Collaborative Filtering in Social Tagging Systems Based on Joint Item-Tag Recommendations

Jing Peng[1] Daniel Zeng[1, 2] Huimin Zhao[3] Fei-yue Wang[1]

[1] Institute of Automation, Chinese Academy of Sciences
[2] Department of Management Information Systems, University of Arizona
[3] Sheldon B. Lubar School of Business, University of Wisconsin-Milwaukee

jing.peng@ia.ac.cn zeng@email.arizona.edu hzhao@uwm.edu feiyue.wang@ia.ac.cn

ABSTRACT

Tapping into the wisdom of the crowd, social tagging can be considered an alternative mechanism—as opposed to Web search—for organizing and discovering information on the Web. Effective tag-based recommendation of information items, such as Web resources, is a critical aspect of this social information discovery mechanism. A precise understanding of the information structure of social tagging systems lies at the core of an effective tag-based recommendation method. While most of the existing research either implicitly or explicitly assumes a simple tripartite graph structure for this purpose, we propose a comprehensive information structure to capture all types of co-occurrence information in the tagging data. Based on the proposed information structure, we further propose a unified user profiling scheme to make full use of all available information. Finally, supported by our proposed user profile, we propose a novel framework for collaborative filtering in social tagging systems. In our proposed framework, we first generate joint item-tag recommendations, with tags indicating topical interests of users in target items. These joint recommendations are then refined by the *wisdom from the crowd* and projected to the item space for final item recommendations. Evaluation using three real-world datasets shows that our proposed recommendation approach significantly outperformed state-of-the-art approaches.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Information filtering*

General Terms

Algorithms, Experimentation, Performance

Keywords

Collaborative filtering, social tagging, tagging structure, joint recommendation, explanation

CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada.
Copyright 2010 ACM 978-1-4503-0099-5/10/10...$10.00.

1. INTRODUCTION

In recent years, social tagging has been gaining wide-spread popularity in a variety of applications, from social bookmarking sites (e.g., Delicious[1]and CiteULike[2]), movie rating sites (e.g., MovieLens[3]), to E-commerce sites (e.g., Amazon[4]). Social tagging systems encourage users to save and annotate Web resources of interest with tags. These tags not only allow users to conveniently revisit and retrieve previously-visited Web resources, but also enable them to search and explore what other users are interested in.

Social tagging can be considered a crowd-wisdom-based approach to information organization and discovery, an alternative to the traditional Web search engine approach. Enabling automated recommendation of various kinds in social tagging systems can further enhance this important social information discovery mechanism. In E-commerce applications, such recommendations can be a direct marketing tool. From the point of view of collaborative filtering research, tagging data generated by social tagging systems offer the potential to deliver substantially improved recommendation results as tags constitute a novel source of data complementing standard user-item interaction/rating information.

However, research on how to improve item recommendation leveraging tagging information is just emerging. Several methods that have been proposed, including the topic-based method [1], which views each tag as a distinct topic and computes the probability of a user saving an item through all tags, and the diffusion-based method [2], which generates recommendations by inspecting integrated information propagation on user-item-tag tripartite graphs. Most of the existing research either implicitly [1, 3-4] or explicitly [2] assumes a simple tripartite graph structure for social tagging systems. As we will point out in this paper, tripartite graph is not a faithful representation of the information structure of social tagging systems. Recently, a tensor-based approach [5] has been proposed to deal with the three dimensional structure of tagging data. Nevertheless, this approach is extremely expensive, computationally and spatially, in that the smoothed

[1] http://delicious.com/

[2] http://www.citeulike.org/

[3] http://movielens.umn.edu/

[4] http://www.amazon.com/

user-item-tag tensor of prediction utility, obtained by High Order SVD (HOSVD) [6], is usually not sparse.

Furthermore, all of the previous research focuses on recommendations of either items or tags, whereas items and tags co-exist in real tagging activities, with tags indicating the specific topics covered by a target item that attract a user. As such, the correctness of the resulting recommendations from the traditional tag-based methods, which do not pinpoint why a user may save an item, cannot be well-justified as they are not guaranteed to fall into a user's interested topics (tags). Realizing that it is of great value to explore the topics (tags) of a target item a user is interested in and how much she likes these topics, we propose a new tag-based recommendation framework by attempting joint item-tag recommendations before projecting them to the item space for final item recommendations. Beyond guaranteeing the justifiability of the item recommendations, joint item-tag recommendation has practical significance in its own right by explaining items with tags. According to a study conducted by Herlocker et al. [7], providing proper explanation is very helpful to promote the acceptance of recommendations.

The rest of this paper is organized as follows. In the next section, we briefly review the literature on tag-based recommendation. We then propose a comprehensive social tagging information structure and a unified user profiling scheme in section 3. Next, we propose a joint item-tag recommendation framework and discuss how to synthesize joint recommendation results for final item recommendations in section 4. In section 5, we report on empirical evaluation using three real-world datasets. Section 6 gives a brief discussion on the computational complexity as well as potential applications of our approach. Finally, we conclude the paper by summarizing our contributions and discussing prominent future research directions in section 7.

2. RELATED WORK

Collaborative filtering (CF) is the most widely-used and commercially successful approach to recommendation. A few methods have been proposed for tag-based collaborative filtering. A straightforward method is to use tags for computing user or item similarity. The user (item) similarities in standard user-based (item-based) CF methods are derived from the similarity of items (user) the users (items) interacted with. Zeng and Li [8] introduced two variants of the standard user- and item-based methods by calculating user and item similarities based on TF-IDF weighted tag vectors. Further, Zhao et al. [9] proposed to compute the similarity of two users based on the semantic distance of their tag sets on common items they have bookmarked. Tso-Sutter et al. [10] extended the item vectors for user profiles and user vectors for item profiles with tags and then constructed the user/item neighborhoods based on the extended user/item profiles. In addition, several other alternatives have been proposed to facilitate similarity computation using tags [11-13].

There are also a number of recent studies aiming at further use of tagging information for tag-based recommendation. The topic-based method [1] exploits tag information in a probabilistic framework, viewing each tag as an indicator of a topic and then estimating the probability of a user bookmarking an item by summing the transition probabilities through all tags. Zhen et al. [4] used users' tag vectors to regularize the user-item matrix factorization results by making sure that the similarity between two user's latent feature vectors are correlated with the tag sets of the two users. The subject-based method [14] tries to extract informative tagging patterns (subjects) from the user-tag and item-tag co-occurrence matrices using Consistent Nonnegative Matrix Factorization to explain why a user saved (or might save) an item. Recently, a diffusion method [2] was proposed to generate recommendations based on fusion of information diffusion on user-item and item-tag bipartite graphs.

A common feature of most of the above-mentioned methods is that the relationships among the three entities, i.e., user, item, and tag, are represented (sometimes implicitly) with a tripartite graph, such as the one shown in Figure 1 [15]. The methods of Zeng and Li [8] and Tso-Sutter et al. [10] assume a user-item-tag representation for their item-based method and tag-user-item representation for their user-based method. The method of Zhen et al. [4] implicitly takes a tag-user-item representation, where tag histories of users are used to regularize user latent feature vectors derived from user-item matrix factorization. The topic-based method [1] takes a user-tag-item representation, as user interest in items is essentially a result of user-tag and tag-item relationships. On the other hand, the diffusion method [2] rests on an explicitly stated user-item-tag representation, and the information is limited to propagate within user-item and item-tag bipartite graphs. Nevertheless, such tripartite graphs capture only two of three binary associations among the three entities, and the ternary association among the entities, which does exist in reality and cannot be decomposed into multiple binary associations, is lost.

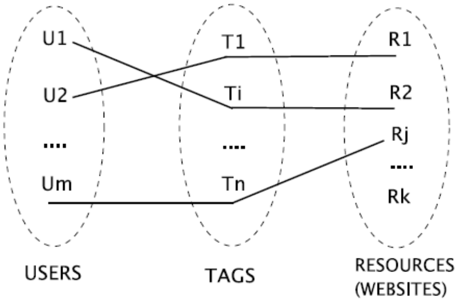


Figure 1. Tripartite graph structure of social tagging.

There are two recent studies [5, 16] that investigate the ternary association among users, items, and tags for more effective tag recommendation using tensor decomposition techniques. However, they cannot be readily adapted to item recommendation due to the different nature of tag recommendation and item recommendation. Tag recommendation aims at predicting the use of tags of a given user on a given item, with two entities predefined, whereas item recommendation aims at predicting the saving/purchasing of items, with only the user specified. Generally, item recommendation is more difficult than tag recommendation in that less information is known about the subject to receive the recommendations. Although Symeonidis et al. [5] argued that their approach was also applicable to item recommendation, they were actually recommending items to a given user with a given tag.

Moreover, since existing tensor factorization approaches, including PARAFAC Decomposition [17], HOSVD [6], and Tucker Decomposition [17], actually unfold a high order tensor into a series of two-dimensional matrices for processing, the relatively lower dimensional co-occurrence information embedded in the original high order tensor is completely ignored. More specifically, in the tagging domain, a user-item-tag tensor is flattened into user×(item,tag), item×(user,tag), and tag×(user,item) matrices. In this case, two users will not be considered correlated at all unless they have bookmarked some common items with the same tags, i.e., the two dimensional user-item interaction information is discarded. Also discarded is the user-tag and item-tag interaction information. As the value of the two dimensional information, which underlies most previous tag-based recommendation methods, has been well-justified in the literature, the user-item-tag tensor adopted in recent studies [5, 16] is not an ideal representation of the tagging data either. Another problem with this tensor representation is that it is unable to capture transactions without tag assignments.

3. UNIFIED USER PROFILING

It is essential for one to gain some deep insight into the underlying structure of social tagging systems before devising an effective user profiling method catering to item recommendation in the tagging domain. Towards that end, we first present our interpretation of the structure of social tagging in subsection 3.1. We then propose a novel user profiling scheme in subsection 3.2 based on our integrated social tagging structure. Subsection 3.3 presents an exemplary weighting strategy for our unified user profile. Finally, we present a tensor decomposition approach to extracting lower dimensional representation of users to ease the computation of user similarities in subsection 3.4.

Notation: In this paper, matrices are denoted by boldface capital letters, e.g., **P**. Tensors are denoted by boldface Calligraphy letters, e.g. $\boldsymbol{\mathcal{X}}$. Scalars are denoted by italic lowercase letters, e.g., λ, p_{ij}. Indices typically range from 1 to their italic capital version, e.g., $k = 1, \ldots, K$. Let $\boldsymbol{u} = \{u_1, u_2, \ldots, u_m\}$ be a set of users, $\boldsymbol{i} = \{i_1, i_2, \ldots, i_n\}$ be a set of items, and $\boldsymbol{t} = \{t_1, t_2, \ldots, t_l\}$ be a set of tags. The probabilities of observing an arbitrary user, item, and tag are represented by $p(u)$, $p(i)$, and $p(t)$, respectively. The joint probability of observing an arbitrary item and tag combination is denoted $p(i, t)$.

3.1 Integrated Structure of Social Tagging

As discussed in section 2, most of the existing research on tag-based recommendation either explicitly or implicitly assumes a tripartite graph structure for social tagging systems. While some recent studies are trying to represent the ternary <user, item, tag> relationship as tensors, the bipartite interaction between any two of these three entities, which underlies most existing research, is actually ignored in the tensor operations. To gain a more comprehensive interpretation of social tagging systems, we propose the following structure of social tagging behaviors, as shown in Figure 2.

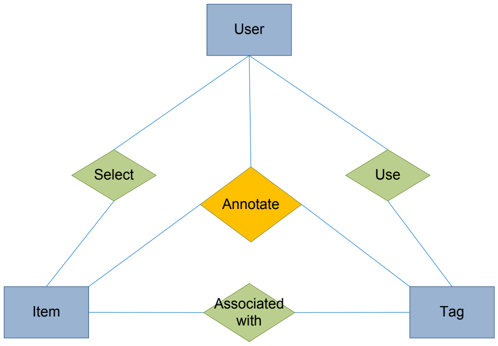


Figure 2. An integrated structure of social tagging.

The fundamental difference of the proposed structure with the commonly adopted tripartite graph structures is that user, item, and tag are treated equally as peer entities and have direct associations with each other. In addition to the binary relationships, a ternary association named "Annotate" is added to capture the simultaneous interactions among the three entities. Although the binary "Use" (B-Use) and "Associated with"" (B-Associated) relationships can be completely derived from the ternary "Annotate" (T-Annotate) relationship, they are not redundant in our structure because the binary information are often discarded when the ternary relationship is represented and operated as a tensor. The binary "Select" (B-Select) relationship cannot be completely derived from the T-Annotate relationship because users often save items without assigning any tag. Hence, our proposed structure has integrated all possible co-occurrence information among the three entities into one framework.

3.2 Unified User Profiling Scheme

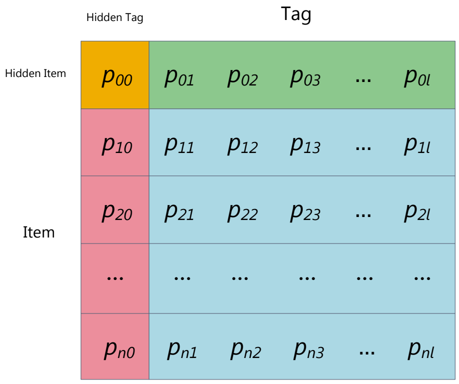
So far, there are three methods to profile users on the tagging data: i) profile a user with the item vector of the user's historical records [4, 9]; ii) profile a user with a tag vector of use frequencies [8]; and iii) profile a user with an extended 0-1 valued item-plus-tag vector [10]. In sum, the similarity between users is usually judged based on some of the following criteria:

1) Users that have saved common items are considered to be similar —B-Select similarity.

2) Users that have used common tags are considered to be similar — B-Use similarity.

3) Users that have used common tags on the same item are considered to be similar — T-Annotate similarity.

All existing profiling schemes are only able to capture user similarity on either one or two criteria, and none of them could take advantage of all these three types of similarities in a unified manner. T-Annotate similarity is the most reliable one in that it requires users to agree on both items and tags. Considering that the historical tagging records of a user naturally form a $n \times l$ item-tag matrix (named tagging matrix), it is straightforward to capture T-Annotate similarity with the tagging matrix. However, this matrix is typically very sparse and does not capture any similarity between two users when they have no T-Annotate similarity but some B-Select and/or B-Use similarities. To ensure that the denser B-Select and B-Use similarities are also functional in the user profile, we extend the tagging matrix as shown in Figure 3.



	Hidden Tag	Tag				
Hidden Item	p_{00}	p_{01}	p_{02}	p_{03}	…	p_{0l}
	p_{10}	p_{11}	p_{12}	p_{13}	…	p_{1l}
Item	p_{20}	p_{21}	p_{22}	p_{23}	…	p_{2l}
	…	…	…	…	…	…
	p_{n0}	p_{n1}	p_{n2}	p_{n3}	…	p_{nl}

Figure 3. Proposed unified user profiling scheme.

For each item, we assume that there exists one highly-correlated Hidden Tag, and whenever the item is being saved, this Hidden Tag will be used automatically. Another way to interpret the meaning of this Hidden Tag is to view an item itself as a super tag. As such, users will be considered to be similar to a certain extent through the Hidden Tag once they have saved the same item, even if they have assigned completely different sets of tags to the item. Likewise, we can assume a Hidden Item for each tag and use it to capture user similarities on merely tags. The underlying reason supporting our introduction of the Hidden Tag is that in most cases, individual users are unable to assign complete and accurate tags when saving an item due to a variety of reasons (e.g., laziness, use of non-descriptive tags for personal uses only, and spelling error), and the Hidden Tag can help to alleviate this problem to some extent. On the other hand, if two users have used the same tag, there should be some common interests between them, and such common interests are substantiated as a Hidden Item in our profiling scheme.

The Hidden Item Row (without the corner entry p_{00}) corresponds to the traditional tag profile, while the Hidden Tag Column corresponds to the traditional item profile. In particular, it is worth noting that the B-Associated relationship, reflected through the overall item-tag co-occurrence matrix, is not used in our profiling scheme as it is independent of individual users. The utility of this type of information will be discussed in section 4.2.

3.3 Weighting of User Profile Matrix

There could be many possible methods for weighting the elements in our user profile matrix. One example is as follows.

$$p_{ij} = \begin{cases} 2^{-\frac{k(1+lnK)}{\alpha}}, & \text{if user saved the } i\text{th item with the } j\text{th tag} \\ 0 \quad , & \text{otherwise} \end{cases} \qquad (1a)$$

$$p_{i0} = \begin{cases} 1, & \text{if user saved the } i\text{th item} \\ 0, & \text{otherwise} \end{cases} \qquad (1b)$$

$$p_{0j} = \sum_i p_{ij} / \sum_i p_{i0} \qquad (1c)$$

$$p_{00} = 0 \qquad (1d)$$

$$(\alpha > 0, 1 \leq i \leq n, 1 \leq j \leq l)$$

For the bottom-right T-Annotate submatrix, it is straightforward to weight each entry in a 0-1 scheme as in most existing approaches, with 1 indicating a tag assignment on an item. However, as argued in [18], it is unfair to treat all tags equally without considering the ranking order of each tag in the set of tags co-assigned to a bookmark as well as the size of this tag set. After moderate adaption to the weighting formula proposed in [18], equation (1a) is used to weight the T-Annotate entries, where k is the ranking index of the tag in the bookmark and K is the total number of tags assigned to this bookmark. α is an empirical parameter used to rescale the weight of tags. In formula (1a), the larger the number of tags co-assigned and the larger the ranking index, the smaller the weighting value a tag has. When α is sufficiently large, our weighting scheme will degenerate to the traditional method that treats all tags equally.

In (1b), we can see that the weight of Hidden Tag is uniformly set to 1. The underlying reason is that we believe that the assumed Hidden Tag is more important than the real tags and presume its ranking at position zero (i.e., k = 0), thus (1b) is a natural choice according to (1a). In addition, we set the weight of each Hidden Item to be the average weight of its corresponding tag over all bookmarking activities. For the corner entry p_{00}, we simply weight it with 0. In practice, it can also be assigned a nonzero weight for smoothing.

3.4 Dimensionality Reduction

While our profiling scheme has integrated all available information for profiling, it is attained at the price of extending the traditional item or tag vector profile to an item-tag matrix. Although this profile matrix can be efficiently stored in a sparse form, calculating similarities between very large sparse matrices could still be time-consuming. Recently, matrix factorization has proved to be an effective dimensionality reduction method in the field of recommender systems [14, 19]. Nevertheless, matrix techniques do not apply to our approach in that the profile matrices of all users actually constitute a three order profile tensor, which can be interpreted as an enhanced version of the traditional user-item-tag tensor in that it is able to capture binary B-Select and B-Use similarities by the introduction of Hidden Tag and Hidden Item. Hence, we employ the Tucker Decomposition [17, 20], which deals with tensor data, to extract informative lower dimensional representation of users. Below, we introduce some basic definitions of tensors. For more details, please refer to [17].

Mode-*n* matrix product The *n*-mode matrix product of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ with a matrix $\mathbf{U} \in \mathbb{R}^{J \times I_n}$ is denoted by $\boldsymbol{\mathcal{X}} \times_n \mathbf{U}$ and is of size $I_1 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N$. Element-wise, we have

$$(\boldsymbol{\mathcal{X}} \times_n \mathbf{U})_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum\nolimits_{i_n=1}^{I_n} x_{i_1 i_2 \cdots i_N} u_{j i_n}$$

Multiple mode-*n* matrix products can be performed in any order:

$$(\boldsymbol{\mathcal{X}} \times_n \mathbf{A}) \times_m \mathbf{B} = (\boldsymbol{\mathcal{X}} \times_m \mathbf{B}) \times_n \mathbf{A}$$

If the modes are the same, then

$$(\boldsymbol{\mathcal{X}} \times_n \mathbf{A}) \times_n \mathbf{B} = \boldsymbol{\mathcal{X}} \times_n (\mathbf{BA})$$

Tucker Decomposition Let $\boldsymbol{\mathcal{X}}$ be a tensor of size $I_1 \times I_2 \times \cdots \times I_N$. A Tucker decomposition of $\boldsymbol{\mathcal{X}}$ yields a core tensor $\boldsymbol{\mathcal{G}}$ of specified size $J_1 \times J_2 \times \cdots \times J_N$ and factor matrices $\mathbf{A}^{(n)}$ of size $I_n \times J_n$ for $n = 1, \ldots, N$ such that

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{G}} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_N \mathbf{A}^{(N)}$$

where the factor matrices $\mathbf{A}^{(n)}$ are assumed to be column-wise orthogonal, i.e., $\mathbf{A}^{(n)^{\mathrm{T}}} \mathbf{A}^{(n)} = \mathbf{I}$.

Tucker decomposition aims at minimizing the Frobenius error and can be solved efficiently using the Alternative Least Squares algorithm [20]. As an alternative generalization of the two-dimensional Singular Value Decomposition, Tucker decomposition gives a more strict approximation [17] of the original tensor as compared to HOSVD [5-6]. In our application, we decompose the three order user-item-tag profile tensor $\boldsymbol{\mathcal{T}}$ as:

$$\boldsymbol{\mathcal{T}} \approx \boldsymbol{\mathcal{G}} \times_1 \mathbf{U}^{(\text{user})} \times_2 \mathbf{U}^{(\text{item})} \times_3 \mathbf{U}^{(\text{tag})}$$

where $\mathbf{U}^{(\text{user})}$, $\mathbf{U}^{(\text{item})}$, $\mathbf{U}^{(\text{tag})}$ are factor matrices spanning the user, item, and tag subspaces of size $m \times d$, $(n+1) \times d$, $(l+1) \times d$, respectively, supposing that the dimensions of all the subspaces are uniformly d. Kolda and Sun proposed to use $\mathbf{U}^{(\text{user})}$ directly as a user representation for clustering [20]. However, $\mathbf{U}^{(\text{user})}$ is actually a basis matrix that spans the user subspace, rather than a feature matrix that holds the coordinates of users on the item×tag subspace. Hence, we propose to represent users with

$\mathcal{G} \times_1 \mathbf{U}^{(\text{user})}$, which is a tensor of size $m \times d \times d$. The $d \times d$ slice matrix of each user can be interpreted as her coordinate values in the item×tag subspace. This strategy is analogous to its two order counterpart in Latent Semantic Analysis [21], which factors a document-term matrix $\mathbf{X}$ as $\mathbf{X} = \mathbf{USV}^{\mathrm{T}}$ and then use $\mathbf{US}$ to represent documents.

Another problem that needs to be addressed is how to project new users into the item×tag subspace. To avoid updating the Tucker model at the arrival of each new user, we fold-in new users as follows:

$$
\begin{aligned}
&\mathcal{G}_{\text{new}} \times_1 \mathbf{U}_{\text{new}}^{(\text{user})} \\
&= \mathcal{G}_{\text{new}} \times_1 \mathbf{U}_{\text{new}}^{(\text{user})} \times_2 \left(\mathbf{U}^{(\text{item})^{\mathrm{T}}} \mathbf{U}^{(\text{item})}\right) \times_3 \left(\mathbf{U}^{(\text{tag})^{\mathrm{T}}} \mathbf{U}^{(\text{tag})}\right) \\
&= \mathcal{G}_{\text{new}} \times_1 \mathbf{U}_{\text{new}}^{(\text{user})} \times_2 \mathbf{U}^{(\text{item})} \times_3 \mathbf{U}^{(\text{tag})} \times_2 \mathbf{U}^{(\text{item})^{\mathrm{T}}} \times_3 \mathbf{U}^{(\text{tag})^{\mathrm{T}}} \\
&\approx \mathcal{T}_{\text{new}} \times_2 \mathbf{U}^{(\text{item})^{\mathrm{T}}} \times_3 \mathbf{U}^{(\text{tag})^{\mathrm{T}}}
\end{aligned}
$$

where $\mathcal{T}_{\text{new}}$ represents the profile tensor of new users. Note that the above approximation holds only when the addition of new users does not incur significantly changes to the result of $\mathbf{U}^{(\text{item})}$ and $\mathbf{U}^{(\text{tag})}$. In fact, if $\mathbf{U}^{(\text{user})}$ were used to represent users, this fold-in approach would not work as it is hard to cancel the core tensor $\mathcal{G}$ in the above induction.

When the lower dimensional representation of users is obtained, we compute the cosine similarity between users *a* and *b* as follows:

$$sim\left(\mathbf{F}^{\mathbf{a}}, \mathbf{F}^{\mathbf{b}}\right) = \frac{\sum_{i,j} f_{ij}^{a} f_{ij}^{b}}{\|\mathbf{F}^{\mathbf{a}}\| \|\mathbf{F}^{\mathbf{b}}\|} \qquad (2)$$

where $\mathbf{F}$ represents a user's feature matrix in the item×tag subspace and f_{ij} represents the <*i*, *j*> entry of this matrix. Note that other similarity metrics, such as Correlation, Euclidean distance, and KL-divergence, are also applicable here.

4. JOINT ITEM-TAG RECOMMENDATION

It is straightforward to make item recommendations directly following the traditional user-based method after the user similarity is computed. However, as we argued before, pure item recommendation faces some essential difficulties in providing quality recommendation due to its inability to capture users' explicit interest in the target items. In subsection 4.1, we present how to make joint item-tag recommendation based on our unified user profile. Then we show how to synthesize the joint item-tag recommendation result before projecting it to the item space for final item recommendation in subsection 4.2.

4.1 Problem Definition

To gain more insight into the real reason of why a user might save an item, we propose to recommend a joint item-tag matrix to each user, with the tags representing the topics of the target item that might attract the user. Given that we have profiled each user with an item×tag matrix, it is straightforward to recommend a profile matrix to the active user. This joint recommendation process is basically the same as the traditional user-based method except that the input and output for each user are matrices rather than vectors. Let $\mathbf{R}^{\mathrm{a}}$ be the recommended profile matrix for user *a*,

$$\mathbf{R}^{\mathrm{a}} = \frac{\sum_{b \neq a} s_{ab} * \mathbf{P}^{\mathrm{b}}}{\sum_{b \neq a} s_{ab}}, \qquad (3)$$

where s_{ab} represents the similarity between users *a* and *b*, and $\mathbf{P}^{\mathrm{b}}$ represents the profile matrix (not feature matrix) of user *b*.

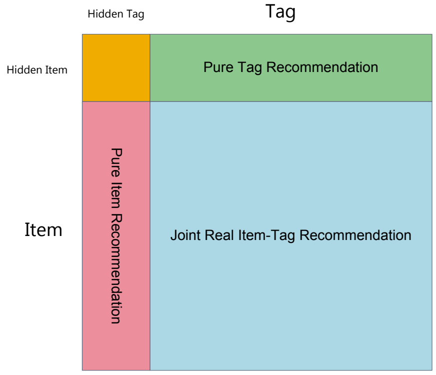


Figure 4. Joint item-tag recommendation results.

As shown in Figure 4, the recommended profile matrix for each user consists of four blocks. It is easy to understand that the bottom-right submatrix represents the joint recommendation result for real items and tags. Recalling that the Hidden Item Row and the Hidden Tag Column correspond to the traditional tag and item vector profiles for the user, we can see that the Hidden Tag Column actually holds the pure item recommendation result while the Hidden Item Row the pure tag recommendation result. The corner entry is zero here and still does not need to be considered yet.

Before generating final item recommendations based on the joint item-tag recommendations, we adjust the joint recommendation results as follows. For the pure item and joint real item-tag recommendations, we reset entries corresponding to items that have already appeared in the training set to 0. This is sensible in that we may not want to recommend resources that have already been saved by the user. For the pure tag recommendation result, we adjust it as follows:

$$p(t)^{pure'} = \lambda * p(t)^{pure} + (1 - \lambda) * p(t)^{init} \; (0 \leq \lambda \leq 1) \qquad (4)$$

where $p(t)^{pure}$ and $p(t)^{init}$ indicate the values of tag *t* in the Hidden Item Row of the recommended and initial profile matrices, respectively. This can be interpreted as a tradeoff between a user's potential (recommended) and current (initial) interest in tags. Note that to facilitate reading, we have normalized all the involved tag vectors to unit sum and denote them as probabilities.

4.2 From Joint Recommendation to Item Recommendation

As discussed before, item-only recommendations do not consider whether users are actually interested in the covered topics of the recommended items, hence might be erroneous as they might not fall into a user's interested topics (tags) at all. On the other hand, joint real item-tag recommendations explicitly consider a user's possible interest in each item with respect to each tag (topic). In addition, tag-only recommendations also provide valuable clues of the user's interest in each tag. Thus, an ideal item recommendation approach should make effective use of all these three types of recommendations in the recommended profile matrix in a principled manner.

We believe that joint recommendation holds the potential to deliver higher quality recommendations than any pure recommendation. However, the only joint form recommendation among these three types of recommendations, i.e., the joint real item-tag recommendation, is very sparse and may subject to noise. An intuitive solution would be generating a denser type of joint recommendations based on the two denser pure recommendations and then fuse it with the joint real item-tag recommendation. Thus, the critical problem lying ahead is how to make reliable joint recommendation based on the pure item and tag recommendations. Without loss of generality, we use the following equation to produce joint recommendation based on pure recommendations:

$$p(i,t)^{pure} \propto p(i)^{pure} \cdot p(t)^{pure} \cdot assoc(i,t) \qquad (5)$$

where $assoc(i,t)$ denotes the association between item i and tag t. Note that a normalization is performed afterward to ensure that $p(i,t)^{pure}$ represents a probability.

There are many ways [22] to calculate the association between two variables. In this paper, we use the Lift or Mutual Affinity [22-23] metric and pre-compute it based on the item-tag co-occurrence matrix in the training data:

$$assoc(i,t) = \frac{p(i,t)}{p(i)p(t)} = \frac{freq(i,t)/\sum_{i,t} freq(i,t)}{freq(i)freq(t)/(\sum_i freq(i) \sum_t freq(t))} \qquad (6)$$

Using equation (6) to compute the item-tag association enables us to take advantage of the B-Associated information, the only relationship that remains unused in our profiling scheme. The use of B-Associated information can be interpreted as correlating items and tags based on the *wisdom from the crowd*, as the item-tag co-occurrence matrix is aggregated over all users.

The final joint item-tag recommendation result can then be computed as a weighted average of the two joint recommendations, i.e.,

$$p(i,t)^{final} = \mu p(i,t)^{pure} + (1-\mu) p(i,t)^{real} \ (0 \le \mu \le 1) \quad (7)$$

where $p(i,t)^{real}$ represents the joint real item-tag recommendation result normalized to unit matrix sum.

Finally, a refined item recommendation can be obtained by marginalizing the final joint recommendation result, i.e.,

$$p(i)^{final} = \sum_t p(i,t)^{final} \qquad (8)$$

Thereafter, we can recommend the top-*N* items with the highest probabilities to the active user, and several relevant tags can be selected, based on the joint item-tag recommendation results, to explain each item recommendation if necessary.

5. EMPIRICAL EVALUATION

5.1 Datasets

We have tested our approach on three datasets. The first dataset was crawled from Delicious, the largest social bookmarking site. The collected dataset consists of bookmarking data of 5000 users dated from 12/1/2008 to 12/31/2008. These 5000 users were identified by taking a breath-first traverse of the Delicious user network, starting from a small set of randomly selected seed users. The second dataset is a snapshot of the CiteULike database[5] downloaded on 1/21/2010. We collected transactions that took place in year 2009. The last dataset is the Bibsonomy dataset[6] widely used in the tagging domain, and what we used is the 2009-07-01 snapshot. The Bibsonomy dataset contains bookmarks for both bibliographies and general Web resources, of which only the part for general Web resources was used in our experiment.

During data preprocessing, we iteratively removed users that had saved less than 10 items and items that had been saved by less than 10 users (8 for the bibsonomy dataset) until unqualified items are less than 20 for each dataset. In addition, we stemmed each tag using the Snowball stemmer[7] to alleviate the effect of word variations. For computational efficiency and recommendation quality, we only considered tags that had occurred more than 20 times (10 for CiteULike) in the training set. If a <user,item> co-occurrence does not involve any frequent tags, we set the tag entry as null but did not remove it. This is the key difference of our cleaning method with the k-core [24] pruning strategy, which enables us to deal with transactions with no tag assignment. Table 1 shows the key statistics of the cleaned datasets.

Table 1. Dataset description

Dataset	Delicious	Citeulike	Bibsonomy
Number of users *m*	177	132	125
Number of items *n*	210	225	388
Number of selected/total tags *l*	116/2251	128/1584	149/2305
Number of transactions p	4093	3300	4383
Density level *p/mn* (%)	11.01	11.11	9.04
Avg. number of items per user	23.12	25.00	35.06
Avg. number of users per item	19.49	14.67	11.30
Avg. frequency of selected tags	99.97	34.20	69.18
Number of items per user	>=10	>=10	>=10
Number of users per item	>=10	>=10	>=8
Frequency of selected tags	>=20	>=10	>=20

*A transaction indicates a user saving an item, no matter how many tags are assigned. We intentionally chose large pruning thresholds to attain relatively smaller datasets, so that we can focus most of our attention on the algorithm itself. According to our experiments, the size of the cleaned datasets has little impact on the relative strengths of the tested algorithms.

5.2 Evaluation Protocols and Metrics

We performed a 5-fold cross validation to test all the implemented algorithms. In each fold, we chose 80% of the saved items of each user for training and withheld the rest for test. In the prediction phase, we recommended the top 5, 10, ⋯, 25 items for each user and then compared them with the items in the test set. The evaluation metrics adopted in our experiment were the commonly used ones for ranked list prediction, namely, precision, recall, F-measure, and rankscore [14], as summarized in Table 2.

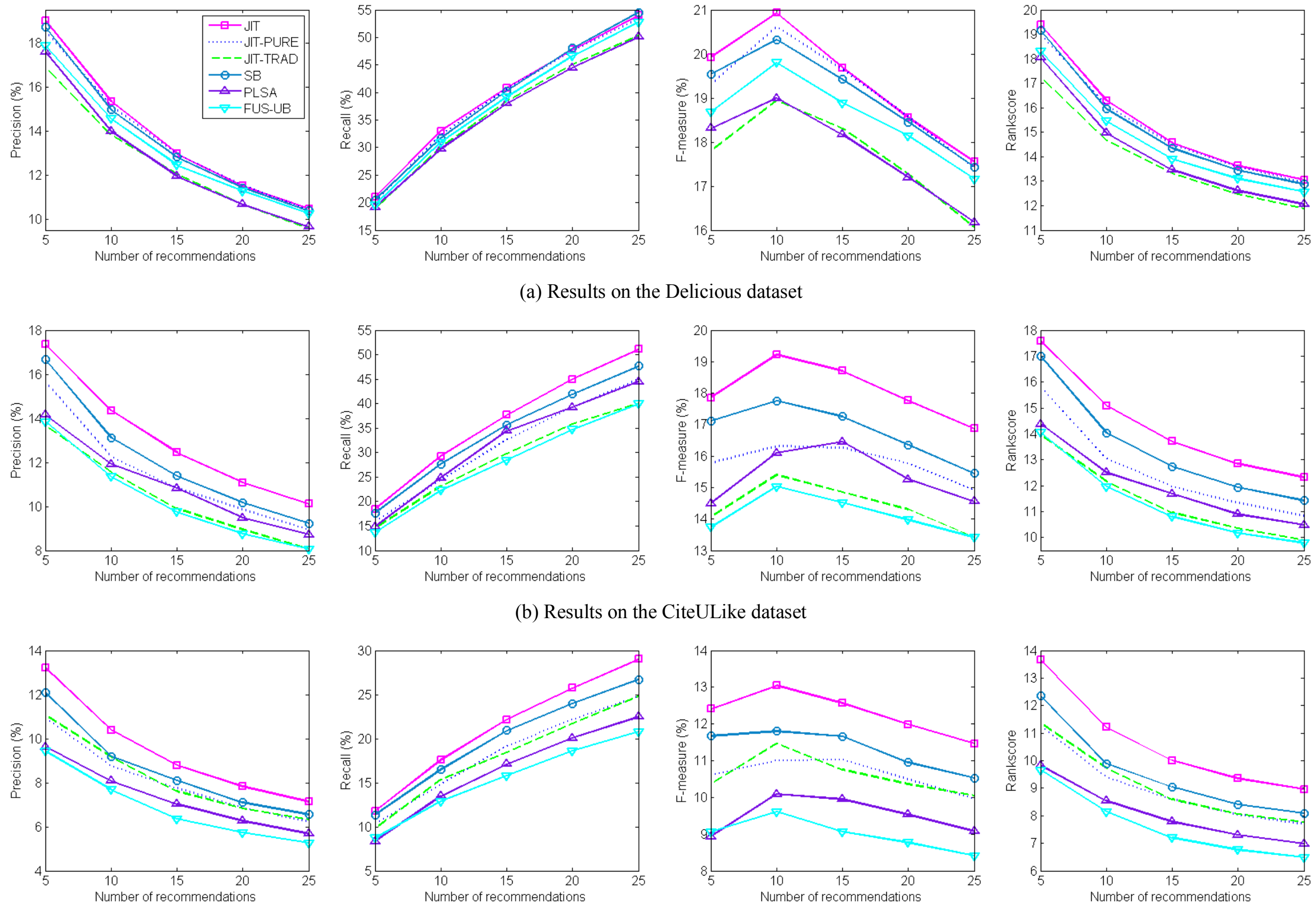
Table 2. Evaluation metrics

Metric	Formula
Precision	N_{hit}/N_{rec}
Recall	N_{hit}/N_{test}
F-measure	$\frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$
Rankscore	$100 \sum_j \frac{q_j}{2^{(j-1)/(h-1)}} / \sum_j \frac{1}{2^{(j-1)/(h-1)}}$

[5] http://www.citeulike.org/faq/data.adp

[6] http://www.kde.cs.uni-kassel.de/bibsonomy/dumps

[7] http://snowball.tartarus.org/



(a) Results on the Delicious dataset

(b) Results on the CiteULike dataset

(c) Results on the Bibsonomy dataset

Figure 5. Experiment results on three different datasets.

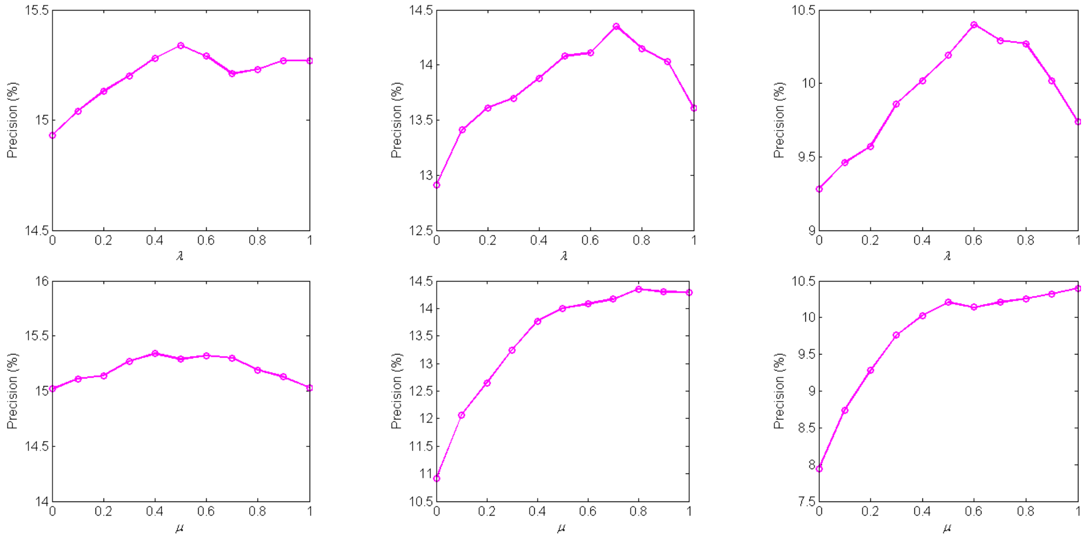
We use N_{rec} to denote the total number of recommendations, N_{hit} the number of correct recommendations, and N_{test} the number of items in the test set. We use j to indicate the index of an item in the predicted ranked list, h the viewing half-life of users (usually set to 10). Parameter q_j assumes value 1 if the j-th item is actually saved by the active user, and 0 otherwise. Note that all the metrics are calculated for individual users and then averaged over all users over all runs.

5.3 Benchmark Algorithms

We compared our joint item-tag recommendation approach with a variety of existing tag-based recommendation algorithms. One of the benchmarks is the state-of-art memory-based method, the fusion (FUS) method [10], which profiles users/items with extended item-tag/user-tag vector. Since this method is actually a fusion of a user-based (FUS-UB) and an item-based (FUS-IB) method, for fairness, we only compared our approach with the user-based part of it in that our approach can also be generally viewed as a user-based method. In the next subsection, we'll show the fusion result of our approach, as opposite to FUS-UB, with the FUS-IB method. We also compared our approach with state-of-art model-based methods, including the subject-based (SB) [14] and the probabilistic Latent Semantic Analysis (PLSA) algorithms [3].

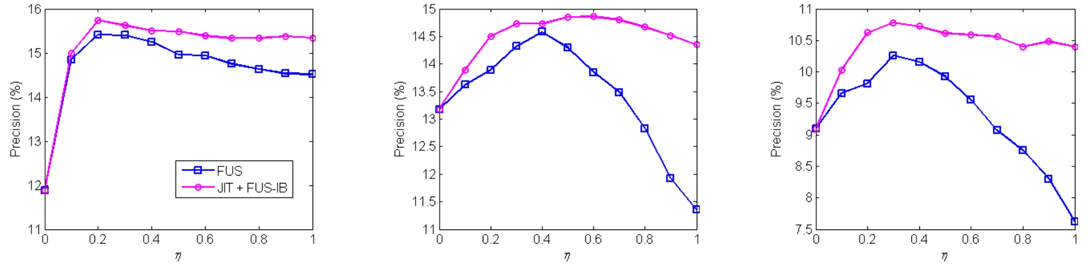
As to our joint item-tag approach, we implemented several variants of it to show the effectiveness of our unified user profiling scheme and joint item-tag recommendation framework. The first one is complete joint item-tag recommendation (JIT), which follows strictly the steps we have discussed before. The second one JIT-TRAD is the same with the first one except that the lower dimensional user representation is extracted from the traditional user-item-tag tensor. The third one is the same as the first one except that it uses the pure item recommendation result in the recommended profile matrix directly for item recommendation. The third approach is actually not a joint approach, but we still abbreviate it as "JIT-PURE" for clarity.

We tuned the parameters of all the control algorithms to their optimum before reporting the final result. The parameters involved in our joint recommendation approach are: d – the dimensionality of the three subspaces, α – the rescaling parameter of tag weights, s – the size of user neighborhood, λ – a parameter reflecting the tradeoff between a user's potential and current interests in tags, μ – a parameter adjusting the relative importance of $p(i,t)^{pure}$ as compared to $p(i,t)^{real}$. For simplicity, we uniformly set α to 5 and d to 50 in our experiments. In fact, according to our experiments, the performance of our approach is very stable with the variation of these two parameters in a large range. For the remaining parameters, we set them as follows: for



(a) Delicious (b) CiteULike (c) Bibsonomy

Figure 6. Sensitivity of the major parameters in joint item-tag recommendation.



(a) Delicious (b) CiteULike (c) Bibsonomy

Figure 7. Comparison of JIT approach with FUS-UB after fusion with FUS-IB.

the Delicious dataset, s = 50, λ = 0.5, μ = 0.4; for the CiteULike dataset, s = 30, λ = 0.7, μ = 0.8; for the Bibsonomy dataset, s = 70, λ = 0.6, μ = 1.0.

5.4 Results

The results with respect to four different evaluation metrics on the three datasets are shown in Figure 5. As can be seen clearly, the proposed joint item-tag recommendation approach achieved the best results consistently on all evaluation metrics across all datasets. In particular, we found that the superiority of our approach is more prominent on the Bibsonomy and CiteULike datasets, of which the tag quality is much poorer than that of Delicious. This observation indicates that our approach is more robust than existing methods due to its capability to make use of complete information embedded in tagging data.

The complete joint recommendation approach outperformed both the JIT-TRAD and JIT-PURE variants significantly, demonstrating the importance of both the unified user profiling scheme and joint item-tag recommendation framework in delivering quality recommendation. In addition, the superior performance of JIT-PURE over FUS-UB across all datasets also proves the advantage of our unified user profiling scheme over traditional user profiling methods.

The subject-based method, which uses the B-Use and B-Associated relationships for matrix factorization, performed consistently better than the JIT-TRAD variant, which utilizes the T-Annotate association for tensor factorization, indicating that binary information is more important than ternary information in our specific application. A critical problem that limits the practical utility of the fine-grained ternary information is sparsity.

In addition, one may easily notice that the relative strength of some methods, especially PLSA and JIT-TRAD, varied largely across datasets. This finding demonstrates the different natures of the three datasets and in turn proves the general applicability of our approach.

5.5 Sensitivity Analysis

We also performed a sensitivity analysis of the major parameters that govern our approach. Apart from the common parameter that controls the neighborhood size of user-based methods, the major parameters of our approach that need to be considered are two fusion parameters λ and μ. The impacts of both parameters are shown in Figure 6. Since the figures for all four evaluation metrics are basically the same, we only report the precision curves for top-10 item recommendation here.

As can be seen, the performance of our algorithm drops significantly as λ approaches either 0 or 1. This demonstrates that to gain a truthful representation of a user's interest in tags, it is of great importance to consider both her current and potential interests. Meanwhile, we can see that the recommended tag profile generally outweigh the initial tag profile in an optimized setting. This can explain the relatively poor performance of the traditional tag-based method, as compared to our approach, to some extent in that they only consider a user's initial tag profile.

It is interesting to see that the performance of our approach was very stable with the variation of μ on all datasets as μ approached 1. This observation demonstrates that the role of the joint real item-tag recommendation result is very minor in our overall joint recommendation framework, which verifies our assumption in subsection 4.2 that the joint real item-tag recommendation result is easily disturbed by noise. Moreover, this important fact enables us to derive a much more efficient simplified version of our joint recommendation approach without losing much of precision. More specifically, the most time-consuming step of our approach, i.e., generating joint real item-tag recommendations, can be saved by setting μ to 1 in equation (7). The computational complexity of our approach will be discussed in detail in section 6.

5.6 Fusion with Item-based Method

To show the superiority of our approach over the FUS method [10], we also compared the performance of our approach, after being fused with an item-based approach, with the complete fusion method. Note that our unified user profiling scheme can be easily adapted to profile items to underlie an item-based approach. For simplicity, however, we mix our approach directly with the item-based part of the FUS method (FUS-IB). Supposing that we weight the user-based method with η and the item-based method with $1-\eta$, the fused results are shown in Figure 7.

It can be seen clearly that our approach is still superior to the FUS method after fusion with the FUS-IB approach. While the best result of FUS method is comparable to the performance of standalone JIT approach, the tuning of the FUS method will be very difficult in practice in that it is very sensitive to η. On the other hand, our approach benefits from the item-based method in a much smoother manner, making the tuning of the fusion parameter much easier.

6. Discussion

A critical issue that concerns most recommender systems is the *cold start* problem. Due to our splitting protocol, we do not encounter any new user or item in our experiments. However, it does not imply that our approach is not able to deal with new users or items. As discussed in subsection 3.4, we have provided a principled method to fold-in new users upon request, provided that the structure of the profile tensor does not change significantly. Thus, we can update the profile tensor and re-compute the Tucker model only when a certain amount of new transactions, which might involve some new users and new items, have arrived at the system. After the batched update, the newly appeared items will also become recommendable in our approach.

Since our approach involves some tensor operations, another issue that needs to be addressed is how to efficiently store and decompose the profile tensor. In fact, given that the user profile tensor is very sparse, we can store and factorize it efficiently using the Matlab Tensor Toolbox[8], which has been reported to be able to deal with very large sparse tensors, e.g., a 1K×1K×1.1K×200 tensor with 5.39 million nonzero entries, in [20].

As the Tucker model can be updated offline, the computational cost of our approach to generate item recommendations for a user mainly comes from three steps: user similarity computation – equation (2), joint item-tag recommendation – equation (3),(4), (5), (7), and marginalizing for item recommendation – equation (8). The time complexity of the similarity computation step is $O(md^2)$, and the time complexity of the joint item-tag recommendation step is $O\big(m(n+1)(l+1)\big)+O(l+1)+O\,(nl)+O(nl)=O(mnl)$. Lastly, the time complexity of marginalizing is $O(nl)$. Considering that d is usually much small that either n or l, the overall computational complexity of our approach is $O(mnl)$. This is of an order higher than $O(mn)$ of the user-based and $O(n^2)$ of the item-based methods. However, as we discussed in subsection 5.5, the joint real item-tag recommendations can be saved in that it is of very minor use in our approach. The time complexity of equation (3) will then drop to $O(m(n+l))$, and the overall computational complexity of our approach will be $O(md^2+mn+ml+nl)$. Given that both l and d can be selected to be very small as compared to m and n, the time complexity of our approach will approximately be $O(mn)$, the same as that of a general user-based method, as the number of users and items become very large.

While we have focused on investigating the problem of item recommendation under our joint recommendation framework in this paper, our framework is also capable of making other types of recommendations, such as tag recommendation for a given item, and paired <item,tag> recommendation for a given user. In fact, given that our unified profiling scheme can be easily extended to profile items and tags to sustain other types of joint recommendations, our joint recommendation framework is suitable to almost any type of recommendation tasks that might arise in the social tagging context. The key advantage of our joint item-tag recommendation framework is that it is able to make full use of tag recommendation, item recommendation, and joint real item-tag recommendation results in a systematic manner. Furthermore, presenting the recommendation results in a joint form also enables us to make effective use of the *crowd wisdom* reflected through the overall item-tag co-occurrence matrix.

7. CONCLUDING REMARKS

Tag-based recommendation holds the potential to deliver substantial quality improvement by exploiting additional information generated by users. In this paper, we have presented a joint item-tag recommendation framework, which is able to utilize complete information in the tagging data, to produce high-quality item recommendations. Specifically, we have made the following main contributions.

1) We have proposed an integrated structure of social tagging systems, which captures all types of co-occurrence information appearing in tagging data.

2) Based on the proposed social tagging structure, we have devised a novel matrix-based user profiling scheme to make full use of all the available information in tagging data.

3) To obtain informative low dimensional representation of users, we presented a tensor decomposition approach to compressing the profile tensor constructed by profiles of all users. In addition, we discussed a systematic method to represent old users and fold-in new users in the low dimensional space based on the tensor factorization result.

[8] http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/

4) Supported by the matrix-based user profiling scheme, we have developed a joint recommendation framework that makes joint item-tag recommendations to users, with the tags explaining the users' topical interest in each item. We have shown empirically that significantly better result could be achieved when these joint recommendations are projected to the item space for final item recommendation.

Our work opens up several avenues for future research. One direction is to find some theoretical foundations for the unified user profiling scheme and to develop more systematic weighting methods for it. Another direction is to explore effective alternative approaches to measuring the association between items and tags. Finally, our framework can be extended to other recommendation tasks, e.g., tag recommendation for a given item.

8. ACKNOWLEDGMENTS

The authors wish to acknowledge research support from the CAS (2F07C01), NNSFC (70890084, 60875049, and 60621001), and MOST (2006AA010106).

9. REFERENCES

[1] J. Peng and D. Zeng. *Topic-based web page recommendation using tags*. In *Proceedings of the 2009 IEEE international conference on Intelligence and security informatics*, 2009.

[2] Z.-K. Zhang, T. Zhou, and Y.-C. Zhang. *Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs*. Physica A: Statistical Mechanics and its Applications, 389(1): 179-186, 2010.

[3] R. Wetzker, W. Umbrath, and A. Said. *A hybrid approach to item recommendation in folksonomies*. In *Proceedings of the WSDM '09 Workshop on Exploiting Semantic Annotations in Information Retrieval*, 2009.

[4] Y. Zhen, W. Li, and D. Yeung. *TagiCoFi: tag informed collaborative filtering*. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, 2009.

[5] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos. *A Unified Framework for Providing Recommendations in Social Tagging Systems Based on Ternary Semantic Analysis*. IEEE Trans. on Knowl. and Data Eng., 22(2): 179-192, 2010.

[6] L.D. Lathauwer, B.D. Moor, and J. Vandewalle. *A Multilinear Singular Value Decomposition*. SIAM J. Matrix Anal. Appl., 21(4): 1253-1278, 2000.

[7] J.L. Herlocker, J.A. Konstan, and J. Riedl. *Explaining collaborative filtering recommendations*. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, 2000.

[8] D. Zeng and H. Li. *How Useful Are Tags? -- An Empirical Analysis of Collaborative Tagging for Web Page Recommendation*. In *Proceedings of the IEEE ISI 2008 PAISI, PACCF, and SOCO international workshops on Intelligence and Security Informatics*, 2008.

[9] S. Zhao, N. Du, A. Nauerz, X. Zhang, Q. Yuan, and R. Fu. *Improved recommendation based on collaborative tagging behaviors*. In *Proceedings of the 13th international conference on Intelligent user interfaces*, 2008.

[10] K.H.L. Tso-Sutter, L.B. Marinho, and L. Schmidt-Thieme. *Tag-aware recommender systems by fusion of collaborative filtering algorithms*. In *Proceedings of the ACM symposium on Applied computing*, 2008.

[11] S. Givon and V. Lavrenko. *Predicting social-tags for cold start book recommendations*. In *Proceedings of the third ACM conference on Recommender systems*, 2009.

[12] D. Parra and P. Brusilovsky. *Collaborative filtering for social tagging systems: an experiment with CiteULike*. In *Proceedings of the third ACM conference on Recommender systems*, 2009.

[13] S. Sen, J. Vig, and J. Riedl. *Tagommenders: connecting users to items through tags*. In *Proceedings of the 18th international conference on World wide web*, 2009.

[14] J. Peng and D. Zeng. *Exploring Information Hidden in Tags: A Subject-based Item Recommendation Approach*. In *Proceedings of 19th Workshop on Information Technologies and Systems*, 2009.

[15] H. Halpin, V. Robu, and H. Shepherd. *The complex dynamics of collaborative tagging*. In *Proceedings of the 16th international conference on World Wide Web*, 2007.

[16] S. Rendle, L.B. Marinho, A. Nanopoulos, and L. Schmidt-Thieme. *Learning optimal ranking with tensor factorization for tag recommendation*. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009.

[17] T.G. Kolda and B.W. Bader. *Tensor Decompositions and Applications*. SIAM Rev., 51(3): 455-500, 2009.

[18] J. Peng and D. Zeng. *Tag-based Smoothing for Item Recommendation*. In *Proceedings of the 2010 IEEE International Conference on Service Operations and Logistics*, 2010.

[19] Y. Koren. *The BellKor Solution to the Netflix Grand Prize*. 2009; Available from: http://www.netflixprize.com/assets/GrandPrize2009_BPC_BellKor.pdf.

[20] T.G. Kolda and J. Sun. *Scalable Tensor Decompositions for Multi-aspect Data Mining*. In *Proceedings of the Eighth IEEE International Conference on Data Mining*, 2008.

[21] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. *Indexing by latent semantic analysis*. Journal of the American Society for Information Science, 41(6): 391-407, 1990.

[22] P.-N. Tan, V. Kumar, and J. Srivastava. *Selecting the right interestingness measure for association patterns*. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002.

[23] B. Kitts, D. Freed, and M. Vrieze. *Cross-sell: a fast promotion-tunable customer-item recommendation method based on conditionally independent probabilities*. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2000.

[24] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. *Tag recommendations in social bookmarking systems*. AI Communications, 21(4): 231-247, 2008.