can collect only limited rating datasets. For example, if the recommender system can recommend many good non-Japanese music items to a user who has been listening to only Japanese music items (and vice versa), the user may widen his interests by accessing those items and enjoy more of the pleasures available in the world.

For achieving the above goal, we take the following two approaches.

The first approach is based on the observation that users who share similar items or who share social connections, can provide recommendation chains (sequences of transitively associated edges[6]) to items in other domains. Take rating web sites against Japanese music items and non-Japanese music items for example. On one hand, Japanese music and non-Japanese music have some correspondence in terms of genre (both have similar genres such as "Rock" and "Pop" though instances are actually different types of music artists.). While the user sets are different from each other, they are the subsets sampled from the same population (this assumption only holds for popular web sites) and so are expected to reflect similar social aspects[3]. Thus, our method first creates a domain-specific-user graph (DSUG) whose nodes are users and sets weighted edges between user nodes according to the similarity of users computed in each domain. It also creates a social-based user graph (SUG); an edge is set between a pair

[1] NTT Cyber Solutions Laboratories, NTT Corporation, Japan, email: nakatsuji.makoto@lab.ntt.co.jp
[2] NTT Cyber Space Laboratories, NTT Corporation, Japan, email: fujiwara.yasuhiro@lab.ntt.co.jp
[3] NTT Cyber Solutions Laboratories, NTT Corporation, Japan, email: tanaka.akimichi@lab.ntt.co.jp
[4] NTT Cyber Solutions Laboratories, NTT Corporation, Japan, email: uchiyama.tadasu@lab.ntt.co.jp
[5] Department of Social Informatics, Kyoto University, Japan, email: ishida@i.kyoto-u.ac.jp
[6] http://www.amazon.com

of user nodes if they exhibit some social relationship. It then connects DSUGs, created for different item domains, and the SUG via the users who rate items in several domains or who have a social relationship, to create a cross-domain-user graph (CDUG). Next, it performs Random Walk with Restarts (RWR)[10] on the CDUG from the active user node and analyzes the frequency with which the walk passes through nodes on the CDUG. It identifies users who are highly related with the active node user even though they are located in domains that the active user node does not lie in. Finally, it incorporates items possessed by those identified users in the recommendations of the active user.

Restricting the process of measuring similarity of users, computed in each domain, to just the items shared by users means that the number of similar users is relatively small. As a result, there are few edges from such user nodes to other user nodes, and the walk on the CDUG cannot transit many user nodes. This situation is often encountered when the walk on a CDUG crosses into a different DSUG via nodes of users that have rated several items in different item domains. For example, there are many users who have rated many Japanese music items but few non-Japanese music items. If the random walk enters from the DSUG created against Japanese music items to the DSUG created against non-Japanese music items via such user nodes, it is unable to transit to many user nodes in the DSUG of the non-Japanese music domain. To set many realistic routes from the active user node to other user nodes in other DSUGs, we adopt the taxonomy-based approach in measuring the similarity of users; the second approach of the paper. Taxonomies of items are designed by the service providers to enable their customers to access their preferred items easily. Thus, we consider that users who like an item, are expressing a liking for the class that includes that item, and our method reflects the rating of the user on an item to that of the class that includes that item. Then, it can measure similarity of users in each item domain by using not only co-rating behaviors against items but also those against classes in the taxonomy.

We evaluate our method using the rating datasets of two different item domains; implicit ratings against non-Japanese music artists (as items) and those against Japanese music artists extracted from blog entries in a blog portal. We also extract users' social connection histories from the same blog portal to create a SUG. Next, we create a CDUG by connecting the DSUGs created by non-Japanese and Japanese datasets, and the SUG. We confirm that our method identifies items in domains, which the active user did not access before, with higher accuracy than the method that predicts user preference from a DSUG created from the mixed dataset; implicit ratings against both non-Japanese and Japanese music artists. We also confirm that our taxonomy-based similarity measure is well suited for creating DSUGs and achieves higher accuracy than existing similarity measures including a previous taxonomy-based method[17].

The paper is organized as follows: we describe related works in the next section and explain the background of our study. Next, we explain how to create a CDUG by connecting DSUGs and a SUG, to achieve cross-domain recommendation and how to measure the similarity of users following the taxonomy of items. We then evaluate our method using two different datasets and users' social connection histories. Finally, we conclude the paper.

2 Related Works

Several machine learning studies try to transfer the knowledge from different domains to learn a classification model or a ranking model in a target domain[9, 8, 15, 2]. However, to the best of our knowledge only one group has tackled cross-domain recommendations over interrelated item domains such as books and movies among those, Bin et al.[9, 8], which was introduced in the above section. This is because there is not enough ratings of the user to be learned when we focus on each user's ratings, to predict item preference of a user in a different domain. Our method can compute the recommendations in a domain that the user has not accessed before even if he has few rating datasets. However, it does need several users who rate items in several different domains to compute such recommendations.

Some researchers have started to use random walks or RWR on a graph to compute recommendations[4, 7, 16]. Konstas and his coworkers used the dataset of last.fm[7], which is one of the social network services in the music domain, and created a graph combining several different types of datasets including not only users' item listening histories but also the social annotation against items and social connections between users[7]. They confirmed that the graph model approach is a natural and effective approach since it allows different types of datasets such as social annotation and friendships to be combined and used to predict user preference. Thus, we also take the graph model approach to compute item preferences of the active user by combining several rating datasets from several different item domains. This represents, to the best of our knowledge, the first study to use random walks on the CDUG to identify items in domains that the active user has not accessed before.

Some CF researchers use a taxonomy of items to raise the accuracy of predicting user preference[17]. Their method was shown to be useful when the transaction data of users was sparse. However, in measuring user similarity, their method focuses only on classes that include items rated by both users and their super classes. As a result, this method naively assumes that users who share many items are highly similar with the user; those users may have many good as well as many not so good items for the user. We improve the prediction accuracy by measuring the similarity of users from a consideration of the "width" of user interests against classes in the taxonomy as we will explain in section 4.4. The authors in [13] assign a-priori scores to the classes in the taxonomy of items, and compute the relationships between the scores assigned to different classes. They then propagate those scores for a specific user to predict each user's preference. Their method was also shown to be useful when the transaction data of users was sparse. Their method lies outside the scope of CF methods because they did not compute similarities of users. Unfortunately, their method is not suitable for identifying items in domains that the user did not access before. It is because their method can not identify items in classes that are far in the taxonomy, from the classes that the user has accessed even if a compound taxonomy that covers several domains is created by merging the taxonomies via the root class of each.

3 Background

Our method extends CF and uses RWR to identify items in the domains that the active user did not access before.

3.1 Collaborative Filtering

In computing the similarity of users, basic CF methods use the Pearson correlation approach[12] or the cosine-based approach[1].

If we define M as the number of items rated by users a and u, r_{a,I_i} is the rating value of user a for item I_i, and $\bar{r_a}$ is the average value of item ratings given by a, the Pearson correlation coefficient measures the similarity $S(a,u)$ between a and u according to equation (1) .

[7] http://www.last.fm

$$S(a,u) = \frac{\sum_i^M (r_{a,I_i} - \bar{r_a})(r_{u,I_i} - \bar{r_u})}{\sqrt{\sum_i^M (r_{a,I_i} - \bar{r_a})^2}\sqrt{\sum_i^M (r_{u,I_i} - \bar{r_u})^2}} \quad (1)$$

When we use the cosine-based approach, we set $\bar{r_a}$ and $\bar{r_u}$ as zero in equation (1). The advantage of the Pearson correlation approach is that it takes into account that different users might have different rating schemes.

If we assume N is the set of users that are most similar to the active user a, the predicted rating of a on item I_i, p_{a,I_i} is obtained by the following equation (2).

$$p_{a,I_i} = \bar{r_a} + \frac{\sum_u^N (r_{u,I_i} - \bar{r_u}) S(a,u)}{\sum_u^N S(a,u)} \quad (2)$$

3.2 Random Walk with Restarts

In a graph, objects and their relationships can be represented as nodes and weighted edges respectively, where weights denote relationship strength. Measuring the relatedness of two nodes in the graph can be achieved by using the RWR technique[10]. Starting from node a, a RWR is performed by following a randomly selected link to another node at each step. Additionally, at every step there is a probability, α, that the walk restarts at a. Let $\mathbf{p^{(t)}}$ be a column vector where $p_u^{(t)}$ denotes the probability that the random walk at step t proceeds from node u. $\mathbf{q}$ is a column vector whose elements are set to zero; only the element corresponding to a is set to one, i.e. $q(a) = 1$. Also let $\mathbf{A}$ be the column-normalized adjacency matrix of the graph. In other words, $\mathbf{A}$ is the transition probability table where element $A(u,v)$ gives the probability of v being the next node given that the current node is u. The stationary probabilities for each node can be obtained by recursively applying equation (3) until convergence, and they give us the long-term visit rate of each node with a bias towards the starting user node.

$$\mathbf{p^{(t+1)}} = (\mathbf{1} - \alpha)\mathbf{A}\mathbf{p^{(t)}} + \alpha\mathbf{q} \quad (3)$$

Therefore, $p_a^{(l)}$, where l is the state after convergence, can be considered as a measure of relatedness between nodes a and u. The time spent in performing RWR on the graph is one problem, however a fast RWR computation technique was proposed by Tong et al.[14].

4 Method

Our method starts by creating a cross-domain-user graph (CDUG).

4.1 Creating a cross-domain-user graph (CDUG)

We first connect domain-specific-user graphs (DSUGs) and a social-based-user graph (SUG), to create a CDUG.

Suppose that we are given rating datasets in z related domains. We denote a set of users, $(U_z = {u^{(z)}}_1, ..., {u^{(z)}}_{n_z})$, make ratings on items in the z-th item domain, where n_z denotes the number of users. Users may belong to several domains, however, we treat users in each domain as different unique users. For example, a user u_1 is denoted as ${u^{(1)}}_1$ in the first domain and ${u^{(2)}}_1$ in the second domain. Items in different domains are assumed to be independent. We also denote a set of users who have social connections as $U_s = ({u^{(s)}}_1, ..., {u^{(s)}}_{n_s})$, where n_s denotes the number of users in U_s.

A DSUG of the z-th item domain is created by setting nodes as users in U_z and weighted edges are assigned from user ${u^{(z)}}_i$ to ${u^{(z)}}_j$ according to similarity of ${u^{(z)}}_i$ to ${u^{(z)}}_j$. The similarity is measured using the rating dataset in the z-th domain based on either the Pearson correlation approach or the cosine-based approach (see previous section). In detail, we use similarity scores between users to build a column-normalized adjacency matrix of the DSUG in equation (3). Each row of values is linearly scaled up such that the maximum of each row corresponds to 1 and the minimum of each row corresponds to 0. The SUG is also created by setting nodes as users and weighted edges are assigned from user node ${u^{(s)}}_i$ to ${u^{(s)}}_j$ according to the access frequency of user u_i to u_j. The column-normalized adjacency matrix of the SUG is created as the same way as that of the DSUG.

Then our method connects the DSUGs via the users who rated items in several domains or via the users who share social connections, to create the CDUG. Here we need to model the column-normalized adjacency matrix $\mathbf{A}$ used in equation (3) against the CDUG. We first set β which is the probability that the random walk enters the adjacent nodes of ${u^{(s)}}_i$ in the SUG after it transits a node ${u^{(z)}}_i$ in the z-th domain. This parameter determines how strongly we emphasize the social connections in predicting users' preference items. If the random walk transits the node ${u^{(z)}}_i$ corresponding to user u_i who rated items in K different domains and shares social connections with some users in all domains, the walk next transits to the user node in one of the K different DSUGs with probability of $(1-\beta)/K$. If user u_i does not share social connections with any user in all domains, the walk next transits to the user node in one of the K different DSUGs with probability of $1/K$.

The initial state probability of the user node corresponding to u_i in the SUG and DSUGs is given as described below. If u_i shares social connections with some users in all domains, the initial state probability of the user node corresponding to u_i in the SUG, ${u^{(s)}}_i$, is β. If u_i shares no social connections with users, the initial state probability is 0. In the same way, if u_i rated items in K different domains and shares social connections with some users in all domains, the initial state probability of the node corresponding to u_i in each DSUG is $(1-\beta)/K$. If u_i rated items in K different domains and shares no social connections with some users, the initial state probability of the node corresponding to u_i in each DSUG is $1/K$.

A toy example that uses two small column-normalized adjacency matrices of the DSUGs and one small matrix of the SUG, is shown in Fig. 1. Here, we set parameter β to one third (1/3). For example, the probability after the random walk transits a node ${u^{(1)}}_1$ and then enters the node ${u^{(1)}}_3$ is computed as $0.7 \times \frac{(1-1/3)}{2} = 0.233$, the node ${u^{(2)}}_3$ is computed as $0.5 \times \frac{(1-1/3)}{2} = 0.166$, and the node ${u^{(s)}}_3$ is computed as $0.3 \times \frac{1}{3} = 0.10$, respectively.

4.2 Identifying items in other domains

Next, it executes RWR on the CDUG. By performing RWR on the graph, we can acquire the probability that a walk from active user node a will pass through user node u_i on the CDUG; RWR, equation (3), is iterated until convergence is realized. Finally, we can acquire the relatedness between active user node a and other user nodes on the CDUG. In Equation (3), decreasing α allows the walk more frequently to pass through users in the DSUGs in which a is not included. We discuss the effect of this parameter in our evaluations.

By computing the *relatedness* between user a and the user in the DSUGs, in which a is not included, and using *relatedness* instead of *similarity* between user a and the user, we compute the prediction values of items in those DSUGs for user a using equation (2).

However, the walk can not transit a lot of user nodes if we use present similarity measurements between users such as Pearson correlation coefficient and cosine-based similarity, to create DSUGs. This is because if there are a few items shared by users, there are few users that are similar to those users. This situation is often observed when the random walk enters one domain from a different

Column-normalized adjacency matrix of the DSUG created for domain 1.

	$u_1^{(1)}$	$u_2^{(1)}$	$u_3^{(1)}$
$u_1^{(1)}$	0.0	0.9	0.4
$u_2^{(1)}$	0.3	0.0	0.6
$u_3^{(1)}$	0.7	0.1	0.0

Column-normalized adjacency matrix of the DSUG created for domain 2.

	$u_1^{(2)}$	$u_3^{(2)}$	$u_4^{(2)}$
$u_1^{(2)}$	0.0	0.4	0.8
$u_3^{(2)}$	0.5	0.0	0.2
$u_4^{(2)}$	0.5	0.6	0.0

Column-normalized adjacency matrix of SUG.

	$u_1^{(s)}$	$u_2^{(s)}$	$u_3^{(s)}$	$u_4^{(s)}$
$u_1^{(s)}$	0.0	0.3	0.8	0.3
$u_2^{(s)}$	0.3	0.0	0.1	0.5
$u_3^{(s)}$	0.3	0.6	0.0	0.2
$u_4^{(s)}$	0.4	0.1	0.1	0.0

Connecting user DSUGs and the SUG.

Column-normalized adjacency matrix of CDUG.

	$u_1^{(1)}$	$u_2^{(1)}$	$u_3^{(1)}$	$u_1^{(2)}$	$u_3^{(2)}$	$u_4^{(2)}$	$u_1^{(s)}$	$u_2^{(s)}$	$u_3^{(s)}$	$u_4^{(s)}$
$u_1^{(1)}$	0.00	0.45	0.133	0.00	0.133	0.00	0.00	0.45	0.133	0.00
$u_2^{(1)}$	0.10	0.00	0.20	0.10	0.20	0.00	0.10	0.00	0.20	0.00
$u_3^{(1)}$	0.233	0.05	0.00	0.233	0.00	0.00	0.233	0.05	0.00	0.00
$u_1^{(2)}$	0.00	0.00	0.133	0.00	0.133	0.40	0.00	0.00	0.133	0.40
$u_3^{(2)}$	0.166	0.00	0.00	0.166	0.00	0.10	0.166	0.00	0.00	0.10
$u_4^{(2)}$	0.166	0.00	0.20	0.166	0.20	0.00	0.166	0.00	0.20	0.00
$u_1^{(s)}$	0.00	0.15	0.266	0.00	0.266	0.15	0.00	0.15	0.266	0.15
$u_2^{(s)}$	0.10	0.00	0.033	0.10	0.033	0.25	0.10	0.00	0.033	0.25
$u_3^{(s)}$	0.10	0.30	0.00	0.10	0.00	0.10	0.10	0.30	0.00	0.10
$u_4^{(s)}$	0.133	0.05	0.033	0.133	0.033	0.00	0.133	0.05	0.033	0.00

Figure1. Example of creating the column-normalized adjacency matrix of the CDUG.

domain. For example, there are many users who have rated many Japanese music items but few non-Japanese music items. If the walk moves from the DSUG created for Japanese music items to that created for non-Japanese music items via those user nodes, it can not transit many user nodes in the DSUG created for non-Japanese music items. Thus, we try to establish more edges between different DSUGs by applying the taxonomy-based approach in measuring user similarity as described in the following subsections.

4.3 Modeling user interests

Taxonomies are becoming more readily available; examples include the taxonomies of music, movies, and game content generated by All Media Guide[8] and ListenJapan[9]. We consider that modeling user interests according to these taxonomies is reasonable because content providers are making significant efforts to optimize the granularity and branching factors of classes to better satisfy their customers.

Our approach is based on the observation that users who are interested in some items, are also interested in the classes that include those items; thus the rating values of the items are reflected in those of the classes that include those items. The rating value for an item is implicitly assigned according to the frequency of a user's access to the item, or explicitly assigned by the user.

We rate the class from the ratings of items in the class. Formally, let $\mathcal{I}$ be an item set in class C_i, the rating value of the class, r_{u,C_i}, is computed as $\sum_{I_i \in \mathcal{I}} r_{u,I_i}$. For example in Fig.2, if user u assigns rating value 4.0 against I_5, and 4.5 against I_6, the rating value of class C_3 for u is 8.5. The rating values of the super class are computed in the same way; a key point is that we use the rating of each class instead of the rating of each item.

4.4 Measuring similarity of users

Next, we explain how to assess the similarity of users a and u according to the taxonomy of items.

4.4.1 Approach

- We first compute $S(a, u, C_i)$, the score of interest agreement between user a and u against class C_i. This rating value takes a smaller value in r_{a,C_i} and r_{u,C_i}. Thus we can filter users of low-interest when measuring the score of interest agreement.
- Next, we compute the similarity of rating behaviors against all classes between a and u, denoted as $S_C(a, u)$, with $S(a, u, C_i)$.

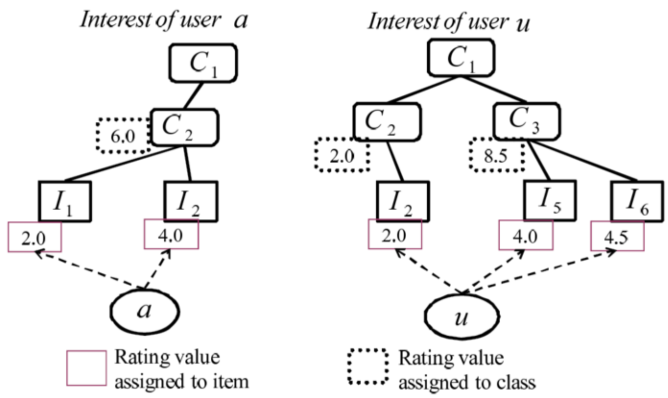

Figure2. Measuring similarity of users a and u.

We utilize the idea of the Jaccard coefficient since it can effectively separate user u who assigns ratings to many classes from a who assigns ratings to fewer classes. The Jaccard coefficient considers *not* co-rating classes as well as co-rating classes by utilizing the union of class sets. In other words, it considers the similarity of the "widths" of users' interests. For example, readers can naturally guess that users who love only rock genre is somewhat different types of users from users who love both rock and classic genres. The Pearson correlation approach and cosine-based approach do not have this property since they only consider the classes that are assigned ratings by both a and u (see equation (1)).
- We then measure $S_I(a, u)$, the similarity of rating behaviors against items between a and u. We use the Pearson correlation approach because it can handle the difference in the rating schemes of each user against items as explained in equation (1). Note our approach can also employ the Jaccard coefficient approach.
- Finally, we combine the two above similarities, $S_C(a, u)$ and $S_I(a, u)$, to evaluate the similarity of rating behaviors against classes and items between users.

The proposed method can effectively measure the similarity of the "widths" of users' interests as well as offset the differences in the rating schemes of users. It is a natural approach and achieves high accuracy as demonstrated in our evaluation.

4.4.2 Algorithm

We introduce the algorithm of our method below. In this algorithm, we use $\mathcal{C}_i(a)$ as the subclasses of class C_i that user a rates.

[8] http://www.allmediaguide.com/
[9] http://listen.jp/

1. First, it computes $S(a,u,C_i)$ as $min(r_{a,C_i}, r_{u,C_i})$.
2. Then, it measures similarity scores $S_C(a,u)$ as follows.

$$S_C(a,u) = \sum_C \frac{\sum_{C_j \in \{\mathcal{C}_i(a) \cap \mathcal{C}_i(u)\}} S(a,u,C_j)}{|\mathcal{C}_i(a) \cup \mathcal{C}_i(u)|} \quad (4)$$

3. Next, it uses the Pearson correlation approach to compute similarity scores $S_I(a,u)$ using equation (1).
4. Finally, it normalizes $S_C(a,u)$ and $S_I(a,u)$ among all users, and determines the similarity of users $S(a,u)$ as $S_C(a,u) + S_I(a,u)$.

Example We explain our algorithm by using the example in Fig.2. (1) $S(a,u,C_2)$ is computed as $min(6.0, 2.0) = 2.0$. $S(a,u,C_3)$ is computed as $min(0.0, 8.5) = 0.0$ (2) $S_C(a,u)$ is computed as $(2.0/2) + 0.0 = 1.0$. (3) Next, $S_I(a,u)$ is computed as $\frac{(4-3)(2-3.5)}{\sqrt{(4-3)^2}\sqrt{(2-3.5)^2}} = -0.667$.(4) Finally, $S(a,u)$ is measured following step four of our algorithm.

5 Evaluation

We now evaluate our method using the following datasets.

5.1 Datasets

User implicit ratings against non-Japanese music artists This dataset includes 48,695 implicit ratings from 3,545 users according to a taxonomy extracted in the experiment of Nakatsuji et al. from the blog portal Doblog[10] and the taxonomy of non-Japanese music artists provided by ListenJapan[11]. The taxonomy contains 279 genres as classes and 21,214 artists as items[11]. Nakatsuji et al. created a user's rating values for each item by analyzing the description frequency of each item among the user's blog entries. The average number of ratings assigned to an item is 6.3. We linearly scaled up each rating value such that the maximum user rating corresponded to 5 and the minimum corresponded to 1 following the range of ratings in MovieLens dataset[12]. The class hierarchy in this taxonomy is deep; it has, on average, four hierarchies, and sometimes has a fifth hierarchy under the root class "Music" with detailed end classes such as "Space rock" and "Acid jazz". This represents detailed expert knowledge that can be used to measure similarity of users accurately.

User implicit ratings against Japanese music artists We also used 58,104 implicit ratings from 2,800 users extracted from blog entries in Doblog using a Japanese taxonomy provided by ListenJapan in the same way as Nakatsuji et al. did for the non-Japanese taxonomy. The Japanese taxonomy contains 153 genres as classes and 7,421 artists as items. The class hierarchy in this taxonomy is also as deep as that in the non-Japanese taxonomy. The average number of ratings assigned to an item is 10.8.

User implicit ratings using mixed taxonomy We also used 106,799 implicit ratings from 4,825 users against items of the mixed taxonomy; non-Japanese taxonomy and Japanese taxonomy. This dataset was acquired by merging the non-Japanese and Japanese datasets. Mixed taxonomy contains 432 genres as classes and 28,635 artists as items. The average number of ratings assigned to an item is 7.4. This dataset can be considered as a mixed but single domain dataset, music domain, and is used for creating a DSUG against the music domain. We used this dataset in comparing the performance when performing RWR on our CDUG connecting DSUGs against non-Japanese and Japanese music dataset with that when performing RWR on the DSUG against the mixed domain.

Social connection histories of users We also analyzed the social connection histories of blog users from April 2006 to April 2008 in Doblog. The social connection histories are stored in the Doblog database whenever a user accesses another user's blog site. The users extracted are restricted as the users in the non-Japanese music artist dataset or in the Japanese music artist dataset. As a result, we extracted 199,3716 social connections among 3432 users. We built a SUG by setting users as nodes and edges are set from user node u to v according to the access frequency of user u to the blog site of v.

5.2 Methodology

We randomly divided dataset D that includes items with user ratings into two datasets: training dataset T and predicted dataset P. Thus, we could acquire users who had items whose domain is in P though it is not included in T. We then measured the similarity of users using T, and created a user graph to measure the relatedness between users on the graph. We prepared T by setting a ratio of T to D, $\frac{T}{D}$, to 0.7.

Following the standard evaluation methodology for CF, we predicted the user ratings only on the withheld ratings in T and computed Mean Absolute Error (MAE), which penalizes each miss by the distance to the actual rating. This measure is written below, where n is the number of entries in P, and P_i and R_i are the predicted and actual ratings of the ith entry, respectively.

$$MAE = \frac{\sum_{i=1}^{n} |P_i - R_i|}{n} \quad (5)$$

We also check the prediction coverage, i.e. the ratio between the items predicted by the method and the items included in P[5].

In our evaluation, we focused on users who had items whose domain is in P though it is not included in T, and evaluated the MAE and the prediction coverage against those users. From our dataset, we could acquire 126 such users when we set $\frac{T}{D}$ to 0.7.

5.3 Compared methods

We compare the performance realized by applying RWR to our CDUG created using non-Japanese dataset and Japanese dataset, with that when performing RWR on the DSUG created using the mixed dataset in music domain. We also compare our similarity measurement method to the following similarity measures.

- Pearson correlation coefficient (*Pearson*): similarity of users is measured by Pearson correlation coefficient.
- Cosine-based approach (*Cosine*): similarity of users is measured by cosine-based approach.
- Method by Ziegler (*Ziegler*): similarity of users is measured by the method proposed by Ziegler et al.[17]. This measures the similarity of users without regard to the "width" of user interests. We set parameter χ in [17] to 0.2 to achieve the most accurate results.
- Taxonomy (Jaccard & Pearson) (*T(J&P)*): this is our method explained in detail in the method section.

5.4 Results

We first set the number of users, N in equation (2), in each method such that the prediction coverage of all methods reached 80% and also the MAE achieved the most lowest value when we changed N from 100 to 1000. As a result, N of our method was set to 1000. We also checked the number of users who accessed two different

[10] http://www.doblog.com/; Unfortunately, Doblog terminated services on May 2009.
[11] The music taxonomies can be accessed from ListenJapan home page.
[12] http://www.grouplens.org/node/73

Table1. MAE when we set $\frac{T}{D} = 0.7$.

α	*Pearson (mixed)*	*Cosine (mixed)*	*Ziegler (mixed)*	*T(J&P) (mixed)*	*Pearson (CDUG)*	*Cosine (CDUG)*	*Ziegler (CDUG)*	*T(J&P) (CDUG)*	*Pearson (social)*	*Cosine (social)*	*Ziegler (social)*	*T(J&P) (social)*
Non	1.14	1.29	1.18	1.18								
0.8	1.25	1.20	1.16	1.19	1.21	1.13	1.11	***1.10***	1.21	1.12	1.11	***1.09***
0.6	1.25	1.19	1.17	1.20	1.21	1.13	***1.10***	***1.10***	1.21	1.13	***1.10***	***1.10***

item domains, and there are 1,387 number of such users among 4,825 users. There are quite a few number of such users.

Next, we evaluated the accuracy of our method. Results when we set $\frac{T}{D}$ to 0.7 are shown in Table 1. Here, "Non" in each table indicates the results when RWR was not performed.

Most methods yielded higher accuracy for CDUG (results identified by "*CDUG*" in Table 1.) than the same methods when RWR was performed on the DSUG created by the mixed dataset (results identified by "*mixed*" in Table 1.). This indicates that the accuracy is not so good if we perform RWR on the DSUG created by the mixed dataset. The similarity of users is computed more properly using a rating dataset against items of a single domain than using that against items of a mixed domain. Thus, we consider that computing users who are similar with the active user in a single domain and analyzing highly correlated users in another domains by performing RWR on the CDUG is a better approach in predicting user interests than performing RWR on the DSUG created based on the similarities of users computed by using the mixed rating dataset.

Furthermore, our method, *T(J&P)*, achieves higher accuracy than the other methods including the previous taxonomy method, which achieves the second highest accuracy among the methods. We investigated the bias of the items in two different domains rated by users who rated items in both domains. As a result, on average, users assigned ratings against items in one domain four times more often than those in another domain. In other words, there are often few edges that connect different DSUGs because users, who rate several different items, often rate many items in some domains but rate few items in other domains. Taxonomy-based methods set many suitable edges from the active user node to user nodes in DSUGs that do not include the node of the active user. Thus, this approach is more suitable in identifying items in domains that user did not access before than other methods.

Next, we create the CDUG by connecting the DSUGs created by the non-Japanese dataset, the Japanese dataset, and the SUG created by analyzing social connections between users. The parameter β in section 4.1 is set to 0.3. The results yielded by the methods when applied to this CDUG are identified by "social" in Table 1. The results show a slight improvement in accuracy of several methods. Thus, social connections between users can be used to more accurately identify items in domains that the active user did not access before.

6 Conclusion

This paper proposed a method that predicts user's interest in items in domains that the user has not accessed before. We calculate taxonomy-based similarity scores to create, in each item domain, a domain-specific-user graph (DSUG) whose nodes are users (weighted edges are assigned between users according to the similarity of users). We also create a social-based-user graph (SUG) by analyzing the social connections between users, and connect DSUGs and the SUG via users who rate items in several domains or who have social connections, to create the cross-domain-user graph (CDUG). Our method then performs Random Walk with Restarts (RWR) on the CDUG from the active user node and extracts user nodes that are present in DSUGs that do not include the node of the active user. An evaluation that used rating datasets against two different domains and an SUG extracted from blogs, indicated that the accuracy of our method is higher than the method that predicts user preference from a DSUG created by merging the rating dataset of each domain. We also confirmed that our taxonomy-based similarity measure well suits the creation of DSUGs and achieves higher accuracy than existing similarity measures. We will apply our method to cross-domain recommendation over more heterogeneous item domains such as music and movies. We consider our method works well in such situations because it uses only domain-independent factors, such as similarity of users and the social relationships of users, to connect different DSUGs.

REFERENCES

[1] J. S. Breese, D. Heckerman, and C. Kadie, 'Empirical analysis of predictive algorithms for collaborative filtering', in *Proc. UAI '98*, pp. 43–52.

[2] Keke Chen, Rongqing Lu, C. K. Wong, Gordon Sun, Larry Heck, and Belle Tseng, 'Trada: tree based ranking function adaptation', in *Proc. CIKM '08*, pp. 1143–1152.

[3] M. Coyle and B. Smyth, 'Web search shared: Social aspects of a collaborative, community-based search network'. Springer, (2008).

[4] M. Gori and A. Pucci, 'Itemrank: A random-walk based scoring algorithm for recommender engines', in *Proc. IJCAI '07*, pp. 2766–2771.

[5] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, 'Evaluating collaborative filtering recommender systems', *ACM Trans. Inf. Syst.*, **22**(1), 5–53, (2004).

[6] Zan Huang, Hsinchun Chen, and Daniel Zeng, 'Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering', *ACM Trans. Inf. Syst.*, **22**(1), 116–142, (2004).

[7] I. Konstas, Stathopoulos, V., and J. M. Jose, 'On social networks and collaborative recommendation', in *Proc. SIGIR '09*, pp. 195–202.

[8] Bin Li, Qiang Yang, and Xiangyang Xue, 'Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction', in *Proc. IJCAI '09*, pp. 2052–2057.

[9] Bin Li, Qiang Yang, and Xiangyang Xue, 'Transfer learning for collaborative filtering via a rating-matrix generative model', in *Proc. ICML '09*, pp. 617–624.

[10] L. Lovasz, *Random Walks on Graphs: A Survey*, volume 2, Bolyai Society Mathematical Studies, combinatorics edn., 1996.

[11] M. Nakatsuji, M. Yoshida, and T. Ishida, 'Detecting innovative topics based on user-interest ontology', *J. Web Sem.*, **7**(2), 107–120, (2009).

[12] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, 'Grouplens: An open architecture for collaborative filtering of netnews', in *proc. CSCW '94*, pp. 175–186.

[13] Vincent Schickel-Zuber and Boi Faltings, 'Inferring user's preferences using ontologies', in *Proc. AAAI '06*, pp. 1413–1418.

[14] H. Tong, C. Faloutsos, and J. Y. Pan, 'Fast random walk with restart and its applications', in *proc. ICDM '06*, pp. 613–622.

[15] Bo Wang, Jie Tang, Wei Fan, Songcan Chen, Zi Yang, and Yanzhu Liu, 'Heterogeneous cross domain ranking in latent space', in *Proc. CIKM '09*, pp. 987–996.

[16] H. Yildirim and M. S. Krishnamoorthy, 'A random walk method for alleviating the sparsity problem in collaborative filtering', in *proc. RecSys '08*, pp. 131–138.

[17] C. N. Ziegler, G. Lausen, and L. S. Thieme, 'Taxonomy-driven computation of product recommendations', in *proc. CIKM '04*, pp. 406–415.