# A Framework for Cross-domain Recommendation in Folksonomies

Ying Guo and Xi Chen
Department of Automation, Tsinghua National Laboratory for Information Science and Technology(TNList), Tsinghua University, Beijing, China. 100084
Email: {guoying10, bjchenxi}@mails.tsinghua.edu.cn

***Abstract*—Even though recommendation systems have achieved great success both in commerce and academy, there is still much to improve in cross-domain recommendation field. In this paper, we propose a novel framework for cross-domain recommendation in folksonomies: CRF. The idea of CRF is generating user's tag-profile in the target domain, based on the correlation of tags between different domains. Then the cross-domain issue is transferred into traditional single domain recommendation problem. Compared to related work, CRF is more flexible and scalable, it can adapt to multi-cross-domain recommendation issues. As it is a framework, CRF can be implemented in various ways according to practical requirements. Moreover, CRF is based on folksonomy, so it can be widely used in various applications of Web 2.0.**

**In addition, data sets from previous work are far from satisfaction, so we build a cross-domain data set for evaluation. We validate different realizations of CRF and demonstrate its effectiveness. The test results show that when we choose typical tags as features, the algorithm performs the best. The experiments also show that CRF is more precise than one-domain recommendation algorithms to solve cold-start problem in the target domain.**

***Index Terms*—data mining, cross-domain recommendation, folksonomy, collaborative filtering.**

## I. Introduction

In the era of explosive information, recommendation systems have gradually become an indispensable part of network applications. In academic research, various issues have been studied, such as recommendation algorithms [1]-[3], evaluation metrics [4] and [5], context-awareness recommendation [6], human interaction [7], and so on; in actual business applications, recommendation systems have also been demonstrated to be very successful [8] and [9]. However, most of the efforts and achievements only focus on within-one-domain recommendation up to now. For cross-domain recommendation problem, there still exists huge potential both in academic and in business.

Cross-domain recommendation can bring many benefits to both users and websites. In traditional recommendation systems, when users are browsing resources from one domain, the recommended list is only generated from this domain. So, why not recommend a classic movie "Forest Gump" when the user is browsing inspirational books? Why not recommend a science fiction when the user's preference to sci-fi movie is known? In this way, user experience improves by providing more diversified and serendipitous recommendations. In addition, as websites already have users' preference information in original domains, cross domain recommendation systems can be used to quickly open up new areas in business, saving precious time and money. Meanwhile, cold-start problem [10] or data sparseness problem in the target domain can be also solved by cross-domain recommendations.

Even though numerous methods have been developed for traditional single-domain recommendation, most of these methods cannot be directly applied to solve cross-domain recommendation problem. Traditional recommendation methods infer user's preferences based on behavior information from the same domain. On the contrary, behavior information in the target domain is unknown or little for cross-domain recommendation, information from other domains is used to make recommendation. In a word, whether known information and inference information are from the same domain is the main difference between traditional recommendation and cross-domain recommendation.

For cross-domain recommendation, if we can use behavior information in the source domain to deduce user's behavior information in the target domain, then the known information and inference information are from the same domain, and the cross-domain issues are transferred to single domain issues, the various methods in traditional single domain can be directly used. Therefore, the key challenge for cross-domain recommendation is how to build bridge to connect different domains.

Domains are mutually exclusive in general, each involving a certain type of resource (e.g., books, music, movies), it is difficult to extract common characteristics from resources to build the bridge among different domains. Here we use user-generated-tags instead of resources to link domains in this paper. Systems which use user-generated-tags are called folksonomy.

A folksonomy is a system that collaboratively creates and manages tags to annotate resources' characteristics [11]. It is widely used in various kinds of online

Manuscript received March 4, 2013; revised May 8, 2013.


doi: 10.12720/joace.1.4.326-331

applications, and becomes the symbol of Web 2.0 services. Instead of selecting specific resources as features, tags in folksonomies have more advantages for solving cross-domain recommendation problem: 1. Different domains have different resources, but share many tags with similar meaning. For example, “love” can be used as a tag for both a love story and a romantic movie. Therefore, it is easy to use tags as bridge to link domains. 2. Tags have better understanding of user’s preferences. If we know user’s favorite tags, we can directly get what factors are key to influence user’s preferences. 3. Tags can alleviate sparsity problem. For example, there are hundreds of thousands of resources in one domain in ecommerce websites, the matrix to describe relationships between users and resources is very sparse. but the number of tags in one domain will not exceed tens of thousands. If we use tags instead of resources to show what the users may like, the problem of matrix sparsity is eased by conversion from resources to tags.

Inspired by these thoughts, we propose a framework for cross-domain recommendation in folksonomies: CRF. The idea of CRF is generating user’s tag-profile in the target domain, based on the correlation of tags from different domains. Then the cross-domain issue is transferred into traditional single domain recommendation problem. CRF is scalable and can be widely used in various applications of Web 2.0. As it is a framework, CRF can be implemented in different ways to meet the needs. We validate different realizations of CRF and demonstrate its effectiveness. The experiments also show that CRF is more precise than one-domain recommendation algorithms to solve cold-start problem in the target domain.

The rest of this paper is organized as follows. In the next section, we introduce some related work in cross-domain recommendation field. In Section III, we give definitions and formulations of our problem and details of CRF are explained in Section IV. In Section V, we illustrate our data set and carry out experiments to validate the performance of CRF.

## II. Related Work in Cross-Domain Recommendation

Compared to the mature investigations in single-domain recommendation, cross-domain recommendation is a field worthy of further in-depth study.

Amit [11] adopts traditional single-domain recommendation methods to recommend resources from other domain, in order to evaluate impacts of different source domains on recommendation results. Cross-system user modeling [12] aggregates tag-based user profiles from different domains. However, all these works do not put forward a cross-domain recommendation algorithm.

As illustrated above, discovering linkage among domains is the key challenge for cross-domain recommendation, there are several papers try to solve it from different aspects. Codebook transfer [13] and rating-matrix generative model [14] both learn cluster-level rating pattern that could be shared between different domains. Multi-domain Collaborative Filtering [15] extends probabilistic matrix factorization to learn a correlation. Both rating pattern and correlation matrix mentioned above are implicit, [13]-[15] use the implicit information to transfer knowledge between different domains. On the contrary, TagCDCF [16] adopts common tags between different domains, and experiments demonstrate that this kind of explicit information is more reliable and effective for cross-domain recommendation. However, realization and effects of TagCDCF rely on common tags between different domains, if there are few even no common tags, TagCDCF can be hardly applied.

Up to now, Most of open data sets focus on single-domain data, for example Netflix data (http://www.netflixprize.com/) and MovieLens data (http://www.grouplens.org/node/73). Suitable cross-domain data sets are hard to obtain, not only because industry does not provide open cross-domain data to academy, but also because data sets from previous works in academy have the following shortcomings:

Firstly, format and content of some data sets are so special that the algorithms rely on them have limited practical usage. In reference [17], structural information of Linked data is used to build directed acyclic gragh based on semantics and then the cross-domain recommendation problem is simplified as how to find a route from the name of source location to the target music. The algorithm relies on Linked data, which includes names of well-known places but doesn’t include the names of hundreds of thousands of common people, so the algorithm is not flexible. Kaminskas and Ricci [18] build a cross-domain data set based on tags, but the categories of tags were limited to only a few predefined categories set and failed to satisfy realistic application condition for recommendation systems.

Secondly, some data sets are combined from multiple classical single-domain recommendation data sets, so the users from different domains cannot be guaranteed to be the same. For those test users whose behavior information in the target domain is missing, the history information of similar users in the target domain is used instead to test the performance. Even though this evaluation method is reasonable, it lacks precision compared to directly using test user’s own behavior information.

## III. Problem Definition

To clarify our framework for cross-domain recommendation in folksonomies, without loss of generality, we first suppose the problem’s environment as follows: there is one source domain $S$ and one target domain $T$. We use subscript $_S$ and $_T$ to differentiate the source domain and the target domain. Moreover, as discussed in previous part of this paper, tags are used instead of resources to represent user’s preference in folksonomies. Therefore, we make the following definitions.

*Definition 1*. Source domain/Target domain can be represented by $S/T = \{U, R, T, Y\}$, where $U$, $R$ and

$T$ are finite sets and called users, resources and tags respectively. $Y$ is the ternary tag assignment relation between them, $Y \subseteq U \times R \times T$. For $\forall u \in U$, $Y^u$ represents $u \times R \times T$.

*Definition 2*. User Profile for user $u$ is denoted as $\overrightarrow{profile_u} = (n_1^u, n_2^u, ..., n_m^u)$, m is the number of tags in this domain. For $\forall i \in [1, m], n_i^u$ measures the times user $u$ uses tag $\boldsymbol{i}$.

For cross-domain recommendation problems, if user's behavior information in the target domain $Y_T^u$ is known, we can directly use traditional single domain recommendation methods. Therefore, in this paper, we focus on the condition that user's behavior information in source domain $Y_S^u$ are known and $Y_T^u$ are few or unknown, in other words, this paper focuses on solving cold-start problems in the target domain from cross-domain recommendation methods. We formulate the problem as follows:

*Problem 1*. Cross-domain recommendation in folksonomies for user $u$. Given source domain $S$ and target domain $T$, $Y_S^u$ are known and $Y_T^u$ are few or unknown. The goal is to rank and recommend resources in $T$ to user $u$.

## IV. THE FRAMEWORK FOR CROSS-DOMAIN RECOMMENDATION IN FOLKSONOMIES(CRF)

TABLE I. THE FRAMEWORK FOR CROSS-DOMAIN RECOMMENDATION IN FOLKSONOMIES

| |
|---|
| **Input**: any test user $u$ and $Y_s^u$, train set of users $T_u$; |
| **Output**: a set of recommended resources in target domain; |
| 1. Given $u$ and $Y_s^u$, use ***Profile Generation Algorithm*** to get $\overrightarrow{profile_S^u}$ in source domain. |
| 2. Transform from $\overrightarrow{profile_S^u}$ to $\overrightarrow{profile_T^u}$ adopting ***Profile Mapping Algorithm***. |
| 3. Adopt classical single domain recommendation algorithms to generate recommendations. |

$$sim(u,b) = \frac{\sum_{i=1}^{m}(n_{iT}^u - \overline{n_T^u})(n_{iT}^b - \overline{n_T^b})}{\sqrt{\sum_{i=1}^{m}(n_{iT}^u - \overline{n_T^u})^2}\sqrt{\sum_{i=1}^{m}(n_{iT}^b - \overline{n_T^b})^2}} \quad (1)$$

$$pred(u,p) = \overline{r_T^u} + \frac{\sum_{b \in neighbors} sim(u,b)(r_{Tp}^b - \overline{r_T^b})}{\sum_{b \in neighbors} sim(u,b)} \quad (2)$$

Inspired by the essence of cross-domain recommendation issues and traditional single domain recommendation, we design the framework for cross-domain recommendation in folksonomies (CRF) as Table I shows. CRF is mainly composed of three steps. Firstly, user's information in source domain is used to build $\overrightarrow{profile_S^u}$. Secondly, we build a bridge between source domain and target domain to transform from $\overrightarrow{profile_S^u}$ to $\overrightarrow{profile_T^u}$, cross-domain issues are converted to single domain problem. At last, we can easily use classical single domain recommendation algorithms to generate recommendations as shown in step 3. Because single domain recommendation algorithms have been investigated by many researchers and our work focuses on "how to build the bridge", in step 3, we use kNN which has been demonstrated as one of the most effective recommendation methods [19]-[21] for ease of discussion. The process of kNN is as follows: given $\overrightarrow{profile_T^u}$, use (1) to find the most similar $k$ neighbors in $T_u$ after normalization, combine what the neighbors like, corresponding feedbacks with neighbor's similarity weights, to generate the output recommended $N$ resources with (2).

Next, we will explain *Profile Generation Algorithm* and *Profile Mapping Algorithm* in details.

### A. *Profile Generation Algorithm*

In general, information in folksonomies can be divided into two parts (see Fig. 1): user-resource information and resource-tag information. User-resource information indicates what movies the user has watched or what books the user has read in the historical records. Resource-tag information indicates what kind of tags people have used to describe the resources. CRF works under the following assumptions:

- Tags of a resource describe the resource's characteristics well;
- The most frequently used tags of a resource can represent all the tags of the resource. We call them as typical tags for short.
- If user $u$ has used resource $r$, the typical tags of $r$ show the preference of $u$.

Under these assumptions, *Profile Generation Algorithm* utilizes user-resource information and resource-tag information to generate user's tag-profile $\overrightarrow{profile_u} = (n_1^u, n_2^u, ..., n_m^u)$. The idea of the algorithm is to use "resource" as a bridge to build relationship between users and tags. We traverse each resource the user has experienced in one domain, for each typical tag the resource has, we add one to the corresponding number in user's tag-profile. The tag information of one resource may cause some errors, However, the statistical information of user's all historical records can exactly show his/her preference.

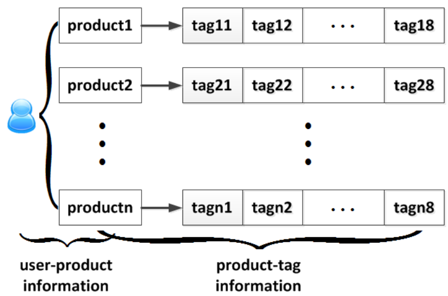


Figure 1. The structure of user's behavior information.

## B. Profile Mapping Algorithm

CRF intends to solve cold-start problem for test users in the target domain. The key is how to generate user's profile in the target domain under these circumstances. *Profile Mapping Algorithm* can solve this problem and its basic ideas are as follows:

Firstly, what we already have is the profile in the source domain and what we want to generate is the profile in the target domain. If we know the correlation between two domains' tags, we can find the most similar tag in the target domain for each tag from the user's given profile. For example, in Fig. 2, the most similar tag for $tag2$ is $tag1'$ in the target domain, so we add $n_{2S}$ in $\overrightarrow{profile_S^u}$ to $n_{1T}$ in $\overrightarrow{profile_T^u}$ ( $\overrightarrow{profile_T^u}$ is initialized as zero vector). In this way, we traverse every tag in the user's given profile, we can generate user's profile in the target domain.

Secondly, the concept of collaborative filtering [21] and [22] is adopted to measure the similarity between two domains' tags. For train set users, their profiles in these two domains are generated using ***Profile Generation Algorithm*** and then the user-tag matrixes like Fig. 3 are composed of these profiles. If we want to know what is the most similar tag in the target domain to $tag2$, we evaluate and rank the similarities of each column-vector in the target domain matrix and column-vector of $tag2$ using (1). In practical application, this step belongs to the preprocessing and does not impact the efficiency of CRF.

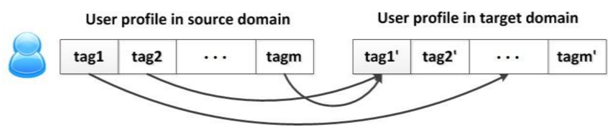


Figure 2. Illustration of tag-mapping process (1).

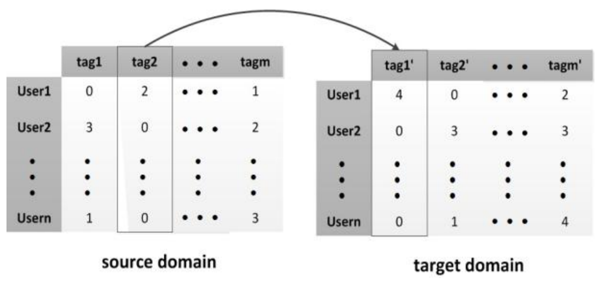


Figure 3. Illustration of tag-mapping process (2).

## C. Tag Selection

Now, we further discuss the methods of feature selection. Tag information is adopted as features to represent user's preference in folksonomies. As shown in Fig. 2 and Fig. 3, we use $[tag1, tag2, ...., tagm]$ and $[tag1', tag2', ...., tagm']$ to build preference space in source domain and target domain, respectively. From the aspect of feature selection, there are three ways to choose $[tag1, tag2, ...., tagm]$ and $[tag1', tag2', ...., tagm']$:

- *CRF-all*: all the distinct tags in this domain are used as features;
- *CRF-common:* as we want to build the bridge between the source domain and the target domain, we can choose the common tags shared by these two domains as features. If we adopt this method, there is no need to calculate the similarities between two domains' tags. CRF-common can also be seen as a specialization of CRF-all;
- *CRF-typical:* the scale of features impacts the algorithm's efficiency, if we can deduce the number of features, we can improve the algorithm's performance. Inspired by this theory, CRF-typical respectively adopts the typical tags in these two domains.

The performances of these CRF realizations are discussed in the following part.

# V. EXPERIMENTAL EVALUATION

## A. Data set Introduction

As illustrated in related work, data sets from previous works are far from satisfaction, so we build a cross-domain data set in this paper. Douban.com (http://www.douban.com/), which is one of the most popular Web 2.0 sites in China, has substantial user information and resource information in many domains. Moreover, it allows users to annotate tags to resources. Therefore, its data is very suitable for cross-domain recommendation research in folksonomies. Moreover, we focus our domains on "book" and "movie" for ease of discussion. Therefore, we have crawled the above information from douban.com as data set in Table II.

TABLE II. DOUBAN.COM DATA SET

| | |
|---|---|
| user information in movie domain | the number of users: 11697<br>average number of movies each user has watched: 81<br>total number of distinct movies: 30660 |
| user information in book domain | the number of users: 13772<br>average number of books each user has read: 20<br>total number of different books: 62510 |
| movie information | For each different movie in user's information, we crawled 8 typical tags.<br>total number of distinct tags: 31096 |
| book information | For each different book in user's information, we crawled 8 typical tags.<br>total number of distinct tags: 75004 |
| Common users from two domains | 7481 common users<br>average number of movies each common user has watched: 115<br>average number of books each common user has read: 33 |
| typical movie tags | 45 |
| typical book tags | 113 |

## B. Evaluations of CRF-all, CRF-common, CRF-typical

The scenario for testing is that given a user's behavior information in movie domain, we use CRF to recommend book list for this user, and validate how many books among the recommendation list the user has really read( the percentage is also called precision). Among the users who have watched more than 100 movies and have read more than 50 books, we randomly select 100 users as train set. The users of test set are randomly selected from the remaining users. For each algorithm, with different parameter values, we do the above experiment settings for three times, and output the average result (see Table III and Table IV). As explained in Section IV, we adopt kNN in the third step of CRF for ease of discussion and comparison.

TABLE III. PRECISION EVALUATIONS OF CRF REALIZATIONS

| | $k$ | $N$ | ***CRF-all*** | ***CRF-common*** | ***CRF-typical*** |
|---|---|---|---|---|---|
| 1 | 5 | 5 | 5.11% | 5.53% | ***7.78%*** |
| 2 | 5 | 10 | 4.04% | 4.47% | 6.56% |
| 3 | 5 | 20 | 3.83% | 4.52% | 5.94% |
| 4 | 10 | 10 | 3.19% | 3.62% | 3.89% |
| 5 | 15 | 10 | 1.17% | 2.77% | 2.00% |
| 6 | 15 | 20 | 1.81% | 2.26% | 2.83% |

TABLE IV. TIME AND SPACE COMPLEXITY OF CRF REALIZATIONS

| | ***CRF-all*** | ***CRF-common*** | ***CRF-typical*** |
|---|---|---|---|
| time complexity (cpu unit) | 15.14 | 0.72 | 0.075 |
| space dimension (source domain->target domain) | 12551->16294 | 1440->1440 | 45->113 |

The scale and quality of train set have important impacts on the evaluations. Therefore, we focus on the relative values of the test results and have the following conclusions:

- For each algorithm, precision decreases as $k$ increases in general. Because, if we add more neighbors' information, the new added user is just the one who has less similar taste with our test user;
- For each algorithm, precision decreases as $N$ increases in general. Because the resource added to the recommendation list is just the one which has lower possibility to satisfy the user's needs.
- Overall comparison of CRF-all, CRF-common and CRF-typical: CRF-all uses all the tags in two domains. So its time and space complexity is the largest. More importantly, its precision is the lowest, because too much redundant information sets obstacles to the analysis of preference. Compared to CRF-all, CRF-common not only deduces the number of dimensions, but also improves precision performances. As some of the common tags are not so informative, CRF-typical is the best of all as it directly focuses on the most typical and frequently used tags.

### C. *Comparison between One-Domain Recommendation and CRF*

CRF intends to find neighbors who have similar taste in the target domain. However, in order to solve cold-start problem in the target domain, there seems a more easy way to do: directly finding out neighbors who have similar taste in the source domain, then checking what resources the neighbors like in the target domain and generating recommendation list. This kind of direct algorithm needs not do the mapping, so we call it one-domain recommendation for short.

In order to validate necessity of cross-domain recommendation, we do experiments and show the results in Table V. The experiment's settings and scenario are the same as in the above experiments. From the results, we can see that although source domain and target domain are relevant, similar users in the source domain may have quite different taste in the target domain. This is why CRF's precision is higher than one-domain recommendation.

TABLE V. COMPARISON BETWEEN ONE-DOMAIN RECOMMENDATION AND CRF

| Precision comparison | ***CRF-all*** | ***CRF-common*** | ***CRF-typical*** |
|---|---|---|---|
| CRF | 5.11% | 5.53% | 7.78% |
| One-domain recommendation | 4.89% | 1.72% | 4.45% |

## VI. CONCLUSION

In this paper, we investigate cross-domain recommendation issues in folksonomies. We give the definitions and formulations of the problem. A novel framework for cross-domain recommendation in folksonomies (CRF) has been proposed to transfer knowledge between domains. As CRF is a framework, it can have different realizations according to practical requirements. We validate various realizations of CRF and the experiments show that CRF is effective and more precise than one-domain recommendation algorithms to solve cold-start problem in the target domain. Moreover, CRF is more scalable and flexible than previous works.

In the future, we can evaluate more realizations of CRF. For example, other classical single domain recommendation algorithms can be used. In addition, we can test CRF on data from more than two domains and further consider the evaluation methodology. It is also a promising direction to try CRF in other research fields, such as cross-domain research collaboration.

## ACKNOWLEDGMENT

This work is funded by Tsinghua National Laboratory for Information Science and Technology (TNList) Cross-discipline Foundation and is partially supported by the 111 International Collaboration Project (B06002).

## REFERENCES

[1] M. J. Pazzani, "A framework for collaborative, content-based and demographic filtering," *Artificial Intelligence Review*, vol. 13, no. 5-6, pp. 393-408, 1999.

[2] I. Markovsky, *Low-Rank Approximation: Algorithms, Implementation, Applications*, Springer, 2012.

[3] N. Barbieri, G. Costa, G. Manco, and R. Ortale, "Modeling item selection and relevance for accurate recommendations: a bayesian approach," in *Proc. fifth ACM Conference on Recommender Systems*, New York, USA, 2011. pp. 21-28.

[4] A. Gunawardana and G. Shani, "A survey of accuracy evaluation metrics of recommendation tasks," *The Journal of Machine Learning Research*, vol. 10, pp. 2935-2962, 2009.

[5] H. Steck, A. Lucent, and M. Hill, "Item popularity and recommendation accuracy," in *Proc. fifth ACM Conference on Recommender Systems*, New York, USA, 2011, pp. 125-132.

[6] G. Adomavicius and A. Tuzhilin, Chapter title: Context-Aware Recommender Systems. Book title: *Recommender Systems Handbook*, Springer, 2011, pp. 217-253.

[7] E. I. Sparling and S. Sen, "Rating: How difficult is it?" in *Proc. Fifth ACM Conference on Recommender Systems*, New York, USA, 2011, pp. 149-156.

[8] M. Levene, *An Introduction to Search Engines and Web Navigation*, Second Edition, Wiley, 2010.

[9] J. Davidson, B. Liebald, *et al.*, "The youtube video recommendation system," in *Proc. Fourth ACM Conference on Recommender Systems*, New York, USA, 2010, pp. 293-296.

[10] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and metrics for cold-start recommendations," in *Proc. 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, 2002, pp. 253–260.

[11] A. Tiroshi and T. Kuflik, "Domain ranking for cross domain collaborative filtering," in *Proc. 20th International Conference on User Modeling, Adaptation, and Personaliztion*, 2012, pp. 328-333.

[12] F. Abel, S. Araujo, Q. Gao, and G. J. Houben, "Analyzing cross-system user modeling on the social web," *ICWE*, vol. 6757, pp. 28-43, 2011.

[13] B. Li, Q. Yang, and X. Xue, "Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction," in *Proc. 21st International Joint Conference on Artificial Intelligence*, 2009, pp. 2052-2057.

[14] B. Li, Q. Yang, and X. Xue, "Transfer learning for collaborative filtering via a rating-matrix generative model," in *Proc. 26th International Conference on Machine Learning*, 2009, pp. 617-624.

[15] Y. Zhang, B. Cao, and D. Y. Yeung, (2012). Multi-domain collaborative filtering. arXiv preprint arXiv:1203.3535.

[16] Y. Shi, M. Larson, and A. Hanjalic, "Tags as bridges between domains: improving recommendation with tag-induced cross-domain collaborative filtering," in *Proc. 19th International Conference on User Modeling, Adaption, and Personalization*, 2011, pp. 306-316.

[17] I. Fernández-Tobías, I. Cantador, M. Kaminskas, and F. Ricci, "A generic semantic-based framework for cross-domain recommendation," in *Proc. 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, 2011, pp. 25-32.

[18] M. Kaminskas and F. Ricci, "Location-adapted music recommendation using tags," in *Proc. 19th International Conference on User Modeling, Adaptation and Personalization*, 2011, pp. 183-194.

[19] R. J. Mooney, "Content-based book recommending using learning for text categorization," in *Proc. Fifth ACM Conf. on Digital Libraries*, ACM Press, 2000, pp. 195-204.

[20] R. M. Bell, "Scalable collaborative filtering with jointly derived neighborhood interpolation weights," in *Proc. Seventh IEEE Int. Conf. on Data Mining, IEEE Computer Society*, Washington, DC, USA, 2007, pp. 43-52.

[21] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: item-to-item collaborative filtering," *Internet Computing*, *IEEE*, vol. 7, no. 1, pp. 76-80, Jan/Feb 2003.

[22] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conference on Computer Supported Cooperative Work*, New York, USA, 1994, pp. 175-186.



**Ying Guo** was born in Anshan, Liaoning Province, China. She received her BS from Tsinghua University, Beijing, China in 2010. Now she is a postgraduate in the Center for Intelligent and Networked Systems, Department of Automation, Tsinghua University, China. Her research interests are recommender systems and data mining.



**Xi Chen** was born in Chengdu, Sichuan Province, China. She received her BSc. and MEng. from Nankai University, Tianjin, China in 1986 and 1989 respectively. From July 1989 to September 1996, she worked with Software Engineering Institute at Beijing University of Aeronautics and Astronautics. Then she studied at the Chinese University of Hong Kong and obtained her PhD in 2000. Her PhD thesis wined the Outstanding Thesis Award, presented by the Faculty of Engineering, the Chinese University of Hong Kong.

She worked as a post-doctoral fellow at Information Communication Institute of Singapore and the Chinese University of Hong Kong during November 2000 and June 2003. She is now working with the Center for Intelligent and Networked Systems, Department of Automation, Tsinghua University, China. Her research interest falls in the fields of stochastic control and optimization, Markov decision process and wireless sensor networks.


 