2011

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

Jing Peng
University of Pennsylvania

Daniel Zeng

Huimin Zhao

Fei-yue Wang

Recommended Citation
Peng, J., Zeng, D., Zhao, H., & Wang, F. (2011). Collaborative Filtering in Social Tagging Systems Based on Joint Item-Tag Recommendations. *CIKM '10 Proceedings of the 19th ACM international conference on Information and knowledge management*, 809-818.
http://dx.doi.org/10.1145/1871437.1871541

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

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.

Keywords
Collaborative filtering, social tagging, tagging structure, joint item-tag recommendation, design science

Disciplines
Other Social and Behavioral Sciences | Sociology

Joint Item-Tag Recommendation Framework for Collaborative Filtering in Social Tagging Systems

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 is becoming an increasingly important mechanism for organizing and discovering information on the Web. Effective tag-based recommendation of information items is one of the key technologies contributing to the success 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 item recommendation method. While most existing methods either implicitly or explicitly assume a simple tripartite graph structure, in this paper, we propose a comprehensive data model to capture all types of co-occurrence information in the social tagging context. Based on this data model, we further propose a unified user profiling scheme to make full use of all available information. Finally, supported by this user profile, we propose a framework for collaborative filtering in social tagging systems. In this 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 (or tag) space for final item (or tag) recommendations. Empirical evaluation using real-world data demonstrates the utility of our proposed approach.

Keywords — Collaborative filtering, social tagging, tagging structure, joint item-tag recommendation, design science

INTRODUCTION

In recent years, social tagging has been gaining wide-spread popularity in a variety of Web applications, ranging from social bookmarking sites (e.g., delicious.com and citeulike.org), movie rating sites (e.g., movielens.org), to E-commerce sites (e.g., amazon.com). Such social tagging systems encourage users to annotate Web resources (items) of their interest with descriptive tags. These tags not only allow users to conveniently revisit and retrieve previously-visited items, but also enable them to search and explore what other users are interested in. Such simple tag-enabled capabilities have been reported to deliver financial returns to firms through knowledge worker productivity gain. According to a blog on IBM DevelopWorks, the deployed Enterprise Tagging Service of IBM saves an average person 12 seconds across over 286,000 searches each week, resulting in a $4.6 million total saving per year[1].

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 and contribute to the success of many business models. Netflix claims that approximately 60% of its rentals originate from recommendations[2]. Amazon says that 35% of its product sales result from recommendations[3]. As even slight improvement on recommendation quality may translate into significant profit growth for such businesses, significant efforts (e.g., the Netflix Prize[4]) have been devoted to algorithmically improving recommendation quality.

Many commercial recommender systems in use now (e.g., amazon.com and taobao.com) have a tagging component allowing users to not only indicate whether or not (or how much) they like a particular item but also attach several tags of their choice to describe the item or their experience with the item. In a way, these tags can be viewed as qualitative multidimensional ratings on the item. They greatly enrich the information contained in users' feedback on items.

---

[1] https://www.ibm.com/developerworks/mydeveloperworks/blogs/rawn/entry/enterprise_tagging_service_social_software?lang=en

[2] http://ir.netflix.com/annuals.cfm (Netflix 2006 annual report)

[3] http://venturebeat.com/2006/12/10/aggregate-knowledge-raises-5m-from-kleiner-on-a-roll/

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

The additional information provided by tags gives rise to opportunities as well as challenges to advance the state of the art in item recommendation to the next level.

Research on item recommendation leveraging tagging information is just emerging. While a few methods have been proposed, our literature review shows that there are still major gaps in the area. First, most existing methods either implicitly (e.g., (Peng et al. 2009b; Wetzker et al. 2009; Zhen et al. 2009)) or explicitly (e.g., (Zhang et al. 2010)) assume a simple tripartite graph structure for social tagging systems. As we will show, tripartite graph is not an adequate representation of the information structure of social tagging systems. Such an inadequate representation poses an intrinsic limitation on the potential of these methods. Recently, a tensor-based approach (Symeonidis et al. 2010)—using a user-item-tag tensor representation—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 user-item-tag tensor of prediction utility, obtained through High Order Singular Value Decomposition (HOSVD) (Lathauwer et al. 2000), is usually not sparse.

Furthermore, most previous methods (Jäschke et al. 2008; Rendle et al. 2009; Tso-Sutter et al. 2008; Zhang et al. 2010) focus on recommendations of either items or tags. However, items and tags indeed co-exist in real tagging activities, with tags indicating the specific topics covered by a target item that attract a user. Consequently, the correctness of the resulting recommendations from previous tag-based methods, which do not pinpoint why a user may select an item, cannot be well-justified as the recommendations are not guaranteed to fall into the set of a user's interested topics (tags).

In striving to bridge these gaps, we take the design science approach (Hevner et al. 2004) and develop a novel approach to effective tag-based recommendation of information items in a social tagging system. First, we propose a comprehensive data model, which captures all types of co-occurrence information in the social tagging context, to address the intrinsic information inadequacy of most existing methods (e.g., (Peng et al. 2009b; Wetzker et al. 2009; Zhang et al. 2010; Zhen et al. 2009)). We then propose a unified user profiling scheme, which fully utilizes all available information in this data model but avoids the prohibitive computational and spatial complexity of the tensor-based approach (Symeonidis et al. 2010). Finally, 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 joint item-tag recommendation framework, in contrast to previous methods that focus on recommendations of either items or tags (Jäschke et al. 2008; Rendle et al. 2009; Tso-Sutter et al. 2008; Zhang et al. 2010). Our empirical evaluation using real-world data shows clear advantages of our proposed approach over previous methods. Furthermore, while our idea of joint item-tag recommendation is mainly motivated by the intention to yield high-quality item recommendation, our approach also applies to the task of tag recommendation.

The rest of the paper is organized as follows. We start with a review of the literature on collaborative filtering, as well as tag and item recommendations in social tagging systems. We then present our integrated data model of social tagging, unified user profiling scheme based on this data model, and joint item-tag recommendation framework. Next, we report on our empirical evaluation. We then discuss potential applications and implications of our approach. Finally, we highlight our contributions and discuss prominent future research directions.

RELATED WORK

Collaborative filtering (CF) is one of the most widely adopted and successful recommendation approaches (Huang et al. 2007). In this section, we first briefly review the literature of standard CF that works with user-item interaction/rating information only and then review recent work on tag and item recommendations in social tagging systems.

Standard CF Methods

Most CF methods fall into two categories: memory-based and model-based. In memory-based methods, all training examples are stored and subsequent predications are made for a target user using information from similar users or items. A crucial step of a memory-based method is similarity calculation. According to the type of similarity used, memory-based methods can be further classified as user-based (Breese et al. 1998; Paul et al. 1994) or item-based (Badrul et al. 2001; Linden et al. 2003). User-based methods glean preference information from similar users, whereas item-based methods generate recommendations based on similar items. Pearson Correlation Coefficient (Paul et al. 1994) and Vector Space Similarity (cosine similarity) (Breese et al. 1998) are two commonly used methods for similarity computation, for

either user similarity or item similarity. Recently, a number of fusion methods (Hao et al. 2007; Jun et al. 2006; Jun et al. 2008) have been developed to take advantage of both user-based and item-based methods. Memory-based methods have grown in popularity because of their simplicity, but they may suffer from low scalability (Hofmann 2004).

In contrast to memory-based methods, training data are used to train (induce) a predictive model in a model-based method. Predications are made thereafter using this trained model. In early CF research, two alternative probabilistic models were proposed: clustering and Bayesian network (Breese et al. 1998). A limitation of these models is that every user is restricted to be within a single class while a user may indeed have multiple interests. Some recent methods, such as the latent semantic model (Hofmann 2004), the personality diagnosis model (Pennock et al. 2000), and the flexible mixture model (Si et al. 2003), attempt to capture multiple interests of a user by classifying the user into multiple clusters. Model-based methods usually scale very well, but the model estimation and updating can be very time-consuming.

Tag Recommendation

The research on tag recommendation dates back to only 2006, when Hotho et al. (Hotho et al. 2006) attempted to explore the structure of folksonomies for search and ranking. Their proposed method named FolkRank, an adaptation of PageRank in social tagging systems, was shown to be able to outperform several baselines, including popularity-based methods and classical CF methods, in a follow-up study (Jäschke et al. 2008). Despite its fairly good recommendation quality, the utility of FolkRank is largely limited in real-world scenarios as a result of its long prediction time. In response to this problem, Song et al. (Song et al. 2008) proposed a Gaussian process classification framework for fast tag recommendations. Nevertheless, user information is not presented in this approach and the generated recommendations are not personalized.

The great success of matrix decomposition methods in the Netflix Prize competition has led to the development of a number of factorization models for recommender systems. Symeonidis et al. (Symeonidis et al. 2010) proposed a low-rank user-item-tag tensor reconstruction method to predict the missing entries in tagging data. In contrast to the traditional factorization model that aims at minimizing the squared reconstruction error, Rendle and Marinho (Rendle et al. 2009) developed a tensor factorization model to maximize a ranking statistic observed in tagging

data. Later, this method was further improved into a more effective pairwise interaction tensor factorization (PITF) method (Rendle et al. 2010), which only takes the binary interactions among users, items, and tags into consideration.

Apart from these increasingly complex tag recommendation models, Gemmell et al. (Gemmell et al. 2010) recently found that a linear combination of simple tag recommendation methods, such as popularity-based recommendation and memory-based CF, is able to deliver comparable or even superior performance in comparison with state-of-the-art tag recommenders. This study shows that successful tag recommendations should be intrinsically multi-sided, while focusing on any single aspect might have difficulty in achieving ideal recommendation quality.

Item Recommendation in Social Tagging Systems

While much of existing CF research in social tagging systems deals with tag recommendation, a few methods have been proposed for tag-based item recommendation. A straightforward method of this kind is to use tags in computing user or item similarity. Zeng and Li (Zeng et al. 2008) introduced two variants of standard user- and item-based CF methods, which calculate user and item similarities based on TF-IDF weighted tag vectors. Zhao et al. (Zhao et al. 2008) proposed to compute the similarity of two users based on the semantic distance of their tag sets on common items they have selected. Tso-Sutter et al. (Tso-Sutter et al. 2008) extended the item vectors for user profiles and the 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 (Givon et al. 2009; Parra et al. 2009; Sen et al. 2009). Markines et al. (Markines et al. 2009) systematically evaluated various similarity measures in the social tagging context.

A few recent studies aimed at further use of tagging information for tag-based item recommendation. The topic-based method (Peng et al. 2009b) exploits tag information in a probabilistic framework, viewing each tag as an indicator of a topic and then estimating, by aggregating the transition probabilities through all tags, the probability of a user selecting an item. Zhen et al. (Zhen et al. 2009) used users' tag vectors to regularize the user-item matrix factorization results by making sure that the similarity between two users' latent feature vectors are correlated with the tag sets of the two users. The subject-based method (Peng et al. 2009a) 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 has selected (or might select) an item. Zhang et al. (Zhang et al. 2010) proposed a diffusion method, which generates recommendations based on the fusion of information diffusion on user-item and item-tag bipartite graphs. Peng et al. (Peng et al. 2010b) proposed a method that iteratively propagates users' preference information between the item space and the tag space.

INTEGRATED DATA MODEL OF SOCIAL TAGGING

A common feature of most existing tag-based item recommendation 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 illustrated in Figure 1 (Halpin et al. 2007). The methods of Zeng and Li (Zeng et al. 2008) and Tso-Sutter et al. (Tso-Sutter et al. 2008) assume a user-item-tag representation for their item-based methods and tag-user-item representation for their user-based methods. The method of Zhen et al. (Zhen et al. 2009) 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 (Peng et al. 2009b) takes a user-tag-item representation, as user interest in items is essentially a result of user-tag and tag-item relationships. The diffusion method (Zhang et al. 2010) rests on an explicitly stated user-item-tag representation and restricts information propagation to be between user-item and item-tag bipartite graphs.

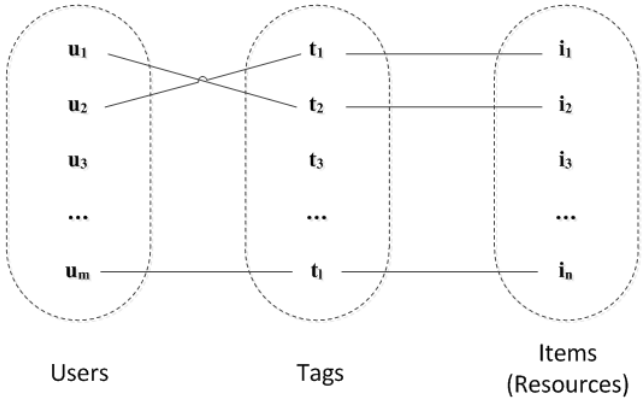


Figure 1. Tripartite graph structure of social tagging.

Nevertheless, such tripartite graphs capture only two of three binary associations among the three entities (i.e., user, item, and tag), and the ternary association among the entities, which does exist in reality and cannot be decomposed into multiple binary associations, is completely lost.

We believe that it is essential for one to gain some deep insight into the underlying structure of social tagging systems before devising a comprehensive and effective data representation method catering to item recommendation in the social tagging context.

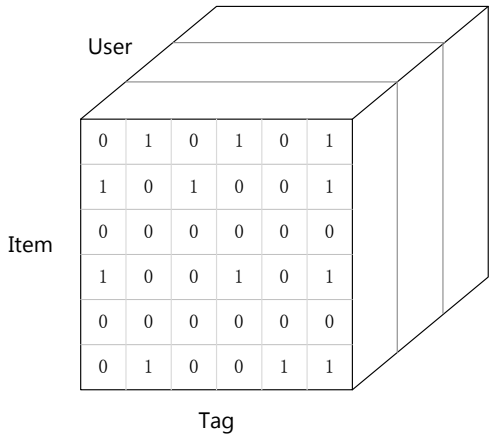


Figure 2. Illustrative tensor representation of tagging data.

There are two recent studies (Rendle et al. 2009; Symeonidis et al. 2010) that investigate the ternary association among users, items, and tags for more effective tag recommendation using tensor decomposition techniques (an illustrative tensor representation of tagging data is shown in Figure 2). However, they cannot be readily adapted to item recommendation due to the difference in nature between tag recommendation and item recommendation. Tag recommendation aims at predicting the use of tags by a given user on a given item, with two entities predefined, whereas item recommendation aims at predicting the selection (saving, bookmarking, purchasing, etc.) of items, with only the user specified. Generally, item recommendation is more difficult than tag recommendation in that less information is provided about the subject to receive recommendations. Although Symeonidis et al. (Symeonidis et al. 2010) argued that their approach was also applicable to item recommendation, they were actually recommending items to a given user with a given tag, which is not a typical item recommendation setting.

Moreover, since existing tensor factorization approaches, including PARAFAC Decomposition (Kolda et al. 2009), HOSVD (Lathauwer et al. 2000), and Tucker Decomposition (Kolda et al. 2009), 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 social tagging

context, a user-item-tag tensor is flattened into user×(item, tag), item×(user, tag), and tag×(user, item) matrices in practical tensor operations. Two users will not be considered correlated at all unless they have annotated 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. Nevertheless, the value of the two-dimensional information, which underlies most previous tag-based item recommendation methods, has been well-justified in the literature. Another problem with this tensor representation is that it is unable to capture user-item interactions when there is no tag assignment. Thus, the user-item-tag tensor adopted in recent studies (Rendle et al. 2009; Symeonidis et al. 2010) is not an ideal representation of the tagging data either.

As discussed above, most of the existing methods for tag-based item recommendation either explicitly or implicitly assume 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 the three entities, say <user, item> or <user, tag>, which underlies most of the existing research, is actually ignored in the tensor operations. To gain a more comprehensive interpretation of social tagging systems, we propose a data model of social tagging behaviors (shown in Figure 3) that captures all possible co-occurrence information among the three entities.

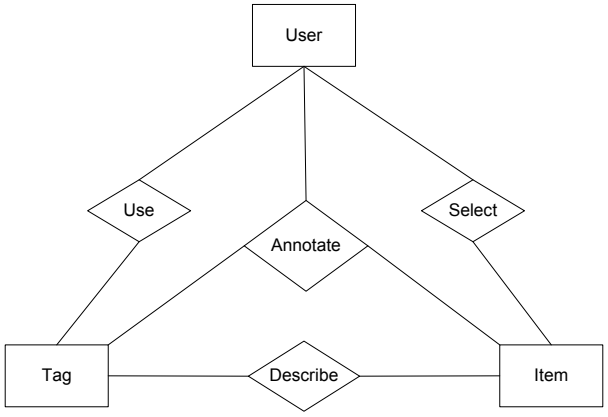


Figure 3. Proposed integrated data model of social tagging.

The fundamental difference of the proposed data model from the commonly-adopted tripartite graph structures is that user, item, and tag are treated equally as peer entities and have direct associations with each another. In addition to the binary relationships, a ternary association

named "Annotate" is added to capture the simultaneous interactions among the three entities. Note that although the binary "Use" (B-Use) and "Describe" (B-Describe) relationships can be derived from the ternary "Annotate" (T-Annotate) relationship, we explicitly present them in our data model because the binary association information is 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 ternary relationship because some users may select items without assigning any tag.

UNIFIED USER PROFILING

Based on our proposed data model (Figure 3), we further propose a unified user profiling scheme. In this section, we present this user profiling scheme and associated subsequent dimensionality reduction and similarity computation. We adopt the following notation in the rest of the paper. Matrices are denoted by boldface capital letters, e.g., $\mathbf{P}$. Tensors are denoted by boldface Calligraphy letters, e.g. $\boldsymbol{\mathcal{X}}$. Scalars are denoted by italic lowercase letters, e.g., λ and p_{ij}. Indices typically range from 1 to their italic capital version, e.g., $k = 1, \dots, K$. Let $\boldsymbol{u} = \{u_1, u_2, \dots, u_m\}$ be a set of users, $\boldsymbol{i} = \{i_1, i_2, \dots, i_n\}$ be a set of items, and $\boldsymbol{t} = \{t_1, t_2, \dots, t_l\}$ be a set of tags. The probability of observing an arbitrary user, item, or tag is represented by $p(u)$, $p(i)$, or $p(t)$, respectively. The joint probability of observing an arbitrary combination of item and tag is denoted $p(i, t)$.

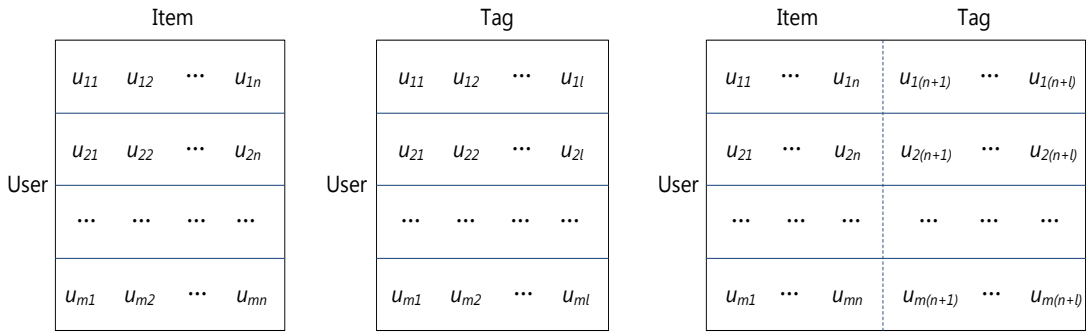
User Profiling Scheme

Our integrated data model for social tagging indicates that the similarity between two users may be judged based on the following criteria:

1) Users that have selected common items may be considered to be similar—B-Select similarity.
2) Users that have used common tags may be considered to be similar—B-Use similarity.
3) Users that have annotated the same item with common tags may be considered to be similar—T-Annotate similarity.

So far, there are three major methods to profile a user in the social tagging context (see Figure 4): i) use the item vector of the user's historical records (Zhao et al. 2008; Zhen et al.

2009); ii) use a tag vector, where each element reflects the frequency the user uses a tag (Zeng et al. 2008); and iii) use an extended 0-1 valued item-plus-tag vector (Tso-Sutter et al. 2008). Existing profiling schemes only capture user similarity on one or two of the criteria we identified above. None of them takes advantage of all these three types of similarity in a unified manner.



(a) item vector (b) tag vector (c) extended item-tag vector

Figure 4. Traditional user profiling methods.

Among the three types of similarity, T-Annotate similarity is the most reliable as it requires users to agree on both items and tags. Considering that the historical tagging records of a user naturally form an $n \times l$ 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.

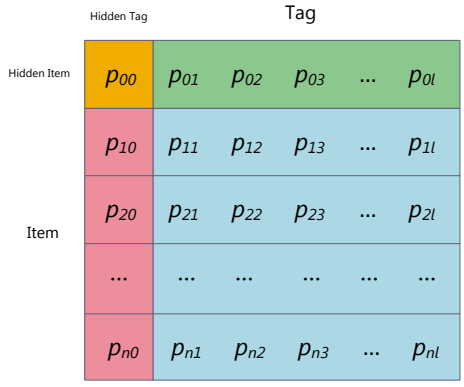


Figure 5. Proposed unified user profiling scheme.

To ensure that the denser B-Select and B-Use similarities are also incorporated in the user profile, we extend the tagging matrix as shown in Figure 5. For each item, we assume that there

exists a Hidden Tag, and whenever the item is selected, this Hidden Tag is 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 selected the same item, even if they have assigned completely different sets of tags to the item. Likewise, we can assume the existence of 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 many cases, individual users are unable to assign complete and accurate tags when selecting an item due to a variety of reasons (e.g., laziness, use of non-descriptive tags for personal annotation only, and spelling errors), 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 may be some common interests between them, and such common interests are substantiated as a Hidden Item in our profiling scheme.

Apparently, the Hidden Item row (without the corner entry) corresponds to the traditional tag profile, whereas the Hidden Tag column corresponds to the traditional item profile. Note that the B-Describe relationship, reflected through the overall item-tag co-occurrence matrix, is not used in our profiling scheme as it is independent of individual users. This type of information pertains to the *wisdom from the crowd* and will be used in our proposed joint item-tag recommendation framework later.

We weight the elements in our user profile matrix (Figure 5) as follows.

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

$$p_{i0} = \begin{cases} 1, & \text{if user selected the } i\text{th item} \\ 0, & \text{otherwise} \end{cases} \tag{1b}$$

$$p_{0j} = \sum_i p_{ij} \,/\, \sum_i p_{i0} \tag{1c}$$

$$p_{00} = 0 \tag{1d}$$

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

For the bottom-right T-Annotate sub matrix, it is straightforward to weight each entry using a 0-1 scheme as in most existing approaches, with 1 indicating a tag assignment on an item. However, as Peng et al. (Peng et al. 2010a) argued, it is inadequate to treat all tags equally

without considering the ranking order of each tag in the set of tags co-assigned to an item as well as the size of this tag set. After moderate adaption to the weighting formula proposed by Peng et al. (Peng et al. 2010a), equation (1a) is used to weight the T-Annotate entries, where k is the ranking index of the tag and K is the total number of tags assigned to this item. α is an empirical parameter used to rescale the weighting of tags. In formula (1a), the larger the number of tags co-assigned and the larger the ranking index, the smaller the weight a tag gets. When α is sufficiently large, our weighting scheme will degenerate to the traditional method that treats all tags equally.

The weight of a Hidden Tag is uniformly set to 1 (equation 1b). The underlying reason is that we believe that the Hidden Tag (or the item itself) is more important than any user-specified tag and presume that it is ranked at index zero (i.e., $k = 0$), making (1b) a natural choice following (1a). In addition, we assume that the importance of each Hidden Item in a user's profile is proportional to the average weight of its corresponding tag over all item selecting activities of this user. Specifically, we use equation (1c) to weight Hidden Items. Note that in this weighting scheme, all the entries are restricted to be within the range of [0,1].

Dimensionality Reduction and Similarity Computation

While our profiling scheme has integrated all available information, it is attained at the price of extending the traditional item or tag vector profile into 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 been shown to be an effective dimensionality reduction method in the field of recommender systems (Koren 2009; Peng et al. 2009a; Takács et al. 2008). Nevertheless, matrix techniques do not apply to our approach in that the profile matrices of all users actually constitute a three-order tensor, which can be interpreted as an enhanced version of the traditional user-item-tag tensor as it is able to capture binary B-Select and B-Use similarities through the introduction of Hidden Tags and Hidden Items. We therefore employ the Tucker Decomposition method (Kolda et al. 2009; Kolda et al. 2008), which deals with tensor data, to extract informative lower-dimensional representation of users. Below, we introduce some operations related to tensors. For more details, please refer to (Kolda et al. 2009).

Mode-*n* matrix product The mode-*n* 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,

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

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 squared reconstruction error and can be solved efficiently using the Alternative Least Squares algorithm (Kolda et al. 2008). As an alternative generalization of the two-dimensional Singular Value Decomposition, Tucker decomposition gives a more strict approximation (Kolda et al. 2009) of the original tensor as compared to HOSVD (Lathauwer et al. 2000; Symeonidis et al. 2010). In our application, we decompose the three-order user-item-tag profile tensor $\boldsymbol{\mathcal{T}}$ as:

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

where $\mathbf{U}^{(\mathrm{user})}$, $\mathbf{U}^{(\mathrm{item})}$, and $\mathbf{U}^{(\mathrm{tag})}$are factor matrices spanning the user, item, and tag subspaces of size $m \times d$, $(n+1) \times d$, and $(l+1) \times d$, respectively, supposing that the dimensions of all the subspaces are uniformly *d*. Figure 6 illustrates the Tucker decomposition on our user-item-tag profile tensor. Kolda et al. (Kolda et al. 2008) suggested to use $\mathbf{U}^{(\mathrm{user})}$ directly as a user representation for clustering. However, $\mathbf{U}^{(\mathrm{user})}$ is actually a basis matrix that spans the user subspace, rather than a feature matrix that holds the coordinates of users in the item×tag subspace. Hence, we propose to represent users with $\boldsymbol{\mathcal{G}} \times_1 \mathbf{U}^{(\mathrm{user})}$, 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 (Deerwester et al. 1990), which factors a document-term matrix $\mathbf{X}$ as $\mathbf{X} = \mathbf{USV}^{\mathrm{T}}$ and then uses $\mathbf{US}$ to represent documents.

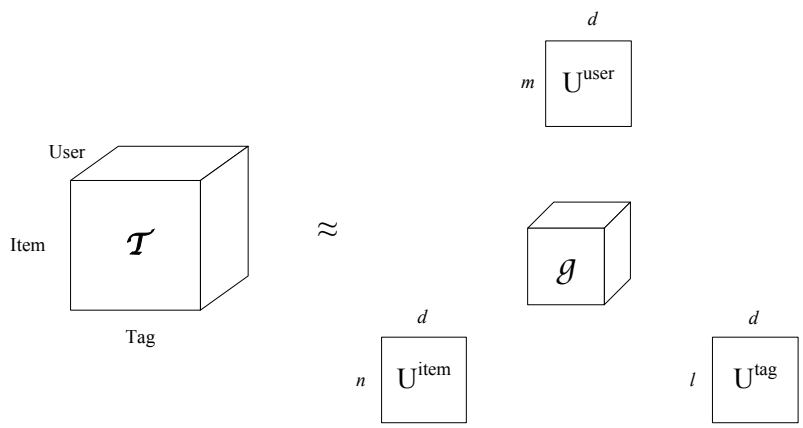


Figure 6. Tucker decomposition on user-item-tag profile tensor

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 significant 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 derivation.

In principle, we can update the profile tensor and re-compute the Tucker model only after many new transactions, which might involve some new users and new items, have arrived at the system. After each batched update, the newly-appeared items will also become recommendable in our approach.

Finally, after the lower-dimensional representation of users has been obtained, we can compute the cosine similarity between users a and b as follows:

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

where **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.

JOINT ITEM-TAG RECOMMENDATION FRAMEWORK

It is straightforward to make item recommendations directly following the traditional user-based method after the user similarity is computed. However, as we argued earlier, pure item recommendation faces some essential difficulties in providing quality recommendations due to its inability to capture users' explicit interests in the target items. To address this problem, we propose to present the recommendations to users in a more traceable joint item-tag form before generating the final item recommendations. Moreover, such joint recommendations can also be easily adopted to generate personalized tag recommendations for each user to annotate items.

Joint Item-Tag Recommendation

To gain more insights into the reason why a user might select 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 already profiled each user with an item×tag matrix, it is straightforward to recommend an integrated profile matrix to a user following equation (3). In fact, this joint recommendation process is basically the same as that of the traditional user-based approach other than 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.

$$\mathbf{R}^{\mathrm{a}} = \frac{\sum_{b \neq a} s_{ab} * \mathbf{P}^{\mathrm{b}}}{\sum_{b \neq a} s_{ab}} \tag{3}$$

where s_{ab} represents the similarity between users a and b, and $\mathbf{P}^{\mathrm{b}}$ represents the profile matrix of user b.

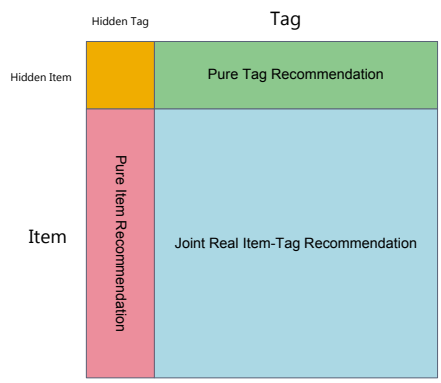


Figure 7. Joint item-tag recommendation results.

As shown in Figure 7, the recommended profile matrix for each user consists of four blocks. It is easy to understand that the bottom-right sub matrix 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 holds the pure tag recommendation result.

The above joint recommendation can be viewed as a smoothing process to predict missing entries and refine nonzero entries in a user's profile. Therefore, the joint recommendation result for each user actually represents a smoothed profile. So far, we have not considered a user's initial profile in the recommendation process yet. Before generating the final item recommendations, we need to synthesize the recommended user profile with the user's initial profile first. To make this synthesis more flexible, we fuse a user's profile on hidden item, hidden tag, and real item-tag blocks separately as follows:

$$p(t|u)^{pure'} = \gamma * p(t|u)^{pure} + (1-\gamma) * p(t|u)^{init} \ (0 \le \gamma \le 1) \tag{4}$$

$$p(i|u)^{pure'} = \lambda * p(i|u)^{pure} + (1-\lambda) * p(i|u)^{init} \ (0 \le \lambda \le 1) \tag{5}$$

$$p(i,t|u)^{real'} = \delta * p(i,t|u)^{real} + (1-\delta) * p(i,t|u)^{init} \ (0 \le \delta \le 1) \tag{6}$$

where $p(t|u)^{pure}$ and $p(t|u)^{init}$ denote the normalized values of tag t in the Hidden Item rows of the recommended and initial profile matrices, respectively. Equation (4) can be interpreted as a tradeoff between a user's potential (recommended) and current (initial) interests in tags. Likewise, $p(i|u)^{pure}$ and $p(i|u)^{init}$ denote the normalized values of item i in the Hidden Tag columns of the recommended and initial profile matrices, respectively. $p(i,t|u)^{init}$ denotes a

user's initial joint profile on real items and tags, whereas $p(i,t|u)^{real}$ denotes the recommended joint profile on real items and tags. Note that all the involved entries are normalized into quantities of probabilities before the fusion. Specifically, we have normalized the Hidden Item row to unit row sum, the Hidden Tag column to unit column sum, and the real item-tag sub matrix to unit matrix sum.

Recommendation Synthesis

As discussed earlier, item-only recommendations do not take into account whether users are actually interested in the covered topics of the recommended items. As such, resulting recommendations might be erroneous as they might not fall into a user's set of interested topics (tags) at all. On the other hand, joint item-tag recommendations explicitly consider a user's possible interest in each item with respect to each tag (topic). In the meanwhile, tag-only recommendations also provide valuable clues indicating the user's interest in each tag. In this sense, an ideal item recommendation approach should be one that is able to make effective use of all these three types of recommendations in a systematic manner, as illustrated in Figure 8.

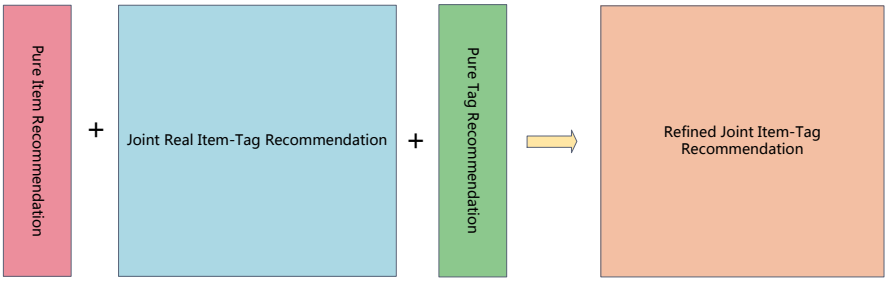


Figure 8. Synthesis of joint item-tag recommendation results.

We believe that joint recommendation holds the potential to deliver better 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 to generate 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:

$$p(i,t|u)^{pure} \propto p(i|u)^{pure} \cdot p(t|u)^{pure} \cdot assoc(i,t) \tag{7}$$

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

There are many ways (Tan et al. 2002) to measure the association between two variables. We use the Lift (or Mutual Affinity) (Kitts et al. 2000; Tan et al. 2002) defined as $Lift(i,t) = p(i,t)/(p(i)p(t))$. As can be seen, the log form of Lift actually represents the mutual information between an item and a tag, indicating the amount of uncertainty reduction in determining a tag after observing an item, or vice versa. The Lift association between an item and a tag can be pre-computed based on the overall item-tag co-occurrence matrix as follows:

$$assoc(i,t) = \frac{p(i,t)}{p(i)p(t)} = \frac{freq(i,t)/\sum_{i,t} freq(i,t)}{freq(i)freq(t)/(\sum_i freq(i) \sum_t freq(t))} \tag{8}$$

where $freq(i,t)$ represents the co-occurrence frequency of item *i* with tag *t* in the training data.

In fact, using equation (8) to compute the item-tag association enables us to take advantage of the B-Describe information, the only relationship in our profiling scheme (Figure 3) that remains unused until now. The use of B-Describe information can be interpreted as correlating or explaining items with tags based on the *wisdom from the crowd*, as the item-tag co-occurrence matrix represents tagging preference 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.,

$$p(i,t|u)^{final} = \mu p(i,t|u)^{pure} + (1-\mu)p(i,t|u)^{real} \ (0 \le \mu \le 1). \tag{9}$$

Item recommendation

After the synthesized joint item-tag recommendations have been obtained, it is straightforward to make item recommendations (we use a superscript "final" to differentiate final item recommendations from pure item recommendations) by marginalizing the final joint recommendation result, i.e.,

$$p(i|u)^{final} = \sum_{t \in \boldsymbol{t}} p(i,t|u)^{final}. \tag{10}$$

Thereafter, we can recommend the top-*N* items with the highest probabilities to the active user. Moreover, we can select several relevant tags based on the joint item-tag recommendation results to explain each item recommendation if so desired.

Tag Recommendation

In addition to item recommendation, our joint item-tag recommendation approach also applies to the task of tag recommendation. In fact, after the final joint-form recommendations have been obtained, we can recommend tags directly to each item upon being selected by the active user. Nevertheless, such tag recommendations might not be very ideal as the joint recommendations typically are very sparse and do not cover all possible items and tags. Alternatively, we propose to make tag recommendations following the item-based CF approach on top of the final joint item-tag recommendations, which enables us to make full use of a user's tagging preference on items other than the target item (including all selected items as well as unselected items predicted to be of high interest to the user). As compared to traditional item-based tag recommendation methods, the advantage of our approach is that the item-tag profile of each user is largely smoothed through joint item-tag recommendation.

When the recommender solicits tag suggestions from a neighboring item of the target item, the similarity between the two items is not the only factor that will impact the weight of the neighboring item. The active user's real interest in this neighboring item should also be taken into account as we aim at personalized tag recommendation. Generally speaking, the more interested a user is in a neighboring item, the more personalized tag suggestions the recommender can get from this item. As a result, we propose to set the weight of each neighboring item as the product of its similarity to the target item and its predicted probability to be selected by the active user, the latter of which can be obtained from pure item recommendation results. Formally, the item-based tag recommendation under the joint item-tag recommendation framework is as follows

$$p(t|u,i_a) \propto \sum_{i_b \in I_{ua}} sim(i_a,i_b)p(i_b|u)p(t|u,i_b) \tag{11}$$

where $p(t|u,i)$ represents the probability of user u using tag t for annotation when selecting item i. $p(t|u,i)$ can be easily derived from the final joint item-tag recommendation result of user u. I_{ua} represents the set (with a given size of s) of user u's neighboring items for item a.

From equation 11, we can see that two differences of our approach from classical item-based tag recommendation methods are the consideration of selecting probability $p(i_b|u)$ (enabled via pure item recommendation) and the smoothing of $p(t|u,i)$ (attained through joint recommendation). As will be shown through empirical evaluation later, both modifications contribute to significant quality improvement on tag recommendation.

Recommendation Procedure

The proposed joint item-tag recommendation approach mainly involves five steps: user profile construction, dimensionality reduction, user similarity computation, joint item-tag recommendation, and final item or tag recommendation on top of the synthesized joint-form recommendations. Figure 9 summarizes the procedure of our item/tag recommendation approach under the joint recommendation framework.

Algorithm: Joint Item-Tag Recommendation

Input Data: training records

Input Parameters: the weighting factor – α, the dimensionality of user/item/tag subspaces – *d*, the size of item neighborhood – *s*, fusion parameters – λ, γ, δ, and μ, and the number of items/tags to be recommended – *N*

1. Construct a profile matrix for each user based on the training data following equations (1a) ~ (1d).
2. Perform Tucker Decomposition on the profile tensor made up of all users' profile matrices.
3. Attain lower-dimensional representation of users, namely $\mathcal{G} \times_1 \mathbf{U}^{(\text{user})}$, based on the Tucker Decomposition results.
4. Compute cosine similarities between users based on the lower-dimensional feature matrices of users following equation (2).
5. Recommend an integrated profile matrix to each user following equation (3) and fuse it with the user's original profile to attain a smoothed profile following equations (4), (5), and (6).
6. Synthesize the three different types of recommendation results within the recommended profile matrix to obtain the final joint item-tag recommendation result following equations (7), (8), and (9).

Item Recommendation (Output: *N* items recommended to each user)

a) Project the final joint item-tag recommendation results to the item space for item recommendation following equation (10).

b) Recommend the top-*N* items with the highest probabilities to the active user.

Tag recommendation (Output: *N* tags recommended for each test <user, item> pair)

a) Perform item-based tag recommendation on top of the synthesized joint recommendation results following equation (11).

b) For each test item of the active user (i.e., each test <user, item> pair), recommend the top-*N* tags with the highest probabilities to the user.

Figure 9. Recommendation procedure.

Complexity

Since our approach involves some tensor operations, an 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. For example, Kolda et al. (Kolda et al. 2008) reported that the Matlab Tensor Toolbox[5] is able to deal with very large sparse tensors (e.g., a 1K×1K×1.1K×200 tensor with 5.39 million nonzero entries out of more than 200 billion possible entries.)

As the Tucker model, users' lower-dimensional representations, and inter-user similarities can be updated offline, the online computational cost of our approach to generate item recommendations for a user mainly comes from three steps: joint item-tag recommendation (equation (3), (4), (5), and (6)), recommendation synthesis (equation (7) and (9)), and recommendation generation (equation (10)). The time complexity of the joint item-tag recommendation step is $O(mnl)$ and further reduces to $O(mn + ml)$ when the joint real item-tag recommendation is avoided (i.e., when setting δ to zero in equation (6)). The time complexity of recommendation synthesis and recommendation generation is $O(nl)$. All together, these lead to a total computational complexity of $O(mnl)$ for the full version of our approach and $O(mn + ml + nl)$ for the simplified version. As compared to traditional item recommendation methods, the time complexity of the simplified version of our approach is basically on the same order as that of the user-based ($O(mn)$) and item-based ($O(n^2)$) item recommendation methods. The time complexity of the simplified version will further approximate to $O(mn)$ when the numbers of users and items become very large, as l (the number of effective tags) converges very quickly as the size of the dataset grows.

The online time complexity of performing tag recommendation under our joint recommendation framework is quite different from that for item recommendation as the smoothed profile of each user can be stored and updated offline in batch for efficient tag recommendation. Therefore, in an online setting, the time complexity of our approach to make tag recommendations for one <user, item> pair is the same as that of the classical user-based and item-based tag recommendation methods, namely $O(sl)$. While the online time complexity of

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

our approach is basically the same as (actually a constant lower than) that of the state-of-the-art PITF method ($O(dl)$, where d represents the dimensionality of the factors)) (Rendle et al. 2010), the offline time complexity of our approach ($O(dmnl)$ at worst and can be greatly reduced when the Tucker decomposition is performed in a sparse form) is at least one order lower than that of PITF (roughly $O(dmnl^2)$).

EMPIRICAL EVALUATION

We have evaluated our approach using data from real-world social tagging systems. The results attest the utility of our approach. In this section, we present our evaluation.

Data

We used three datasets. One dataset was crawled from Delicious, the largest social bookmarking site. The collected dataset consists of bookmarking data of 5000 users dated from 7/1/2008 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. Another dataset is a snapshot of the CiteULike database[6] downloaded on 1/21/2010. We collected transactions that took place in year 2009. The last dataset is the widely-used Bibsonomy dataset[7], and what we used is the 2009-07-01 snapshot. The Bibsonomy dataset contains bookmarks for both bibliographies and general Web resources, of which the part for general Web resources was used.

During data preprocessing, we pruned the datasets based on their characteristics. We iteratively removed users that had selected very few items and items that had been selected by very few users (see Table 1 for the pruning thresholds) until less than 0.5% of items were unqualified. Note that if a user or item was removed, all transactions related to this user or item were discarded. In addition, we stemmed each tag using the Snowball stemmer[8] (Porter 2) to alleviate the effect of word variations. We also discarded tags that had appeared very few (<10) times. Table 1 lists some statistics about the pruned datasets.

---

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

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

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

Table 1. Dataset description

Dataset		Delicious	CiteULike	Bibsonomy
Number of users *m*		548	338	296
Number of items *n*		1080	392	1499
Number of tags *l*		1,439	222	576
Number of user-item interactions *p*		28,591	6,031	11,768
Number of <user, item, tag> triplets		167,072	16,598	53,239
Density level *p*/*mn*(%)		4.83	4.55	2.65
Average number of items per user		52.17	17.84	39.76
Average number of users per item		26.47	15.39	7.85
Average number of tags per user-item interaction		5.84	2.75	4.52
Average frequency of use per tag		78.47	57.90	37.49
Pruning thresholds	Number of items per user	>=15	>=5	>=5
	Number of users per item	>=15	>=10	>=5
	Frequency of selected tags	>=10	>=10	>=10

Evaluation Procedure and Metrics

We used a five-fold cross validation to test all the implemented methods. Specifically, we randomly divided the selected items of each user into five parts (folds). We used four folds of each user's items (including the tags attached to them) as training data and the remaining fold for testing. We repeated this five times, using a different fold for testing each time. For the item recommendation task, we recommended the top 1, 2, 3, 4, or 5 items to the active user and then compared them with the items in the test set of the active user. Similarly, for the tag recommendation task, we recommended the top 1, 2, 3, 4, or 5 tags to each test item of the active user and then compared them with the real tags the active user had assigned to the test item.

The performance metrics we adopted in our evaluation are precision, recall, F-measure, and Rankscore (summarized in Table 2). Precision reflects the proportion of correct recommendations (N_{hit}) within the list of item or tag recommendations (N_{rec}). Recall, on the other hand, represents the proportion of test items or tags (N_{test}) that are successfully identified by the recommender (N_{hit}). F-measure reflects a tradeoff between precision and recall. In our experiment, precision and recall were given the same weight in computing F-measure. In addition to inspecting the number of correct recommendations, we were also interested in the order of items and tags in the recommendation list. Precision, recall, and F-measure do not distinguish between two recommenders that generate the same number of correct recommendations but rank the correct recommendations differently in the list of recommended items/tags. Apparently, an ideal recommender is one that ranks all the correct recommendations

before the incorrect ones. To highlight such a difference regarding the ranking order of recommended items/tags, we used the Rankscore (Breese et al. 1998) to measure the ability of recommenders in ranking important items/tags before less important ones.

To calculate Rankscore (Breese et al. 1998), we first compute a correctness score q_j for each recommendation in the list of items/tags. For the j-th item recommendation, its correctness score q_j is 1 if it falls into the test item set and 0 otherwise. For the j-th tag recommendation, q_j is given by equation (1) (a nonzero value) if it falls into the real tag set and 0 otherwise. We then take the ranking position of each recommendation into consideration and weight its correctness score with a rank-related factor $2^{-(j-1)/(h-1)}$. In this factor, j is the ranking position of the recommended items/tags and h is the viewing half-life (typically set to 5 or 10, indicating that the possibility of a recommendation being noticed by the user decays to one half of the first recommendation when j equals h). Finally, by summing up these correctness scores weighted by ranking positions, we get an overall Rankscore for the list of recommendations. The overall Rankscore can then be normalized by the ideal overall Rankscore (when the recommended items/tags are exactly the same as those of the test set, including their ranking order) and rescaled to the range of 0 to 100.

Table 2. Performance metrics

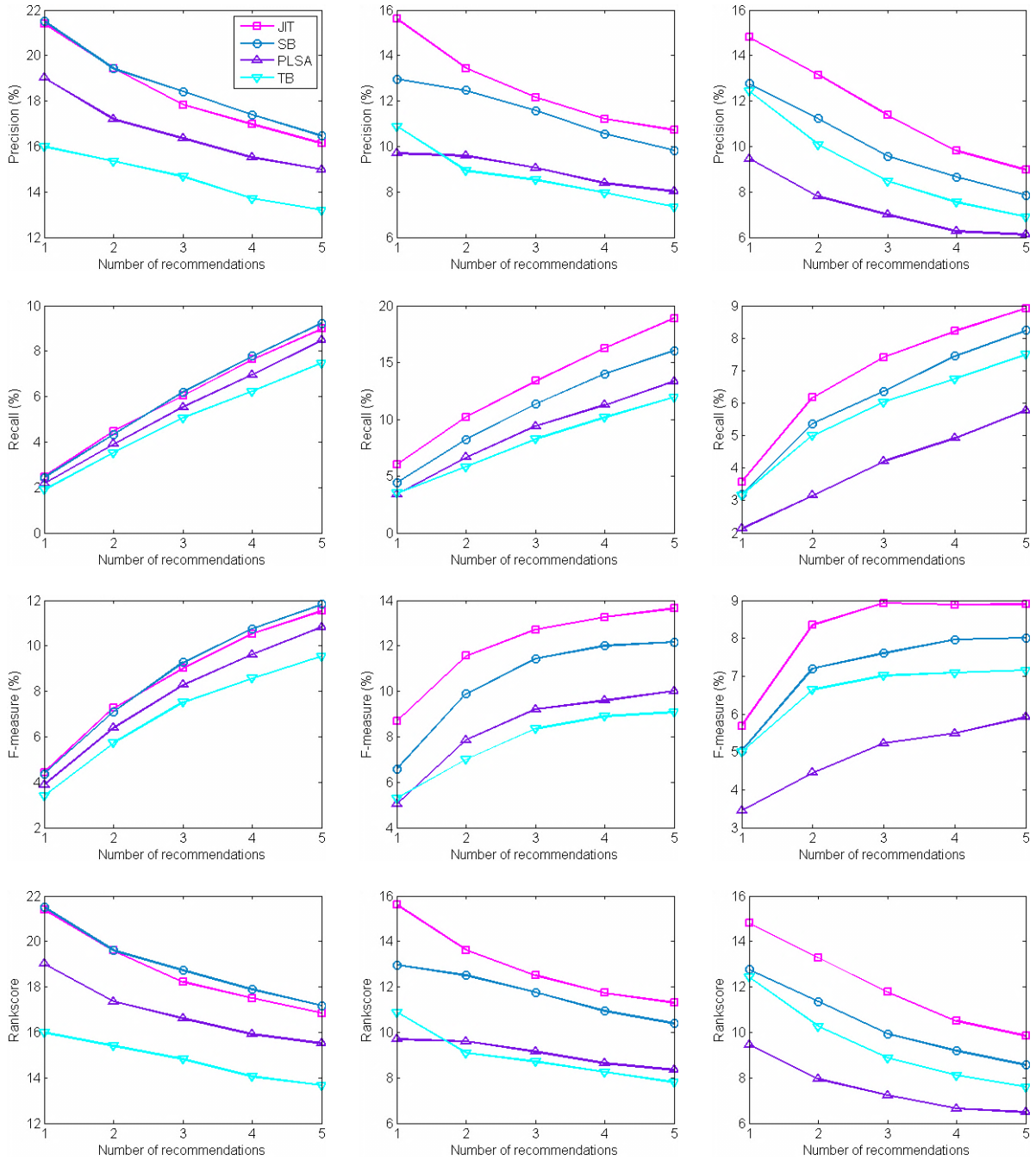
Metric	Formula
Precision	N_{hit}/N_{rec}
Recall	N_{hit}/N_{test}
F-measure	$\frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$
Rankscore	$100 \sum_j \frac{q_j}{2^{(j-1)/(h-1)}} / \sum_j \frac{q_j^{ideal}}{2^{(j-1)/(h-1)}}$

Note that for item recommendation, all the metrics were calculated for individual users and then averaged over all users over all runs. For tag recommendation, all the metrics were calculated for individual items of the active user and then averaged over all items and all users.

Item Recommendation

As to the item recommendation task, we compared our joint item-tag (JIT) recommendation approach with several state-of-the-art tag-aware recommendation methods, including the topic-

based (TB) method (Peng et al. 2009b), the subject-based (SB) method (Peng et al. 2009a), and the probabilistic Latent Semantic Analysis (PLSA) method (Wetzker et al. 2009). The results with regard to the four performance metrics on the three datasets are summarized in Figure 10.



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

Figure 10. Item recommendation results.

Our proposed JIT approach clearly outperformed all other methods on the CiteULike and

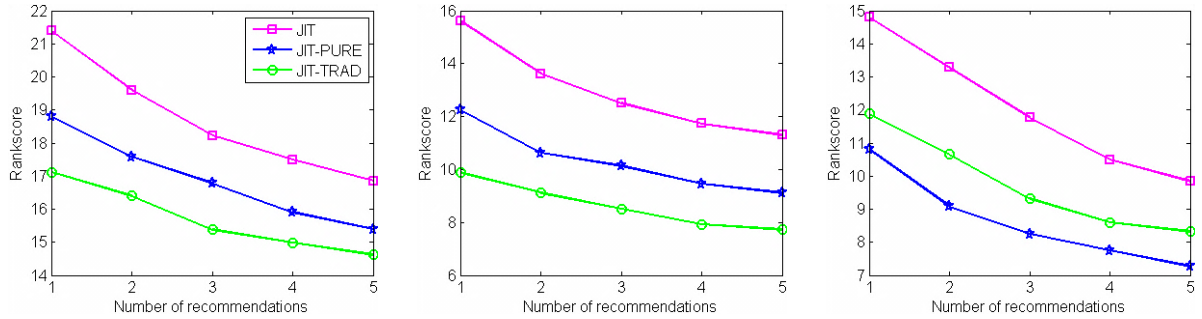
Bibsonomy datasets with regard to every performance metric and under every top-N setting. This demonstrates the advantage of our approach—by exploiting all available information—over the other methods in yielding high quality recommendations.

On the Delicious dataset, the performance of our approach is comparable to that of the subject-based recommendation method. However, the relative performance gain of our approach over the worst performing method on the Delicious dataset is not as large as that on the other two datasets. Taking top-1 item recommendation for example, the relative performance gain of our approach over the worst performing method is about 61% on the CiteULike dataset (namely JIT over PLSA) and 56% on the Bibsonomy dataset (namely JIT over PLSA), but only 34% on the Delicious dataset (namely JIT over TB). This difference can be expected as the Delicious dataset has the highest data quality among the three, no matter in terms of dataset size, data density, tagging intensity (i.e., average number of tags per item and average frequency of use per tag), or tagging variety (i.e., number of tags used more than ten times). The key advantage of our approach is its ability to aggregate all kinds of available information for recommendation, while the high data quality of the Delicious dataset leaves us little room to get the most out of our approach's ability to deal with data sparsity.

Overall, the subject-based approach employing matrix factorization was just second to our JIT approach and significantly outperformed the other two. This finding is in line with the great success matrix factorization techniques have achieved in recommender systems over the past few years. A shortcoming of the subject-based approach is that it only considers the B-Use (or B-Select) and B-Describe relationships in the tagging data and neglects the other two relationships. The PLSA method, which trains a mixed latent model from the user-item interaction and item-tag interaction data, performed very poorly. One reason is that the proposed model in the PLSA method is only able to consider the B-Select and B-Describe relationships in the tagging data. Another possible reason is that the underlying model of the PLSA method is not a faithful representation of tagging data. The topic-based method, which simply considers the B-Use and B-Describe relationships for conditional probability estimation and does not take further moves, generally performed the worst among all methods. The poor performance of the topic-based method was mainly caused by the non-standardized usage of tags by individual users, which brings about a lot of noise to users' tag profiles.

Variants for Item Recommendation

To further investigate the effectiveness of our unified user profile scheme and joint item-tag recommendation framework, we also implemented two variants of our JIT approach. One of the variants (denoted JIT-TRAD) is the same as the complete JIT except that the lower-dimensional user representation is extracted from the traditional user-item-tag tensor. The other variant is the same as the complete JIT except that it uses the pure item recommendation result directly for item recommendation. Strictly speaking, the second variant is not a joint recommendation approach, but we still refer to it as "JIT-PURE" for consistency.



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

Figure 11. Comparison of the variants of JIT.

Figure 11 contrasts the complete JIT and the two variants with respect to Rankscore (results on the other metrics are similar and hence omitted). The complete JIT dominated the two variants remarkably across all datasets, indicating that both the unified user profiling scheme and the joint item-tag recommendation framework contribute to the delivery of high quality recommendations. These two elements corroborate and bring out the best utility of each other.

Interpretative Analysis

To further investigate why our proposed joint item-tag recommendation framework was able to outperform traditional item-only recommendation methods, we compared the detailed item recommendations of the complete JIT method and the item-only JIT-PURE method to individual users. Table 3 shows the recommendation results of both methods for a representative user in a joint item-tag form. To save space, we only show the top-10 tags the user is most interested in based on the pure tag recommendation result (i.e., $p(t|u)^{pure}$) and the top-10 items the user is most likely to select based on the pure item recommendation result (i.e., $p(i|u)^{pure}$). Without loss of generality, we refer to these tags/items as Tag/Item 1, 2, …, 10. The probabilities

of the given user's interest in these tags and items are shown in the second column on the left and the second row on the top, respectively. Each entry in the central area holds two values (separated by a slash). The first value in the <*j*, *k*> entry represents the joint probability of the specified user selecting Item *k* with Tag *j* (i.e., $p(i,t|u)^{final}$), and the second value represents the overall association between Item *k* and Tag *j* (i.e., $a \quad oc(i,t)$). An empty entry indicates that both values are zero. The bottom row holds the item recommendation scores of the complete JIT method (i.e., $p(i|u)^{final}$) and the ranking positions of these scores. Columns corresponding to correct item recommendations (i.e., items appearing in the test set of the given user) are highlighted.

Table 3. Recommendation result for a representative user

$p(i,t\|u)^{final}$/assoc(i,t) \ $p(i\|u)^{pure}$		Item 1	Item 2	* Item 3	Item 4	Item 5	Item 6	Item 7	Item 8	Item 9	* Item 10
$p(t\|u)^{pure}$		0.0117	0.0111	0.0107	0.0106	0.0103	0.0099	0.0097	0.0094	0.0093	0.0093
Tag 1	0.0974		.0136/.0010	.0199/.0016	.0024/.0002			.0010/.0001	.0009/.0001		.0184/.0014
Tag 2	0.0626	.0040/.0002	.0033/.0002	.0073/.0005	.0014/.0001	.0031/.0004	.0038/.0003	.0021/.0002	.0021/.0002	.0009/.0001	.0033/.0002
Tag 3	0.0579										
Tag 4	0.0576	.0027/.0007	.0018/.0005	.0008/.0002	.0015/.0006	.0005/.0003	.0012/.0005	.0018/.0009	.0015/.0007	.0016/.0009	.0005/.0001
Tag 5	0.0326	.0033/.0001									
Tag 6	0.0323		.0002/.0001		.0004/.0002				.0004/.0003	.0004/.0004	
Tag 7	0.0290					.0016/.0017				.0005/.0005	
Tag 8	0.0282		.0004/.0002		.0002/.0001		.0003/.0002	.0005/.0005	.0001/.0001	.0007/.0008	
Tag 9	0.0280		.0012/.0001						.0014/.0002		
Tag 10	0.0255		.0017/.0001							.0010/.0001	
JIT Score ($p(i\|u)^{final}$)/Rank		**0.0241/5**	**0.0282/2**	**0.0341/1**	**0.0134/9**	**0.0103/22**	**0.0131/12**	**0.0117/18**	**0.0133/11**	**0.0093/29**	**0.0271/4**

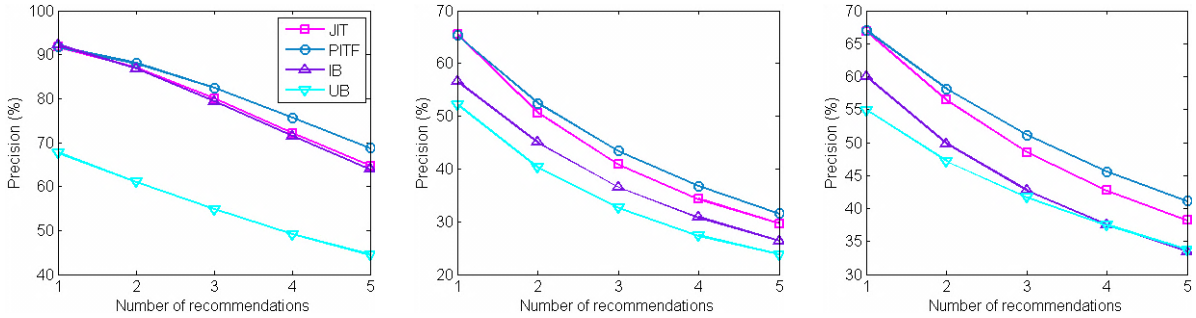
The rankings of every item, except Item 2, based on $p(i|u)^{pure}$ and $p(i|u)^{final}$ are different. In particular, the JIT-PURE method ranked the two correct item recommendations at the 3rd and 10th places, whereas the JIT method ranked them at the 1st and 4th places, respectively. These differences intuitively explain why our joint item-tag recommendation framework was able to produce better item recommendations. However, a question remains to be answered is how such differences come out. To answer this question, we take the contrast of Item 1 and 3 as an example to see how the rankings of items were re-arranged under the joint item-tag recommendation framework. As can be seen, although Item 1 got a higher pure item recommendation score than Item 3, Item 1 was substantially less suitable to the given user in terms of its association ($assoc(i,t)$) to the user's favorite topics (tags). In the proposed joint

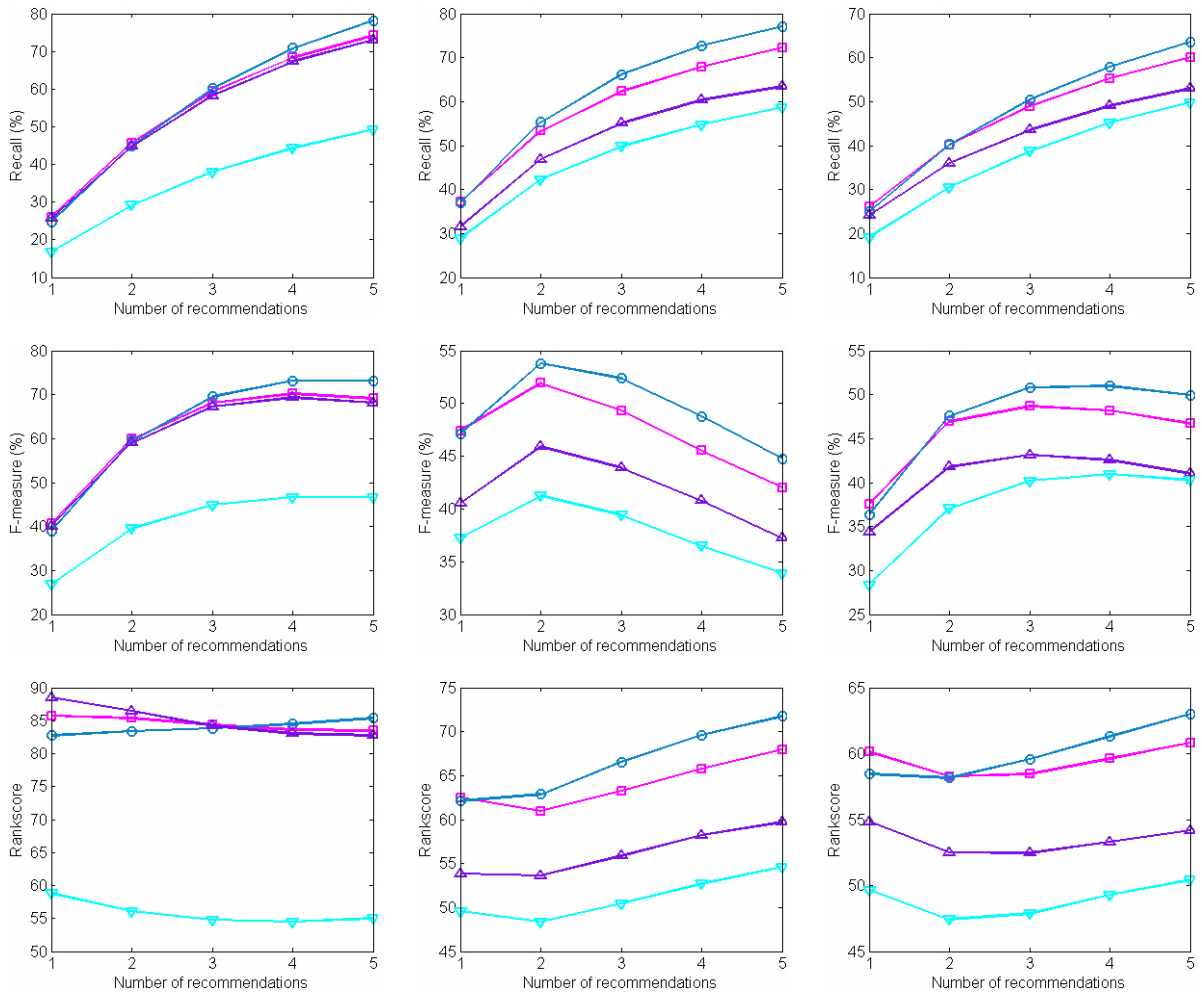
item-tag recommendation framework, we take both factors into consideration and compute the user's probability to select items and tags jointly. Since the probabilities of Item 1 to be recommended jointly with the user's interested tags ($p(i,t|u)^{final}$) are not as high as those of Item 3, it turned out that Item 3 was a better recommendation than Item 1 when these joint probabilities were summed over all tags to attain the final refined item recommendation score. The fact that the item-only JIT-PURE method ranked many less relevant items (in terms of their degree of matching to the user's topical interest) like Item 1 and 5 highly supports our earlier conjecture that the correctness of item-only recommendations may not be well-justified as they may fall out of users' topics of interest, hence demonstrating the necessity of performing joint item-tag recommendation to obtain more traceable item recommendations.

Moreover, it is straightforward to explain item recommendations with tags under the joint item-tag recommendation framework. For example, when Item 3 is recommended to the given user, we may use Tag 1, 2, and 4 to indicate the possible topics within this item that may attract the user.

Tag Recommendation

As to the tag recommendation task, we compared our approach with several most representative tag recommendation methods. First, we implemented the standard user-based (UB) and item-based (IB) tag recommendation methods as benchmarks. Gemmell et al. (Gemmell et al. 2010) showed that both simple methods may constitute important components of fused methods with exceptional performance. In addition, we compared our approach with the state-of-the-art tag recommendation method PITF (Rendle et al. 2010), which won the ECML/PKDD 2009 challenge. Figure 12 summarizes the results of the implemented methods.





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

Figure 12. Tag recommendation results.

Generally speaking, our joint recommendation approach outperformed the user-based and item-based methods and was very close to PITF. Although our approach did not outperform PITF, it should be noted that PITF trains a complex Bayesian ranking model for prediction based on the partial order ranking information among tags whereas our approach does not require the training of a prediction model. This difference makes our approach much more efficient and scalable than PITF. As discussed earlier, the offline time complexity of our approach is at least one order lower than that of PITF. The extremely high training cost of PITF may make it infeasible to obtain a well-tuned prediction model on large real-world datasets, while our approach provides a much cheaper and more applicable alternative with little performance loss.

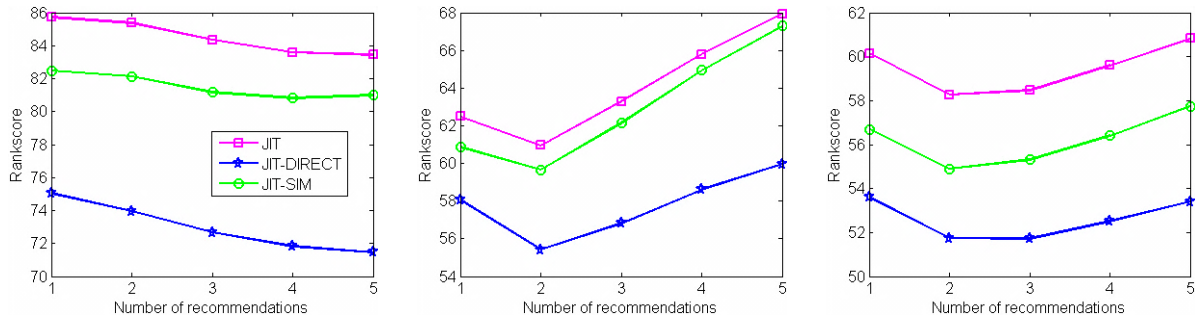
Furthermore, for top-1 tag recommendation, our approach was actually able to outperform PITF slightly in terms of F-measure and more evidently in terms of Rankscore. Considering that our approach is of magnitudes faster than PITF and much more applicable in real-world applications, this finding is very exciting as it demonstrates that our approach is a better choice than PITF when only one tag (perhaps two as well) recommendation is required. The more evident advantage of our approach in terms of Rankscore for top-1 recommendation deserves further deliberation. Rankscore is concerned with not only the ranking order between correct and incorrect tag recommendations but also the ranking order among correct tag recommendations. If the first tag of an item is successfully recommended, the recommender will receive a higher appraisal than if the second tag is successfully recommended. The more evident advantage of JIT over PITF in terms of Rankscore for top-1 recommendation shows that our approach does better in identifying higher-rank tags (to the best of our knowledge, this is the first paper that investigates the ranking quality of tag recommenders). PITF only cares about the partial order between used tags and unused tags on an item, but does not consider the relative importance among used tags. In other words, all used tags are treated equally by PITF. This explains why PITF performs consistently in predicting different numbers of tags. Our approach, on the other hand, explicitly considers the differential importance of tags at different positions. Under our unified user profiling scheme, used tags are weighted according to their positions. As a result, more attention is given to the first tag in our joint recommendation approach, hence the good prediction quality on the first tag.

Comparing JIT and IB, while they both follow the item-based CF approach, JIT generally performed significantly better than IB. One reason is that in the neighboring item weighting process, we not only consider its similarity to the target item but also its probability to be selected by the active user. The explicit consideration of the user's quantified interest in each neighboring item makes the recommendation of tags more personalized and thus more accurate. Another reason is that user's item-tag profiles are largely smoothed under the joint recommendation framework and the smoothed profile will enlarge the effective item neighborhood size. More specifically, since the smoothed item-tag profile contains predicted tag assignments for many unselected but potential (likely to be selected) items, those potential items similar to the test item can also be added into the neighborhood. These possible reasons will be further examined in following sections. On the pass, note that the impressive result of the item-

based approach on the Delicious dataset shows that simple methods can also yield very good results when data quality is sufficiently good.

Variants for Tag Recommendation

To further examine the effectiveness of the way we chose to generate tag recommendations, we compared it with two variants. One variant (referred to as JIT-DIRECT) recommends high probability tags directly for each test item based on the joint recommendation result. The other variant (referred to as JIT-SIM) performs item-based tag recommendation on top of joint recommendation result without considering the possibility of test items being selected. Figure 13 contrasts JIT and the variants with regard to Rankscore (results on other metrics are similar and hence omitted.)



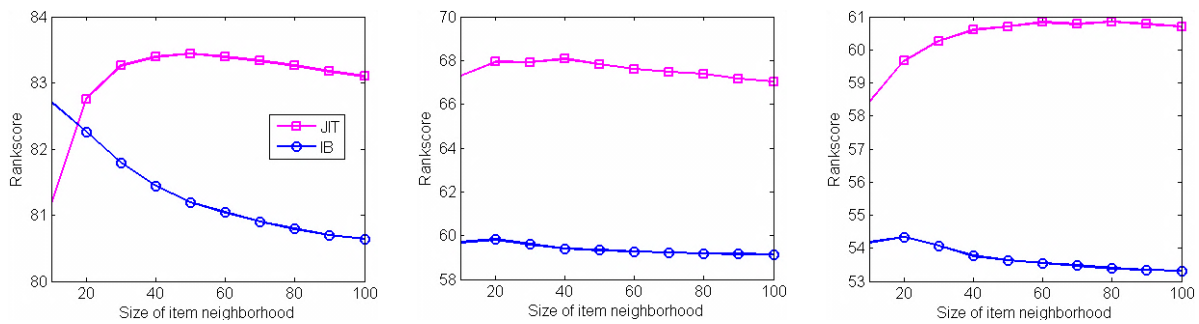
(a) Delicious (b) CiteULike (c) Bibsonomy

Figure 13. Impact of considering users' potential preference on items.

The tag recommendation approach we adopted consistently outperformed both variants across all datasets. The performance advantage of our approach over JIT-SIM lies in the fact that users typically have different extents of interest in different items. Usually, a user's varied preference in her selected items cannot be differentiated explicitly in the tagging data as a result of the uniform tagging process. However, under our joint recommendation framework, a user's interest in each item can be pre-identified before making final tag recommendations. Taking advantage of such pre-identified preference information enables us to produce more personalized and accurate tag recommendation. In addition, the clear superiority of JIT-SIM over JIT-DIRECT shows that (predicted) tag assignments of the same user on other items, especially those potential (highly similar but unselected) items, are of great importance in making successful tag recommendations.

The Smoothing Effect of Joint Recommendation

To better understand the underlying reason of the performance superiority of JIT over IB, we analyzed how their performance varies with item neighborhood size. For both methods, we used the same way to calculate the similarities between items. Specifically, the similarity between two items was measured by the cosine distance of their corresponding TF-IDF weighted tag vectors obtained through the overall item-tag co-occurrence matrix. Figure 14 shows the Rankscore versus item neighborhood size for top-5 tag recommendation (results on other performance metrics and for other numbers of recommendations are similar and hence omitted.)



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

Figure 14. Rankscore versus item neighborhood size.

The results show that the performance of our joint recommendation approach improved quickly with the increase of the item neighborhood size and became stable when item neighborhood size reached around 40~60, whereas the performance of the traditional item-based tag recommendation started to drop after the item neighborhood size surpassed 10 or 20. This striking difference indicates that the smoothed item-tag profile through joint recommendation enables us to identify more similar but unselected items as effective neighbors of the test item (of course, these unselected items are useful only if they are recommended jointly with high quality tags to the user).

Parameter Settings

The key parameters involved in our approach are: d – the dimensionality of the three subspaces, α – the rescaling parameter of tag weights, λ – a parameter reflecting the tradeoff between a user's potential and current interests in items, γ – 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 $p(i,t)^{real}$, and s – the item neighborhood size (for tag

recommendation only). As our preliminary experiment showed that the performance of our approach for both item and tag recommendations was quite stable with the change of both α and d in a large range, we uniformly set α to 20 and d to 50 for both item and tag recommendations. The optimal values of the remaining parameters for item recommendation were: $\lambda = 1$ and $\mu = 1$ for all datasets, and $\gamma = 0.5$, 0.5, and 0.2 for the Delicious, CiteULike, and Bibsonomy datasets, respectively. The optimal settings for tag recommendation were: $\lambda = 0.3$, $\gamma = 0$, $\mu = 0.2$, and $s = 50$ for the Delicious dataset; $\lambda = 0.7$, $\gamma = 0.2$, $\mu = 0.2$, and $s = 20$ for the CiteULike dataset; $\lambda = 0.7$, $\gamma = 0.3$, $\mu = 0.8$, and $s = 60$ for the Bibsonomy dataset.

The optimal settings we found show a striking difference between item recommendation and tag recommendation. For item recommendation, the optimal λ and μ were both 1. $\lambda = 1$ indicates that a user's current interest in items is of no use. Recalling that we should not recommend items that have been previously selected by users in a top-N item recommendation task, this is a natural result. $\mu = 1$ resulted from the same reason. On the other hand, for tag recommendation, the optimal values for λ and μ both lay between 0 and 1. The underlying reason is that a user's past interest in items as well as her preference in annotating these items is of great importance in predicting her future tag assignments. Therefore, we need to tune four parameters (λ, γ, μ, and s) for tag recommendation and only one (γ) for item recommendation.

Note that for the sake of computational efficiency, we set δ in equation (6) to zero and did not consider it during our parameter tuning process. The advantage of this simplified strategy is that the time complexity of our approach drops significantly and the scalability of our approach also extends largely, while the side effect is that the performance might be somewhat sacrificed. However, our evaluation shows that even with this simplification, our joint recommendation approach was still able to yield very competitive recommendation results.

DISCUSSION

While the selection of items and the assignment of tags take place simultaneously in real tagging activities, item recommendation and tag recommendation have been addressed separately in previous research. In response to this asymmetric treatment of item and tag in existing recommendation approaches, we proposed a joint item-tag recommendation framework

to recommend items and tags simultaneously, with tags explaining the reasons to select items. The idea to make joint item-tag recommendations essentially calls for a symmetric way to understand the underlying structure of tagging data. As a result, we systematically examined all the possible interactions among users, items, and tags, and proposed an integrated data model to represent the tagging data. On top of this data model, we developed a unified scheme to profile users in the item and tag space jointly. The unique contribution of this unified user profiling scheme is the integration of binary interactions and ternary interactions through the introduction of hidden items and hidden tags. The significance of this paper is that it provides a new perspective to better understand tagging data, a new mechanism to take advantage of all available information, and more importantly, a completely new way to generate recommendations in social tagging systems. We believe that it will contribute many insights and inspirations to further research in this area.

While we have focused on item recommendation and tag recommendation problems under our joint recommendation framework in this paper, this framework is actually applicable to many other types of recommendations, such as recommending tags to a given user (for topic subscription), recommending items to a given user with several given tags (for context-aware recommendation), and recommending <item, tag> pairs to "lazy" users. 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 framework is suitable for almost any type of recommendation tasks that might arise in the social tagging context. The key advantage of our framework is that it is able to take advantage of tag recommendation and item recommendation results simultaneously. Furthermore, only when the recommendations to users are presented in a joint form, will we be able to make effective use of the *wisdom from the crowd* reflected through the overall item-tag co-occurrence matrix.

From the perspective of item recommendation, the ability of our joint recommendation approach to explain the item recommendations with tags has practical implications in its own right, besides providing justifiability of recommendations. Some of the benefits are:

1) Providing transparency to understand item recommendations without involving any technical details of the recommender systems. In the few representative existing approaches for recommendation explanation (Bilgic et al. 2004; Herlocker et al. 2000), users have to know how

the system works to some extent. However, our joint item-tag recommendation framework poses no such need as tags can explain items directly with their semantic meanings.

2) Facilitating the exploration of the recommendations. Tags can be viewed as labels for the recommendations, allowing users to navigate interested recommendations through tags.

3) Promoting the acceptance of recommendations by users. As Herlocker et al. (Herlocker et al. 2000) reported, a user is more likely to try (or click) a recommended item if she is explicitly informed why this recommendation has been made. On the other hand, a user will often neglect a recommendation if no trustable information about the target item is provided.

4) Promoting more active and targeted participation from users to improve the recommendations. If a user finds some of the recommendations not satisfactory (often as a result of non-standard usage of tags), she can get some clues from the explaining tags and correct her bad tagging behavior accordingly to attain better recommendations from the system.

CONCLUDING REMARKS

Social tagging is a main component of Web 2.0 and has been widely recognized as one of the key technologies underpinning next-generation knowledge management platforms. A large amount of tacit knowledge in users or employees' minds transforms into explicit annotations during the tagging process. Tag-based item recommendation holds the potential to yield substantial quality improvement (over traditional item recommendation) by exploiting the additional information embedded in such annotations. In this paper, we have presented a joint item-tag recommendation framework, which is able to utilize complete information in the tagging data and produce high-quality recommendations. Specifically, we have made the following major contributions.

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

2) Based on the proposed data model, 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 have presented a tensor decomposition approach to compressing the profile tensor constructed from profiles of all

users. In addition, we have discussed a systematic method to represent old users and fold-in new users in the low-dimensional space based on the tensor factorization result.

4) On top of the matrix-based user profiling scheme, we have developed a joint recommendation framework that makes joint item-tag recommendations to users, with tags indicating the users' topical interest in each item. The proposed recommendation framework deals with items and tags in a symmetric manner and applies to both item recommendation and tag recommendation.

While the utility of our proposed approach has been demonstrated in our evaluation using real-world data, our work certainly has its limitations. The particular ways we have adopted to make recommendations and synthesize item and tag recommendations are heuristic in nature. How to maximize the utility of all available information is still an open question. Thus, our work opens up several avenues for future research. One direction is to find 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 for synthesizing the raw joint item-tag recommendations. The third direction is to develop new and better methods to make item/tag recommendations based on the synthesized joint item-tag recommendations. Finally, our framework can be extended to other recommendation tasks, such as context-aware item recommendation for a given user with several given tags and paired <item, tag> recommendation to a user.

REFERENCES

Badrul, S., George, K., Joseph, K., and John, R. "Item-based Collaborative Filtering Recommendation Algorithms," in: *Proceedings of the 10th international conference on World Wide Web*, ACM, Hong Kong, Hong Kong, 2001, pp. 285-295.

Bilgic, M., Mooney, R., and Rich, E. "Explanation for Recommender Systems: Satisfaction vs. Promotion," in: *Department of Computer Sciences*, University of Texas at Austin, Austin, Texas, USA, 2004.

Breese, J.S., Heckerman, D., and Kadie, C. "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," in: *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers Inc., Madison, Wisconsin, USA, 1998, pp. 43-52.

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

Gemmell, J., Schimoler, T., Mobasher, B., and Burke, R. "Hybrid tag recommendation for social annotation systems," in: *Proceedings of the 19th ACM international conference on Information and knowledge management*, ACM, Toronto, ON, Canada, 2010, pp. 829-838.

Givon, S., and Lavrenko, V. "Predicting Social-tags for Cold Start Book Recommendations," in: *Proceedings of the third ACM conference on Recommender systems*, ACM, New York, New York, USA, 2009, pp. 333-336.

Halpin, H., Robu, V., and Shepherd, H. "The Complex Dynamics of Collaborative Tagging," in: *Proceedings of the 16th international conference on World Wide Web*, ACM, Banff, Alberta, Canada, 2007, pp. 211-220.

Hao, M., Irwin, K., and Michael, R.L. "Effective missing data prediction for collaborative filtering," in: *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, Amsterdam, The Netherlands, 2007, pp. 39-46.

Herlocker, J.L., Konstan, J.A., and Riedl, J. "Explaining collaborative filtering recommendations," in: *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, ACM, Philadelphia, Pennsylvania, USA, 2000, pp. 241-250.

Hevner, A.R., March, S.T., Park, J., and Ram, S. "Design science in information systems research," *MIS Quarterly* (28:1) 2004, pp 75-105.

Hofmann, T. "Latent semantic models for collaborative filtering," *ACM Transactions on Information Systems* (22:1) 2004, pp 89-115.

Hotho, A., Jäschke, R., Schmitz, C., and Stumme, G. "Information Retrieval in Folksonomies: Search and Ranking," in: *The Semantic Web: Research and Applications,* Y. Sure and J. Domingue (eds.), Springer Berlin / Heidelberg, 2006, pp. 411-426.

Huang, Z., Zeng, D., and Chen, H. "A Comparison of Collaborative-Filtering Recommendation Algorithms for E-commerce," *Intelligent Systems, IEEE* (22:5) 2007, pp 68-78.

Jäschke, R., Marinho, L., Hotho, A., Schmidt-Thieme, L., and Stumme, G. "Tag Recommendations in Social Bookmarking Systems," *AI Communications* (21:4) 2008, pp 231-247.

Jun, W., Arjen, P.d.V., and Marcel, J.T.R. "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," in: *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, Seattle, Washington, USA, 2006, pp. 501-508.

Jun, W., Arjen, P.d.V., and Marcel, J.T.R. "Unified relevance models for rating prediction in collaborative filtering," *ACM Transactions on Information Systems* (26:3) 2008, pp 1-42.

Kitts, B., Freed, D., and Vrieze, M. "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*, ACM, Boston, Massachusetts, USA, 2000, pp. 437-446.

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

Kolda, T.G., and Sun, J. "Scalable Tensor Decompositions for Multi-aspect Data Mining," in: *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, IEEE Computer Society, Pisa, Italy, 2008, pp. 363-372.

Koren, Y. "The BellKor Solution to the Netflix Grand Prize," http://www.netflixprize.com/community/viewtopic.php?id=1537, 2009.

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

Linden, G., Smith, B., and York, J. "Amazon.com recommendations: item-to-item collaborative filtering," *Internet Computing, IEEE* (7:1) 2003, pp 76-80.

Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., and Stumme, G. "Evaluating similarity measures for emergent semantics of social tagging," in: *Proceedings of the 18th international conference on World Wide Web*, ACM, Madrid, Spain, 2009, pp. 641-650.

Parra, D., and Brusilovsky, P. "Collaborative Filtering for Social Tagging Systems: An Experiment with CiteULike," in: *Proceedings of the third ACM conference on Recommender systems*, ACM, New York, New York, USA, 2009, pp. 237-240.

Paul, R., Neophytos, I., Mitesh, S., Peter, B., and John, R. "GroupLens: an open architecture for collaborative filtering of netnews," in: *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, ACM, Chapel Hill, North Carolina, USA, 1994.

Peng, J., and Zeng, D. "Exploring Information Hidden in Tags: A Subject-based Item Recommendation Approach," Proceedings of 19th Workshop on Information Technologies and Systems, Phoenix, USA, 2009a, pp. 73-78.

Peng, J., and Zeng, D. "Topic-based Web Page Recommendation Using Tags," Proceedings of the 2009 IEEE international conference on Intelligence and security informatics, IEEE, Dallas, Texas, USA, 2009b, pp. 269-271.

Peng, J., and Zeng, D. "Tag-based Smoothing for Item Recommendation," in: *The 2010 IEEE International Conference on Service Operations and Logistics*, IEEE, Qingdao, China, 2010a, pp. 452-456.

Peng, J., Zeng, D., Liu, B., and Zhao, H. "CFUI: Collaborative Filtering with Unlabeled Items," Proceedings of 20th Workshop on Information Technologies and Systems, St. Louis, Missouri, USA, 2010b.

Pennock, D.M., Horvitz, E., Lawrence, S., and Giles, C.L. "Collaborative Filtering by Personality Diagnosis: A Hybrid Memory and Model-Based Approach," in: *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers Inc., Stanford, California, USA, 2000, pp. 473-480.

Rendle, S., Marinho, L.B., Nanopoulos, A., and Schmidt-Thieme, L. "Learning Optimal Ranking with Tensor Factorization for Tag Recommendation," in: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, Paris, France, 2009, pp. 727-736.

Rendle, S., and Schmidt-Thieme, L. "Pairwise interaction tensor factorization for personalized tag recommendation," in: *Proceedings of the third ACM international conference on Web search and data mining*, ACM, New York, New York, USA, 2010, pp. 81-90.

Sen, S., Vig, J., and Riedl, J. "Tagommenders: Connecting Users to Items through Tags," in: *Proceedings of the 18th international conference on World Wide Web*, ACM, Madrid, Spain, 2009, pp. 671-680.

Si, L., and Jin, R. "Flexible mixture model for collaborative filtering," Proceedings of the 20th International Conference on Machine Learning, Washington DC, USA, 2003, pp. 704-711.

Song, Y., Zhang, L., and Giles, C.L. "A sparse gaussian processes classification framework for fast tag suggestions," in: *Proceeding of the 17th ACM conference on Information and knowledge management*, ACM, Napa Valley, California, USA, 2008, pp. 93-102.

Symeonidis, P., Nanopoulos, A., and Manolopoulos, Y. "A Unified Framework for Providing Recommendations in Social Tagging Systems Based on Ternary Semantic Analysis," *IEEE Transactions on Knowledge and Data Engineering* (22:2) 2010, pp 179-192.

Takács, G., Pilászy, I., and Németh, B. "Investigation of various matrix factorization methods for large recommender systems," in: *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*, ACM, Las Vegas, Nevada, 2008, pp. 1-8.

Tan, P.-N., Kumar, V., and Srivastava, J. "Selecting the right interestingness measure for association patterns," in: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, Edmonton, Alberta, Canada, 2002, pp. 32-41.

Tso-Sutter, K.H.L., Marinho, L.B., and Schmidt-Thieme, L. "Tag-aware Recommender Systems by Fusion of Collaborative Filtering Algorithms," in: *Proceedings of the ACM symposium on Applied computing*, ACM, Fortaleza, Ceara, Brazil, 2008, pp. 1995-1999.

Wetzker, R., Umbrath, W., and Said, A. "A hybrid approach to item recommendation in folksonomies," in: *Proceedings of the WSDM '09 Workshop on Exploiting Semantic Annotations in Information Retrieval*, ACM, Barcelona, Spain, 2009, pp. 25-29.

Zeng, D., and Li, H. "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*, Springer-Verlag, Taipei, Taiwan, 2008, pp. 320-330.

Zhang, Z.-K., Zhou, T., and Zhang, Y.-C. "Personalized Recommendation via Integrated Diffusion on User-Item-Tag Tripartite Graphs," *Physica A: Statistical Mechanics and its Applications* (389:1) 2010, pp 179-186.

Zhao, S., Du, N., Nauerz, A., Zhang, X., Yuan, Q., and Fu, R. "Improved Recommendation based on Collaborative Tagging Behaviors," in: *Proceedings of the 13th international conference on Intelligent user interfaces*, ACM, Gran Canaria, Spain, 2008, pp. 413-416.

Zhen, Y., Li, W., and Yeung, D. "TagiCoFi: Tag Informed Collaborative Filtering," in: *Proceedings of the third ACM conference on Recommender systems*, ACM, New York, New York, USA, 2009, pp. 69-76.