Personalization of Tagging Systems

Jun Wang, *University College London, UK*
Maarten Clements, *Delft University of Technology, the Netherlands*
Jie Yang, *Loomia, USA*
Arjen P. de Vries, *Centrum Wiskunde & Informatica, the Netherlands*
Marcel J.T. Reinders, *Delft University of Technology, the Netherlands*

Abstract

Social media systems have encouraged end user participation in the Internet, for the purpose of storing and distributing Internet content, sharing opinions and maintaining relationships. Collaborative tagging allows users to annotate the resulting user-generated content, and enables effective retrieval of otherwise uncategorised data. However, compared to professional web content production, collaborative tagging systems face the challenge that end-users assign tags in an uncontrolled manner, resulting in unsystematic and inconsistent metadata.

This paper introduces a framework for the personalization of social media systems. We pinpoint three tasks that would benefit from personalization: *collaborative tagging*, *collaborative browsing* and *collaborative search*. We propose a ranking model for each task that integrates the individual user's tagging history in the recommendation of tags and content, to align its suggestions to the individual user preferences. We demonstrate on two real data sets that for all three tasks, the personalized ranking should take into account both the user's own preference and the opinion of others.

Key words: User-Generated Content, Social Media, Collaborative Tagging, Collaborative Filtering, Personalization

1. Introduction

User-generated content has enjoyed an enormous growth. Many web content publishers have shifted from creating their on-line content themselves to providing *collaborative systems*, tools as a playground for 'ordinary' users to publish self-produced content: bookmarks (del.icio.us), photographs (flickr.com), publication references (CiteULike.org) and video clips (YouTube.com). People seem to like these collaborative systems because they enjoy the openness of *social media*. They like the stage provided to exhibit their own creations (or even some representation of themselves), and they appreciate how collaborative systems allow like-minded people to discover those easily.

The flow chart in Figure 1 shows an abstract view of collaborative systems. We distinghuish two usage phases: *indexing*, where users add content they consider interesting or relevant, and, *retrieval*, where users search and explore relevant content. Any user who discovers content can extend the current indexing data (e.g., 'tags' assigned by the creator upon inject) with their own descriptive information or opinion; all users *collaboratively* create the index used in the retrieval phase.

Content can be indexed in many ways. In the traditional library or archive, indexing has been the task of professionals focused on consistency, usually organizing the content through a hierarchical system. With the introduction of tags and ratings in today's online databases however, content indexing has shifted from restricted hierarchies to a more subjective categorization. Tagging allows arbitrary users to assign the keywords (called tags) that they consider representative for the topic of the items. Opinions about the quality of content can often be expressed through ratings. Thanks to their popularity, these *(collaborative) tagging systems* have become valuable tools for sharing and exploring content, where tag-item associations can be aggregated over thousands or even millions of users. Multiple assignment of the same tag by different users provides a basis for index quality, countering the fact that end-users assign their tags in an uncontrolled manner. Still, indexing content through tagging is prone to unsystematic and inconsistent indexing that could harm retrieval performance.

Personalization of tagging systems could support users in both phases, to improve consistency of tag usage among the community, and to improve effectiveness in the retrieval phase. More specifically, we can personalize a collaborative tagging system by combining the target user's preferences with the general opinion expressed by all users collaboratively. This paper first identifies twelve basic tasks that qualify for personalization in tagging systems (Section 2). We formally study and model three of these tasks: *collaborative tagging*, *collaborative browsing*, and *collaborative search*. Using a probabilistic framework, we show in Section 3 how the underlying personalized ranking scores for a given candidate (an item or a tag, depending on

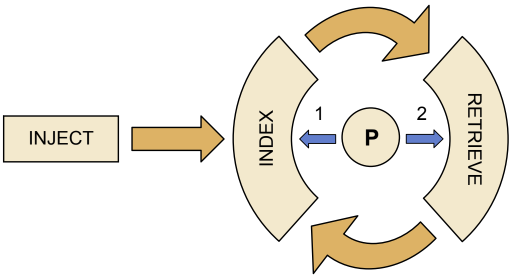


Figure 1: Collaborative Social Media. Content that is injected by any user can be retrieved and indexed by everyone. A personalization engine (**P**) can assist users in both the indexing (P1) and retrieval of content (P2).

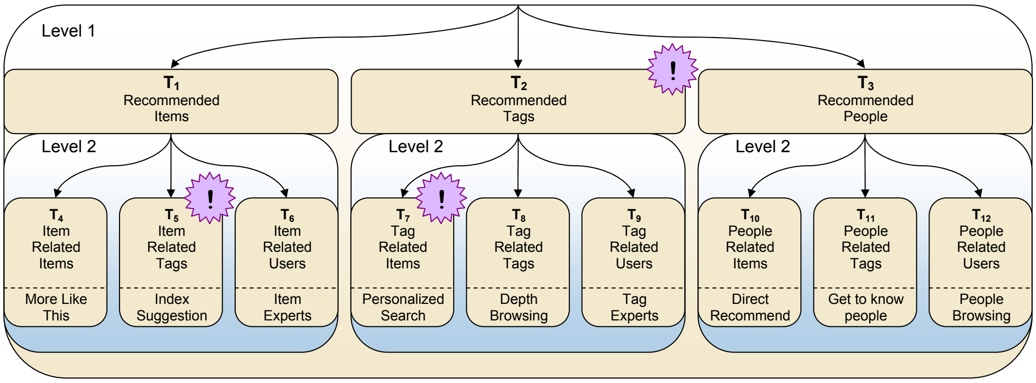


Figure 2: A schematic tree view of the tasks in a social browsing environment. The tasks in level 1 only depend on the target user, while level 2 tasks depend on both the target user and the selected element from level 1. The tasks with a star indicate the focus of this article.

the task) combine its popularity with its likelihood towards the user preference. For probability estimation, we consider different types of generative processes in the tagging data. We choose an optimal candidate model and its smoothing for each of the three tasks, and estimate the probability of the user preferences being generated from that candidate model. Section 4 presents empirical results on two real data sets. The experiments demonstrate effectiveness of the methods, and show that the three personalized models perform significantly better than the non-personalized ones. The collaborative browsing model is shown to outperform ranking-based collaborative filtering approaches provided the availability of sufficient user preference data.

2. User Tasks for Personalization

The three entities of interest in this paper are content, tags, and people. Figure 2 shows the user tasks suited for personalization in a collaborative tagging system. *Level 1* shows the three tasks that apply to users entering the system (T_1-T_3): selecting an item, a tag or a person. *Level 2* indicates the view on the network after the user has selected either an item, tag or user. The 12 resulting tasks that apply for personalization in a collaborative system include common tasks like the recommendation of tags when interesting content has been found (T_5), retrieving relevant content by using tags as queries (T_7), finding experts on a certain topic (T_6,T_9), and, making friends and discovering relevant content through them (T_{10}-T_{12}).

Actual collaborative tagging systems have effectively implemented support for most of these tasks. Table 1 lists some popular systems that allow users to tag content, and evaluates the systems on the twelve tasks defined in Figure 2. The first two columns distinguish between systems that focus on publishing their users' own creations (Video, Photographs, Recipes, Art), and systems that allow users to maintain references to artifacts not necessarily created by themselves (Books, Web pages, Scientific papers). Systems of the first group do usually not support *collaborative* tagging; only the injector of content can assign tags (we call this *individual* tagging to differentiate the two types of systems). The difference can be motivated by the assumption that injectors of self-created content can be expected to know best how to index it.

This paper focuses on a feature missing from most systems listed in Table 1: personalization, i.e., adapting the tagging system to the user's preferences. We concentrate on three common user tasks:
In the indexing (or tagging) phase:
1. Collaborative Tagging: personalizing the tagging process, when a user assigns tags to index content (Figure 2, T_5). Tags act as an indication of subject matter. But, most users are not experienced to describe content by tags precisely, and are insufficiently aware of tags in use by others. For instance, users might tag the same content using 'computer game', 'computergame' or 'computer_games'. Ideally, the system should suggest tags from a common vocabulary that fits the user's intention or taste but is also consistent with other users' tagging behavior. As a result, users discover suitable tagging keywords more easily and, more important, inconsistent tagging behavior is reduced; it has even been claimed that this support for suggesting tags when

Table 1: Social features and personalization of popular tagging systems.

System	**Content**	UG	CT	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}
www.snacksby.com	Recipes	■	■	■	■			■		■	■				
www.youtube.com	Amateur Video	■		■		■	■	■	■	■			■		
www.flickr.com	Photographs	■		■	■			■	■	■	■		■	■	
www.etsy.com	Art	■		■				■	■	■			■		
www.last.fm	Music	◩	■	▨	■	▨	■	■	■	■	■		▨		■
del.icio.us	Bookmarks		■	■	▨			▨	■	■	■	■	■	■	
myweb.yahoo.com	Bookmarks		■	■	■			■	■	■	■	■	■	■	
www.librarything.com	Book Reference		■	▨	■	■	■	■	■	■	■	■	▨	■	■
www.citeulike.org	Citations		■	■	■			■	■	■			■	■	
www.technorati.com	Weblogs		■	■	■	■		■	■	■	■	■	■		
www.amazon.com	Book Reference		■	▨	▨		■	■	■	■	■	■	■	■	

UG: ■ User Generated Content □ Reference to Content
CT: ■ Collaborative Tagging □ Individual Tagging
T: ■ Supported □ Not Supported ▨ Personalized Support

a user is asked to label a certain item would lead to a more coherent 'folksonomy' [6, 18]. Section 3.2 describes our tag suggestion model, and shows how it reinforces tags that have been used frequently by the target user as well as others.

In the retrieval phase:

2. Collaborative Browsing: Navigation through tags provides an effective way to explore and discover relevant content. A common interface element to support content exploration based on tags is the 'tag cloud', that visualizes the tag popularity of the entire network (*popularity cloud*) or the user's previously used tags (*personal cloud*). A personal tag cloud is very useful for navigating to your own items, but if the cloud is used for exploration of other content (Figure 2, T_2), selecting these tags may often result in previously seen content (because unseen items are tagged differently). Popularity-based exploration on the other hand is also limited, as the individual user need may not correspond to the majority one. Personalizing tag exploration could alleviate the search cost and improve the retrieval performance. The proposed collaborative browsing model ranks the tags for a specific user (see Section 3.3).

3. Collaborative Item Search: All systems in Table 1 support the retrieval of content based on tag queries, by either clicking a link or typing the word in a search box (Figure 2, T_7). The amount of data in a collaborative system can however grow extremely fast once it becomes popular; a frequent tag in a system like YouTube results in a list of hundreds of thousands of items. Most of the existing collaborative tagging systems base the item ranking solely on the association between item and query tag, where a combination of the item's popularity and 'freshness' provides the ranking score. However, due to its ambiguity, a tag alone is not semantically and contextually expressive enough to represent the needs of a particular user. For example, the term 'apple' can refer to a type of fruit, a computer brand or the famous city. Section 3.4 proposes to address this problem by personalizing the search, inspired by previous work in personalized text retrieval (e.g., [4, 22, 24, 26, 27]). The model unifies user preferences and tags in a probabilistic framework, to rank items for the user who issued a tag.

3. Personalization Models

3.1. Tagging Data

To describe tagging data, let u be a discrete random variable over the sample space of users $\Phi_U = \{1, ..., M\}$, let i be a random variable over the sample space of items (content) $\Phi_I = \{1, ..., K\}$, and let t be a random variable over the sample space of tags $\Phi_T = \{1, ..., L\}$ (where M, K, and L are the number of users, items, and tags in the collection). Tagging data can be viewed as a 3D matrix, where each element indicates whether a user tagged an item with a specific tag (the matrix is extended when people enter the network, content is introduced or someone uses a new tag). Because the resulting tagging data is usually very sparse, we sum over the 3 dimensions of the matrix to obtain the following three matrices, each representing a simplified view of the original problem (analogously to [19]):

User-Tag (UT): Element (u, t) equals the number of items that user u tagged with tag t;

Item-Tag (IT): Element (i, t) equals the number of users that tagged item i with tag t;

User-Item (UI): Element (u, i) equals the number of tags that user u assigned to item i.

Because the number of tags assigned to an item is not very telling about the user's preference towards that item, we *binarize* the UI matrix (replace non-zero values by 1), representing in element (u, i) only the fact that user u tagged item i.[1]

We now assume that tagging data can be viewed as the result of a two-stage generative process, where we first select a user u, the user generates a tag t, and, the tag in turn generates an item i. The final step in the process is assumed conditionally independent from the user variable to reduce the number of parameters (given the sparsity in the data). The joint probability distribution of this simple generative model equals $p(u, i, t) = p(i|t)p(t|u)p(u)$. The rationale behind the two stage generative process is the following: a user has a preference for certain types of information. It is characterized by the frequency of the tags that he or she uses. Thus for each user, we have a distribution of his and her preference, represented by tags. In the second stage, referred types of information, tags, are instantiated in items. In other words, each tag is characterized by the items that it annotates. In the generative modelling approach, the users generate tags to model their preferences, and the tags will generate items modelling their instantiation in real-world items.

3.2. Collaborative Tagging Model

Personalized collaborative tagging refers to determining which tags to suggest to the user when tagging a given information item, from the pool of tags employed by other users (Figure 2, T_5). The proposed method then suggests tags based on the probability of candidate tag t being used by user u to label item i, i.e., $p(t|u, i)$. We estimate this probability for each candidate tag and suggest the highest ranking ones to the user.

We obtain the conditional probability $p(t|u, i)$ from the generative model:

$$p(t|u, i) = \frac{p(u, i, t)}{p(u, i)} = \frac{p(i|t)p(t|u)p(u)}{p(u, i)} = \frac{p(i|t)p(t|u)}{p(i|u)} \tag{1}$$

Applying a logarithm and ignoring $p(i|u)$ (which does not influence the tag ranking because it is independent from t) we get

$$p(t|u, i) \propto_t \log p(t|u) + \log p(i|t) \tag{2}$$

where $\propto_t$ denotes same rank order with respect to t.

[1] If however explicit preference data like ratings would be available, the UI matrix could instead use these ratings as graded relevance indicators (representing more accurately the degree of relevance of item i to user u).

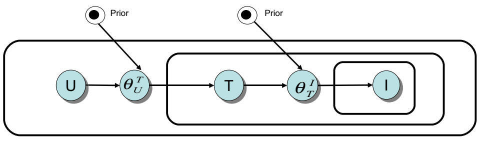


Figure 3: A Generative Model of Tagging Data for Collaborative Tagging.

We instantiate the abstract two-stage generative model into the model shown in Figure 3. A particular user's decision to choose a tag is the result of *choosing* a generative model for that particular user, and then generating the tag using that model. More formally, for each user $u \in \Phi_U$, we choose a *tag*-generative model Θ_u^T:

$$\Theta_u^T = (\theta_u^1, ...\theta_u^t..., \theta_u^L), \text{with } \theta_u^t \in [0,1], \sum_u \theta_u^t = 1, \tag{3}$$

where θ_u^t indicates the probability of generating a tag t from the distribution belonging to the generative model of a user u. At this point, the model does not yet depend on a specific choice of distribution, but later on we will assume a multinomial distribution over the vocabulary of tags. In the second stage, items are the output of a generative process associated with each tag, Θ_t^I. We will assume a multinomial distribution over the vocabulary of items as well.

In Bayesian inference [5], the generative process can be expressed as an integration over all the model parameters to take the uncertainty about the right model into account. In the case of $p(t|u)$, we have:

$$p(t|u) = \int_{\Theta_u^T} p(t|\Theta_u^T, u) p(\Theta_u^T|u) d\Theta_u^T \tag{4}$$

where $p(\Theta_u^T|u)$ is the posterior probability of model parameter Θ_u^T when we have observed some tags (denoted as $\{n(u,t)\}_{t=1}^L$) associated with this user u, and $p(t|\Theta_u^T, u)$ describes the generative process from the estimated model to a tag.

In practice, it is common to approximate the full Bayesian integration over the model by estimating the 'optimal' model parameters $\hat{\Theta}_u^T$ (e.g. by Maximizing their *A Posteriori* probability (MAP)) and then setting $p(\Theta_u^T|u) \approx \delta(\Theta_u^T, \hat{\Theta}_u^T)$ [14]:

$$p(t|u) \approx \int_{\Theta_u^T} p(t|\Theta_u^T, u) \delta(\Theta_u^T, \hat{\Theta}_u^T) d\Theta_u^T = p(t|\hat{\Theta}_u^T) \tag{5}$$

We take the approximation approach, estimating model $\hat{\Theta}_u^T$ that maximizes the probabilities of tag observations $\{n(u,t)\}_{t=1}^L$, and substitute it into Eq. 2. Doing the same for the item generation process gives

$$p(t|u,i) \propto_t \log p(t|\hat{\Theta}_u^T) + \log p(i|\hat{\Theta}_t^I) \tag{6}$$

The tag's ranking scores combine the weights of the two generative processes The first process calculates how probable the candidate keyword is to be generated from the user model (a completely personal suggestion), while the other one computes from the candidate tag (keyword) model how probable the query item would be generated (a completely popularity-based suggestion). Tags that have been used frequently in the past by the target user *and* by other users for the target item will get the highest ranking scores.

Sparse observation data remains a problem for probability estimation using this model. Research on the language modeling approach for information retrieval has however identified various so-called *smoothing methods* to estimate the term probabilities in document models in spite of the sparseness in the term-document matrix [32]. We consider the three main smoothing methods for application in our scenario, giving the details of the derivation in Appendix A. Table 2 summarizes the resulting probability estimations. The smoothing parameter in a user model balances the personal versus the popularity-based suggestion, depending on the estimation method chosen; see also Eq. 19 in Appendix A. Fig. 4 shows *precision at five* (the proportion of relevant tags in the first five suggestions) of tag suggestion in del.icio.us using Bayes' smoothing. As expected, the optimal value of μ shows that tag suggestion performs best when combining personal and popularity-based tags.

3.3. Collaborative Browsing Model

We now discuss how to personalize tag clouds, to improve support of the collaborative browsing task. Hereto, we need to predict the relevance of 'new' tags, i.e., tags that do not yet exist in the given user preference. The $\hat{\Theta}_u^T$ per user u cannot be used directly because, by definition of the task, we have no observations to estimate the user model for the candidate tags (or items).

To address this problem, we invert the Bayesian inference: infer the user's tags rather than deploy them. We represent the user preferences explicitly, such that they can be linked to the preferences of other users. Formally, $\mathbf{q}_u$ denotes the preferences of user u, either based on items or on tags. In the

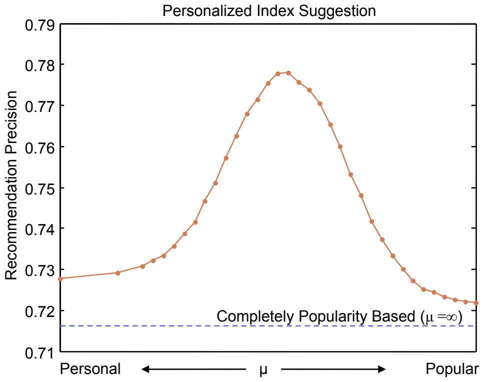


Figure 4: Collaborative tagging should suggest tags based on a mix of personal and popularity-based tags.

former case, item-representation, user preferences are modelled by the set of items that this user has tagged or preferred, i.e., $\mathbf{q}_u = \{i|n(u,i) > 0\}$, where $n(u,i)$ denotes the number of times a user u has tagged an item i. In the tag representation alternative, user preferences are estimated from the set of tags that this user has used, $\mathbf{q}_u = \{t|n(u,t) > 0\}$. Personalizing the 'tag cloud' now corresponds to determining the probabilities of candidate tags t given a user profile $\mathbf{q}_u$, i.e. $p(t|\mathbf{q}_u)$:

$$\begin{aligned} p(t|\mathbf{q}_u) &= \frac{p(\mathbf{q}_u|t)p(t)}{p(\mathbf{q}_u)} \\ &\propto_t \log p(\mathbf{q}_u|t) + \log p(t) - \log p(\mathbf{q}_u) \\ &\propto_t \log p(\mathbf{q}_u|t) + \log p(t) \end{aligned} \tag{7}$$

(where $\log p(\mathbf{q}_u)$ can be ignored for ranking, since it is independent of the target tag t). This ranking formula consists of two parts: the relation between tag and user preference expressed by $p(\mathbf{q}_u|t)$, and global tag popularity $p(t)$. Probability $p(t)$ can be easily estimated from the occurrence frequency in the collection. To estimate the likelihood $p(\mathbf{q}_u|t)$, we choose an optimal tag model $\hat{\Theta}_t^I$ (an item-generation model) for each candidate tag t, and then estimate the probability of the user preference (as query) being generated by the candidate tag model:

$$p(t|\mathbf{q}_u) \propto_t \log p(\mathbf{q}_u|\hat{\Theta}_t^I) + \log p(t) \tag{8}$$

The estimation of $p(\mathbf{q}_u|\hat{\Theta}_t^I)$ depends upon the representation of the user preferences (using items or tags). If we use the representation as a set of

Table 2: Probability estimation. $n(u,t)$ denotes the observation of the number of times that a tag t has been used by the user u while $n(i,t)$ denotes the number of times that a tag t has been used to tag item i. α_t and α_i are the hyper-parameters. ν, μ and λ are the smoothing parameters for the different smoothing methods. (Refer to Appendix A for the derivation.)

(a) probability $p(t|\hat{\Theta}_u^T)$

ML	Laplace	Bayes'	Jelinek-Mercer
$\frac{n(u,t)}{\sum_t n(u,t)}$ $\alpha_t = 1$	$\frac{n(u,t)+\nu}{\sum_t n(u,t)+\nu L}$ $\alpha_t = \nu + 1$	$\frac{n(u,t)+\mu(\sum_u n(u,t)/\sum_{u,t} n(u,t))}{\sum_t n(u,t)+\mu}$ $\alpha_t = \mu\frac{\sum_u n(u,t)}{\sum_{u,t} n(u,t)} + 1$	$\lambda\frac{n(u,t)}{\sum_t n(u,t)} + (1-\lambda)\frac{\sum_u n(u,t)}{\sum_{u,t} n(u,t)}$ -
$\frac{n(i,t)}{\sum_i n(i,t)}$ $\alpha_i = 1$	$\frac{n(i,t)+\nu}{\sum_i n(i,t)+\nu K}$ $\alpha_i = \nu + 1$	$\frac{n(i,t)+\mu\sum_t n(i,t)/\sum_{i,t} n(i,t)}{\sum_i n(i,t)+\mu}$ $\alpha_i = \mu\frac{\sum_t n(i,t)}{\sum_{i,t} n(i,t)} + 1$	$\lambda\frac{n(i,t)}{\sum_i n(i,t)} + (1-\lambda)\frac{\sum_t n(i,t)}{\sum_{i,t} n(i,t)}$ -

(b) probability $p(i|\hat{\Theta}_t^I)$

ML	Laplace	Bayes'	Jelinek-Mercer
$\frac{n(u,t)}{\sum_t n(u,t)}$ $\alpha_t = 1$	$\frac{n(u,t)+\nu}{\sum_t n(u,t)+\nu L}$ $\alpha_t = \nu + 1$	$\frac{n(u,t)+\mu(\sum_u n(u,t)/\sum_{u,t} n(u,t))}{\sum_t n(u,t)+\mu}$ $\alpha_t = \mu\frac{\sum_u n(u,t)}{\sum_{u,t} n(u,t)} + 1$	$\lambda\frac{n(u,t)}{\sum_t n(u,t)} + (1-\lambda)\frac{\sum_u n(u,t)}{\sum_{u,t} n(u,t)}$ -
$\frac{n(i,t)}{\sum_i n(i,t)}$ $\alpha_i = 1$	$\frac{n(i,t)+\nu}{\sum_i n(i,t)+\nu K}$ $\alpha_i = \nu + 1$	$\frac{n(i,t)+\mu\sum_t n(i,t)/\sum_{i,t} n(i,t)}{\sum_i n(i,t)+\mu}$ $\alpha_i = \mu\frac{\sum_t n(i,t)}{\sum_{i,t} n(i,t)} + 1$	$\lambda\frac{n(i,t)}{\sum_i n(i,t)} + (1-\lambda)\frac{\sum_t n(i,t)}{\sum_{i,t} n(i,t)}$ -

items and assume that each item in the user preference is independently generated, we get

$$p(t|\mathbf{q}_u) \propto_t \sum_{i' \in \mathbf{q}_u} \log p(i'|\hat{\Theta}_t^I) + \log p(t) \tag{9}$$

Users with multiple items in their profile get a personalized tag cloud that is selected on the basis of the best tags for all their items. Assuming that the user profile consists of only a single item i, then Eq. 9 resolves to Eq. 6 for the tag suggestion task, because smoothing results in an estimate of $p(t|\hat{\Theta}_u^T)$ that is based solely on background probability $p(t)$.

Taking the alternative representation of user preferences (by their preferred tags), assuming that each tag in the user preference is independently generated, results in the following ranking score:

$$
\begin{aligned}
p(t|\mathbf{q}_u) &\propto_t \log p(\mathbf{q}_u|\hat{\Theta}_t^I) + \log p(t) \\
&= \sum_{t' \in \mathbf{q}_u} n(t', u) \log p(t'|\hat{\Theta}_t^I) + \log p(t) \\
&= \sum_{t' \in \mathbf{q}_u} n(t', u) \log \big(\sum_{i'} p_{\mathrm{ML}}(t'|i') p(i'|\hat{\Theta}_t^I) \big) + \log p(t)
\end{aligned}
\tag{10}
$$

where $p_{\mathrm{ML}}(t|i) = \frac{n(i,t)}{\sum_t n(i,t)}$. The ranking corresponds to the sum (in logarithm domain) of a personalized suggestion and the popularity suggestion. When we know little about the user, we have less observations on the generation from the target to the user preference and thus the prediction comes mainly from the popularity part. The smoothing parameters balance the two suggestions. For instance, in Jelinek-Mercer smoothing, when λ is zero, the first term becomes constant for all the candidate tags and the prediction relies solely on popularity.

The resulting equations are similar to methods for query expansion using relevance feedback in text retrieval [28, 2], where terms are ranked against a set of judged documents from a given user. Still, the underlying problems differ. As we set out to include tags that have not been used previously by this user, we use the information from other users to find suitable tags. We achieve this by looking at how similar the tag is to the items that the target user preferred (item-based user preference), or to the tags that the target user used (tag-based user preference).

3.4. Collaborative Item Search Model

Most tagging systems support the retrieval of items annotated with a given tag, for example by clicking a tag in the browsing interface or typing a word in a search box (Figure 2, T_7). If many different items have been assigned the same popular tags, it becomes however a challenge to find relevant items. We propose to incorporate user preferences to order items on the basis of $p(i|\mathbf{q}_u, t)$, the probability that item i is relevant to tag query t given user profile $\mathbf{q}_u$. It is worth noticing that, by generating multiple tags from an item, the current formulation can be extended to handle a multiple tag query. It is similar to the language models approaches to information retrieval [32].

Using Bayes' and assuming conditional independence between users and tags given an item leads to

$$\begin{aligned} p(i|\mathbf{q}_u, t) = \frac{p(\mathbf{q}_u, t|i)p(i)}{p(\mathbf{q}_u, t)} &= \frac{p(\mathbf{q}_u|i)p(t|i)p(i)}{p(\mathbf{q}_u, t)} \\ &\propto_i \log p(\mathbf{q}_u|i) + \log p(t|i) + \log p(i) \end{aligned} \tag{11}$$

Considering the generative process of tags from items, we derive the item ranking model like before, representing user preferences by their previously used tags (the model using item-based user preference can be obtained similarly):

$$\begin{aligned} p(i|\mathbf{q}_u, t) &\propto_i \log p(\mathbf{q}_u|\hat{\Theta}_i^T) + \log p(t|\hat{\Theta}_i^T) + \log p(i) \\ &\propto_i \Big(\sum_{t' \in \mathbf{q}_u} n(t', u) \log p(t'|\hat{\Theta}_i^T) \Big) + \\ &\qquad \log p(t|\hat{\Theta}_i^T) + \log p(i) \end{aligned} \tag{12}$$

The resulting item ranking is a combination of its popularity ($p(i)$), its probability of generating the query tag ($p(t|\hat{\Theta}_i^T)$), and its probability of generating the user preference ($p(\mathbf{q}_u|\hat{\Theta}_i^T)$).

The model provides a personalized ordering of items in collaborative tagging systems. It combines user preferences for items with the observed user actions involving tags (e.g., selecting a tag to explore items, or tagging a particular item). The role of tags distinguishes this approach from the suggestions provided by existing collaborative filtering approaches, where items are ranked based on user preferences alone, i.e., using $p(i|u)$. Of course, this probability can be derived from our model by marginalizing out the tags, $p(i|\mathbf{q}_u) = \sum_t p(i|t, \mathbf{q}_u)p(t|\mathbf{q}_u)$. In other words, the usage of tags makes the proposed suggestion models more context-aware than traditional collaborative filtering approaches [8, 12, 17].

4. Experiments

4.1. Data Set Preparation

We are not aware of standard data sets suited for the evaluation of our models. We therefore collected data from two well-known collaborative tagging web sites, del.icio.us and CiteULike. The corpus has been crawled between May and October 2006. We collected a number of the most popular

Table 3: Characteristics of the test data sets.

	del.icio.us	CiteULike
Num. of Users	1731	741
Num. of Items	3370	2179
Num. of Tags	1097	960
Num. of User-Item-Tag Triples	772087	20703
Avg. Num. of Tags per User	109	12
Avg. Num. of Items per Tag	135	14
Avg. Num. of Tags per Item	44	6

Table 4: Personalized vs. non-personalized Collaborative Tagging Significant differences marked as ∗.

Precision:

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**3**	**5**	**1**	**3**	**5**	**1**	**3**	**5**
Tagging-BS	**0.852***	**0.749***	**0.656***	**0.867***	**0.768***	**0.673***	**0.866***	**0.780***	**0.683***
Non-Personalized	0.801	0.692	0.615	0.807	0.702	0.626	0.806	0.698	0.631

Recall:

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**3**	**5**	**1**	**3**	**5**	**1**	**3**	**5**
Tagging-BS	**0.137***	**0.358***	**0.517***	**0.138***	**0.361***	**0.522***	**0.135***	**0.360***	**0.520***
Non-Personalized	0.128	0.329	0.482	0.128	0.329	0.484	0.126	0.322	0.480

(a) in the del.icio.us Data Set.

Precision:

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**3**	**5**	**1**	**3**	**5**	**1**	**3**	**5**
Tagging-BS	**0.661***	**0.452***	**0.322***	**0.671***	**0.479***	**0.336***	**0.662***	**0.462***	**0.324***
Non-Personalized	0.515	0.393	0.298	0.522	0.405	0.306	0.503	0.373	0.288

Recall:

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**3**	**5**	**1**	**3**	**5**	**1**	**3**	**5**
Tagging-BS	**0.190***	**0.384***	**0.451***	**0.191***	**0.401***	**0.462***	**0.190***	**0.391***	**0.451***
Non-Personalized	0.147	0.333	0.419	0.147	0.337	0.424	0.142	0.314	0.404

(b) in the CiteULike Data Set.

tags, found which users were using these tags, and downloaded the complete profiles of these users. We applied standard term tokenization techniques from text retrieval followed by stopword removal. Finally, we extracted the user-item-tag triples from each of the user profiles. User IDs are randomly generated to keep the users anonymous. Table 3 summarizes the basic characteristics of the data sets; they can be downloaded from the author's web-site[2].

4.2. Evaluation Protocol

Evaluation Methodology: Since the three user tasks have been transformed into predicting items or predicting tags, we can evaluate the perfor-

[2] http://www.adastral.ucl.ac.uk/~junwang/CollaborativeFiltering.html

mance of our models by holding out a part of the data set as ground-truth data (the test set), and building prediction models from the remaining data (the training set). Prediction accuracy is then measured by ranking items or tags for test users represented by only a part of their profiles, and then compare these ranked items or tags with those in the remaining part of their profile, as known from the held-out ground-truth.

We randomly divide the data set into a training set (80% of the users) and a test set (20% of the users). For cross-validation, all reported results have been averaged over five different random samplings of the data set into training and test set. Experiments with sparsity of user profiles vary the proportion of items and tags that are used in each test user's profile list (e.g., 40%, 60%, 80%). The remaining items and their associated tags are then used to measure prediction performance of the suggestions made by the models.

Before proceeding to the experiments assessing the value of personalization, we first fix the hyper-parameter using five-fold cross-validation on the data set. The value obtained is held constant throughout the rest of the paper. We apply five-fold cross-validation again in all subsequent experiments, to estimate the model parameters from five newly sampled training and test splits.

Evaluation Metrics: We evaluate the effectiveness of the proposed models using evaluation measures at fixed cut-offs, thus normalizing the effectiveness on user-effort (see e.g. [13]). Significance testing is based on the Wilcoxon signed-rank test (again, following the recommendations of [13]).

The collaborative browsing ($p(t|u)$) and search ($p(i|u,t)$) models are designed primarily to assist users with finding relevant tags or items. We measure effectiveness using precision, estimated by the proportion of suggested tags (the collaborative browsing model) or items (the collaborative search model) that are ground truth tags or items. Note that the items and tags in the profiles of the test user represent only a fraction of the items that the user *truly* liked, so we probably underestimate the true precision. On the other hand, we make the assumption that bookmarking an item on a public site indicates the item's relevance, which may overestimate the true precision.

For the collaborative tagging model ($p(t|u,i)$), the motivating user need is to select good keywords to label the given item, and tag *recall* seems an important performance indicator, estimated by the proportion of the ground truth tags that are indeed suggested. We therefore evaluate collaborative tagging using both precision and recall.

Table 5: Precision of personalized vs. non-personalized Collaborative Item Search. Significant differences marked as $*$.

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**
ItemUP-BS	**0.280***	**0.191***	**0.154***	**0.249***	**0.175***	**0.139***	**0.240***	**0.138***	**0.113***
Non-Personalized	0.257	0.171	0.139	0.228	0.144	0.119	0.186	0.112	0.094

(a) in the del.icio.us Data Set.

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**
Tag UP-BS	**0.193***	**0.095***	**0.065***	**0.174***	**0.081***	**0.052***	**0.183***	**0.078***	**0.048***
Non-Personalized	0.141	0.075	0.057	0.118	0.062	0.044	0.111	0.052	0.040

(b) in the CiteULike Data Set.

Table 6: Precision of Collaborative Browsing vs. alternatives,del.icio.us, significant differences over the second best marked as $*$.

User Prof.:	**40%**			**60%**			**80%**		
Top-N:	**1**	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**
Item UP-BS	**0.835***	**0.753***	**0.688***	**0.776***	**0.666***	**0.588***	**0.645***	**0.495***	**0.404***
Non-Person.	0.705	0.690	0.623	0.631	0.591	0.507	0.504	0.413	0.328

(a) Comparison with popularity-based.

User Prof.:	**40%**			**60%**			**80%**		
Top-N:	**1**	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**
Item UP-BS	**0.836***	**0.754***	**0.688**	**0.776***	**0.667**	**0.588***	**0.645***	**0.495***	**0.404***
ItemProb	0.803	0.729	0.672	0.738	0.636	0.561	0.580	0.455	0.379
ItemCos	0.815	0.740	0.683	0.748	0.656	0.576	0.597	0.470	0.385
UserCos	0.793	0.732	0.674	0.733	0.647	0.571	0.583	0.471	0.386

(b) Comparison with ranking-based collaborative filtering.

4.3. Performance of Personalization Models

The first experiments assess the performance of collaborative tagging and collaborative item search. The purpose of the models is to integrate 'collaborative' user behavior into the ranking scores. We therefore compare personalized results to those obtained with a non-personalized ranking (i.e., applying the standard language modelling approach for text retrieval[11]: a generative model from candidate item to the query tag or vice versa). Tables 4 and 5 summarize our results for the two tasks. The models use Bayes' smoothing (see Table 2).

The experimental results support the hypothesis that personalized collaborative models outperform significantly the non-personalized approach, irrespective of the sparsity of user preferences, and irrespective of the data set used. Comparing results on different user profile lengths, we see that the performance improvement of personalized models over non-personalized ones is higher when we have more observations about user preferences.

Table 7: Precision of tag-based vs. item-based user profile representation.

User Prof.:	**40%**			**60%**			**80%**		
Top-N:	**1**	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**
Item UP-BS	**0.835**	**0.753**	**0.688**	**0.776**	**0.666**	**0.588**	**0.645**	**0.495**	**0.404**
Item UP-JMS	0.763	0.723	0.664	0.709	0.632	0.564	0.595	0.473	0.392
Tag UP-BS	0.797	0.719	0.646	0.715	0.623	0.542	0.578	0.447	0.358
Tag UP-JMS	0.716	0.695	0.625	0.656	0.595	0.515	0.530	0.422	0.345

(a) Collaborative Browsing in del.icio.us.

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**
ItemUP-BS	**0.280**	**0.191**	**0.154**	**0.249**	**0.175**	**0.139**	**0.240**	**0.138**	**0.113**
TagUP-BS	0.274	0.187	0.149	0.248	0.164	0.132	0.213	0.132	0.107

(b) Collaborative Item Search in del.icio.us.

User Prof. Length:	**40%**			**60%**			**80%**		
Top-N Returned:	**1**	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**
Item UP-BS	0.173	0.080	0.054	0.134	0.065	0.043	0.140	0.062	0.040
Tag UP-BS	**0.193**	**0.095**	**0.065**	**0.174**	**0.081**	**0.052**	**0.183**	**0.078**	**0.048**

(c) Collaborative Item Search in CiteULike.

Finally, we evaluate the collaborative browsing task. Table 6(a) shows that our model with Bayes' smoothing (denoted as *ItemUP-BS*) outperforms the popularity-based ranking significantly, in all configurations.

When we treat tags as items, item-ranking based collaborative filtering could provide a competing approach. We compare our tag ratings to those of the (state-of-the-art) item-based top-N recommendation [3], using item-based TF×IDF-like (denoted as ItemProb) and user-based cosine similarity recommendation [8] (denoted as UserCos), as implemented in the *Top-N-suggest* recommendation engine[3] [15] (parameters set as specified in the user manual). We report cosine similarity results for item-based approaches [20] as well (denoted as ItemCos). Results in Table 6(b) show that our model usually outperforms the ranking-based collaborative filtering approaches. Significance of the improvement depends on the amount of user profile data for parameter estimation. The explanation for the advantage of our model is that it captures the two generative processes in tagging data, while the existing collaborative filtering techniques take only one generative process into account (in this case, the user-to-tag process).

4.4. Representation of User Profiles

This Section evaluates the effect of the choice of user profile representation (using items or tags, i.e., comparing *ItemUP* and *TagUP*). Table 7(a) shows that, in general, the item-based user preference representation outperforms

[3] `http://www-users.cs.umn.edu/~karypis/suggest/`

the tag-based representation on the collaborative browsing task. We explain this as follows. Users of a tagging system like del.icio.us seem to consider new tags only when they find their previously used tags insufficiently expressive to describe newly added items; otherwise, they will stick to their old tags, and, they are not so likely to update tags assigned previously (to other items). Therefore, the correlation between two related tags within a user profile may be less strong than the relation between two related items. Consequently, using 'old' tags to predict and rank new tags is not as reliable as using the 'old' items for this purpose.

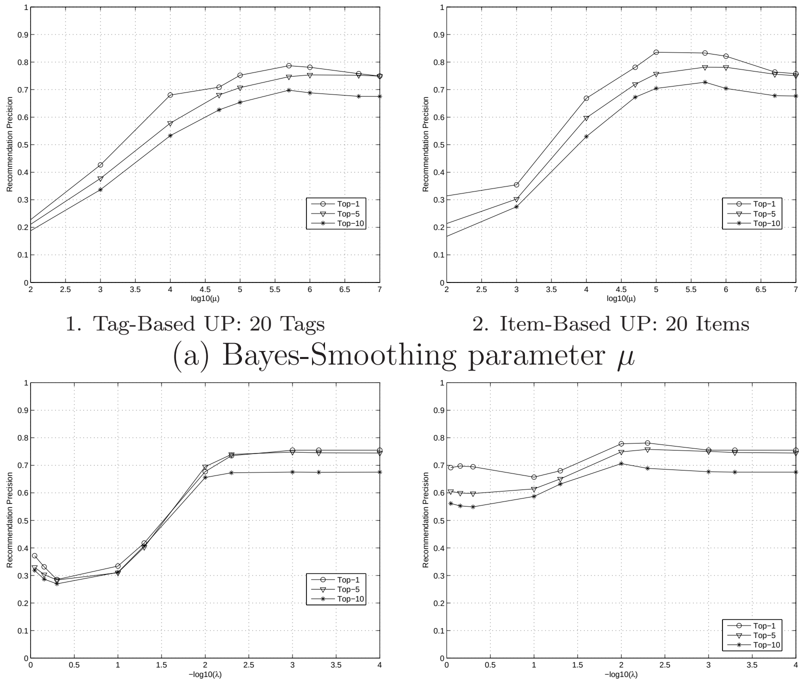
For the collaborative item search model however, the two user profile representations perform differently on different data sets. Notice from Table 7(b) and (c) that the tag-based user preference representation (TagUP-BS) usually outperforms the item-based one in the CiteULike data set, but behaves differently in del.icio.us. We explain the difference by the assumption that tags assigned to scientific papers could be more specific than those assigned to arbitrary URLs. If so, tags in CiteULike may indeed be expected to represent users' interests more accurately than tags in del.icio.us would.

We also observe in Table 7(a) that Bayes' smoothing leads to better results than Jelinek-Mercer smoothing, regardless of the representation of user profiles. We explain this by pointing out that candidate tags are associated to varying numbers of items (such that profile length varies, similar to document length in text retrieval). Bayes' smoothing adapts to the 'tag length' (see explanation in Eq. 20) and therefore performs better.

4.5. Impact of Parameters

This section evaluates sensitivity and impact of the smoothing (hyper-)parameters in the del.icio.us data set. The first experiments address the retrieval phase. Fig. 5(a) and (b) plot precision against hyper-parameter μ in Bayes' smoothing (BS) and λ in Jelinek-Mercer smoothing (JMS), respectively (on a logarithmic scale[4]). We observe that the optimal precision is achieved for higher values of μ and lower values of λ, indicating that parameter estimation benefits from strong smoothing with the background model (which is popularity-based). We attribute this large amount of smoothing to data sparsity. The optimal results of Bayes' smoothing are relatively stable in a wide range between 10^5 to 10^6, and those of Jelinek-Mercer smoothing

[4] We plot $-\log 10(\lambda)$ so smoothing increases along the axis.



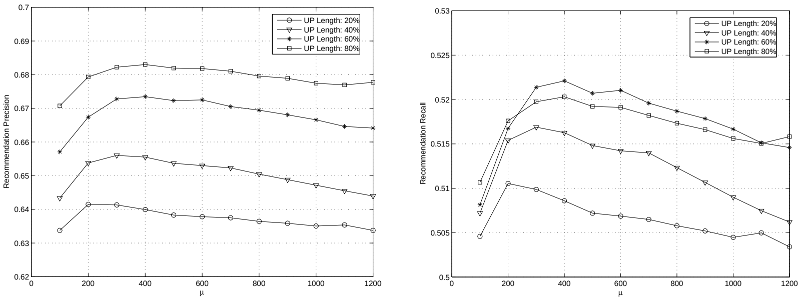
1. Tag-Based UP: 20 Tags
2. Item-Based UP: 20 Items

(a) Bayes-Smoothing parameter μ

1. Tag-based UP: 20 Tags
2. Item-based UP: 20 Items

(b) Jelinek-Mercer Smoothing parameter λ

Figure 5: Impact of hyper-parameters in the Collaborative Browsing Model. In the del.icio.us Data Set.



(a) Precision at Top-5

(b) Recall at Top-5

Figure 6: Impact of hyper-parameter μ in the Collaborative Tagging Model. In the del.icio.us Data Set.

are in the range of 10^{-4} and 10^{-2}, independent from representation and user profile length. The precision obtained with Jelinek-Mercer smoothing is more sensitive to lambda using tag-based user preferences than for item-based user preferences. Tag-based user preferences also require more smoothing (almost corresponding to coordination level matching [32]). Additional experiments (not reported here) show that smoothing in collaborative item search exhibits similar behavior.

Regarding collaborative tagging, recall how Eq. 6 is based on two generative models, the user's tag-generation model and tag's item-generation model. Since the goal is to create a vocabulary shared by all users, we do not want to suggest tags that no other user assigned to the item, so we choose the maximum-likelihood estimator for the tag's item-generation model. Bayes' smoothing is applied in the user's tag-generation model. Fig. 6 plots precision and recall against parameter μ, showing that the optimal μ has a relatively small value when compared to the one in the collaborative browsing model. This means that the indexing model needs less smoothing from the background collection model, which can be explained by the assumption that users tend to prefer previously used tags when tagging new items.

5. Discussion and Related Work

Collaborative tagging systems have recently emerged as tools to structure online databases and user-generated content. To improve the understanding of these social categorization systems, Golder and Huberman conducted an investigation of *del.icio.us*, a web bookmarking system [6]. Many of their findings on system dynamics and semantic problems have been confirmed by measurements on the online photo album *Flickr* by Marlow et al. [18]. These works have also investigated the incentives for users to collaborate in a social tagging system, and although users mostly tag their items for personal use, these tags can still be a great contribution to social exploratory search. Halpin et al. studied the dynamics of collaborative tagging, showing that tagging distributions tend to stabilize into power law distributions [7]. We believe providing a tag suggestion could accelerate this stabilization process. Recently, some early steps have been taken in [10, 23, 25]. Heymann et al. [10] studied del.icio.us, proposing an entropy-based metric for the tag suggestion task; Sigurbjörnsson and Van Zwol [23] proposed tag suggestion for photos in flickr; Song et al. [25] looked into efficiency issues, proposing a fast tag recommendation method. Compared to our paper, these studies have implicitly removed the user from the relationship between tags and items. Our experimental results demonstrate however that the match between tags and information items is incomplete, and the user's personal interests should not be ignored. We have demonstrated significant performance gains by combining other users' tagging behaviors with the user's own tagging vocabulary. In operational tagging systems, we would expect additional benefits from personalized collaborative tagging (T_5), as it may also result in improved categorization consistency.

Previous studies have already shown that tags can not only improve the search effectiveness [9, 29], but also support knowledge discovery [16]. Schenkel et al. [21] rank top-k results looking at social and semantic dimensions. Collaborative filtering predicts a user's interests by looking at other but similar users (user-based collaborative filtering, e.g., [8, 33]) or other but similar items to the target item (e.g., item-based collaborative filtering [3]). Collaborative filtering has so far ignored the tag structure, relying on user-item interactions only. Predictions made about user preferences are conditional on the full user profile, and therefore independent of the user's task. Without other input, collaborative filtering cannot accurately model the important aspects of users or items. Although several hidden aspect models have been proposed to compute recommendations (e.g., [12]), the interpretation of the hidden aspects in terms of their meaning remains usually unclear. User-input in the form of tags could however provide an effective channel to infer and learn the aspects of user interests and contents, resulting in more specific and task-focused recommendations. Our model for collaborative item search (T_7) generates a content ranking that combines the user's preference and the user's task in the form of a tag query. We have shown that this combination retrieves content that is more relevant to the user, compared to a ranking solely on the tag query.

An advantage of tagging systems over recommenders is that users of collaborative filtering systems do usually not benefit directly from rating content, and may view it therefore more as an altruistic activity. Tagging serves however directly the future benefit of effective retrieval of items from the user's personal library: the return on investment is more clear. Likewise, if a system allows the injection of user-generated content, tags are actively used to make the content more easy to retrieve as users like to distribute their creations.

Collaborative item search (T_7) is even more closely related to work on personalized search [1, 22, 24, 26, 27, 30]. Like personalized search, our model utilizes the users' preferences, but it serves a different purpose. Most personalized approaches focus on resolving ambiguity of textual queries. A recent study revealed however that personalized search, due to a lack of a mechanism to model different types of information goals, has little effect on queries with small click entropy and even harms retrieval performance under some situations [4].

Our ranking models integrate the collaborative nature of recommendation systems and the smoothing methods from information retrieval. Recommendation systems have often been limited because of the sparseness of the data in many social networks. Also, new users suffer from cold start problems, because they have not built up their preference profile yet. The smoothing models from information retrieval can relieve collaborative filtering from these problems. The field of information retrieval itself can also benefit from our model, because collaboration has up to this date not been actively used by retrieval systems. The current trend in information systems shows that user statistics are more frequently stored, allowing retrieval systems to integrate this information in the relevance ranking. We have shown that fusion of these two fields leads to better recommendations and retrieval, adapted to individual information needs.

References

[1] P. A. Chirita, C. S. Firan, and W. Nejdl. Personalized query expansion for the web. In *SIGIR*, 2007.

[2] H. Cui, J.-R. Wen, J.-Y. Nie, and W.-Y. Ma. Query expansion by mining user logs. *IEEE TKDE*, 15(4):829–839, 2003.

[3] M. Deshpande and G. Karypis. Item-based top-N recommendation algorithms. *ACM TOIS*, 22(1):143–177, 2004.

[4] Z. Dou, R. Song, and J.-R. Wen. A large-scale evaluation and analysis of personalized search strategies. In *WWW*, 2007.

[5] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman and Hall, 2003.

[6] S. A. Golder and B. A. Huberman. Usage patterns of collaborative tagging systems. *J. Inf. Sci.*, 32(2):198–208, 2006.

[7] H. Halpin, V. Robu, and H. Shepherd. The complex dynamics of collaborative tagging. In *WWW*, 2007.

[8] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR*, 1999.

[9] P. Heymann, G. Koutrika, and H. Garcia-Molina. Can social bookmarking improve web search? In *WSDM*, 2008.

[10] P. Heymann, D. Ramage, and H. Garcia-Molina. Social tag prediction. In *SIGIR*, 2008.

[11] D. Hiemstra. *Using language models for information retrieval.* Doctoral thesis, University of Twente, 2001.

[12] T. Hofmann. Latent semantic models for collaborative filtering. *ACM TOIS*, Vol 22(1):89–115, 2004.

[13] D. Hull. Using statistical testing in the evaluation of retrieval experiments. In *SIGIR*, 1993.

[14] T. Jebara. *Machine Learning: Discriminative and Generative.* Kluwer Academic Publishers, Norwell, MA, USA, 2003.

[15] G. Karypis. Evaluation of item-based top-n recommendation algorithms. In *CIKM*, 2001.

[16] X. Li, L. Guo, and Y. E. Zhao. Tag-based social interest discovery. In *WWW*, 2008.

[17] G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, Jan/Feb.:76–80, 2003.

[18] C. Marlow, M. Naaman, D. Boyd, and M. Davis. Ht06, tagging paper, taxonomy, flickr, academic article, to read. In *HYPERTEXT*, 2006.

[19] P. Mika. Ontologies are us: A unified model of social networks and semantics. In *ISWC*, 2005.

[20] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *WWW*, 2001.

[21] R. Schenkel, T. Crecelius, M. Kacimi, S. Michel, T. Neumann, J. X. Parreira, and G. Weikum. Efficient top-k querying over social-tagging networks. In *SIGIR*, 2008.

[22] X. Shen, B. Tan, and C. Zhai. Context-sensitive information retrieval using implicit feedback. In *SIGIR*, 2005.

[23] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW*, 2008.

[24] B. Smyth and E. Balfe. Anonymous personalization in collaborative web search. *Inf. Retr.*, 9(2):165–190, 2006.

[25] Y. Song, Z. Zhuang, H. Li, Q. Zhao, J. Li, W.-C. Lee, and C. L. Giles. Real-time automatic tag recommendation. In *SIGIR*, 2008.

[26] J.-T. Sun, X. Wang, D. Shen, H.-J. Zeng, and Z. Chen. Mining click-through data for collaborative web search. In *WWW*, 2006.

[27] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *SIGIR*, 2005.

[28] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *SIGIR*, 1996.

[29] S. Xu, S. Bao, B. Fei, Z. Su, and Y. Yu. Exploring folksonomy for personalized search. In *SIGIR*, 2008.

[30] G.-R. Xue, H.-J. Zeng, Z. Chen, Y. Yu, W.-Y. Ma, W. Xi, and W. Fan. Optimizing web search using web click-through data. In *CIKM*, 2004.

[31] H. Zaragoza, D. Hiemstra, and M. Tipping. Bayesian extension to the language model for ad hoc information retrieval. In *SIGIR*, 2003.

[32] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR*, 2001.

[33] Y. Zhang and J. Koren. Efficient Bayesian hierarchical user modeling for recommendation system. In *SIGIR*, 2007.

A. Probability Estimation

We detail parameter estimation for the user's tag-generation model only. Treat a user u's parameters Θ_u^T as random variables (Fig. 3) and estimate their value by maximizing the *a posterior* [14]:

$$\hat{\Theta}_u^T = \max_{\Theta_u^T} p(\Theta_u^T | \{n(u,t)\}_{t=1}^L, \mathbf{a}_u^T) \tag{13}$$

where $n(u,t)$ denotes the number of times that tag t has been used by user u and $\mathbf{a}_u^T$ denotes the parameters of the prior distribution (the *hyper-parameters*). $p(\Theta_u^T | \{n(u,t)\}_{t=1}^L, \mathbf{a}_u^T)$ ($p(\Theta_u^T|u)$ in Eq. 4) is the posterior probability of model parameter Θ_u^T, when we have observed some tags (denoted as $\{n(u,t)\}_{t=1}^L$) associated with this user u. The posterior probability is proportional to the product of the likelihood and the prior probability:

$$p(\Theta_u^T | \{n(u,t)\}_{t=1}^L, \mathbf{a}_u^T) \propto p(\{n(u,t)\}_{t=1}^L | \Theta_u^T) p(\Theta_u^T | \mathbf{a}_u^T) \tag{14}$$

Likelihood $p(\{n(u,t)\}_{t=1}^L | \Theta_u^T) \propto \prod_t (\theta_u^t)^{n(u,t)}$ captures our knowledge about the model parameters from the observed data ($\{n(u,t)\}_{t=1}^L$). Data sparsity causes however often a lack of data for 'accurate' parameter estimation. A solution is to deploy the prior $p(\Theta_u^T | \mathbf{a}_u^T)$ to incorporate prior knowledge of the model parameters. The multinomial's conjugate distribution (the Dirichlet) is chosen as prior to simplify estimation [5]:

$$p(\Theta_u^T | \mathbf{a}_u^T) \propto \prod_t (\theta_u^t)^{a_t - 1} \tag{15}$$

where $\mathbf{a}_u^T = (a_1, \ldots, a_L)$ are the parameters of the Dirichlet distribution. Being the conjugate, the posterior probability after observing data corresponds again to a Dirichlet, with updated parameters:

$$\begin{aligned} p(\Theta_u^T | \{n(u,t)\}_{t=1}^L, \mathbf{a}_u^T) &\propto \prod_t (\theta_u^t)^{n(u,t)} \prod_t (\theta_u^t)^{a_t - 1} \\ &= \prod_t (\theta_u^t)^{n(u,t) + a_t - 1} \end{aligned} \tag{16}$$

Maximizing the posterior probability in Eq. 16 (taking the mode [5]) gives the estimation of the probabilities in the tag-generation model.

$$p(t | \hat{\Theta}_u^T) = \hat{\theta}_u^t = \frac{n(u,t) + a_t - 1}{(\sum_t n(u,t)) + (\sum_t a_t) - L} \tag{17}$$

Varying choices for hyper-parameter a_t lead to different estimators [31]. A constant value $a_t = 1$ gives the *maximum-likelihood* estimator. Setting $a_t = \nu + 1$, where ν is a free parameter, results in the generalized *Laplace smoothing* estimator. Alternatively, the prior can be fit on the distribution of the tags in a given collection:

$$a_t = \mu \cdot p_{\mathrm{ML}}(t) + 1, \text{ where } p_{\mathrm{ML}}(t) = \frac{\sum_u n(u,t)}{\sum_{u,t} n(u,t)} \tag{18}$$

where p_{ML} is the maximum-likelihood estimator. Substituting Eq. 18 into Eq. 17 results in the *Bayes' smoothing* estimator [31]

$$p(t|\hat{\Theta}_u^T) = \hat{\theta}_u^t = \frac{n(u,t) + \mu \cdot p_{\mathrm{ML}}(t)}{(\sum_t n(u,t)) + \mu} \tag{19}$$

Eq. 19 is equivalent to (details in [32])

$$p(t|\hat{\Theta}_u^T) = \hat{\theta}_u^t = \lambda_u p_{\mathrm{ML}}(t|u) + (1 - \lambda_u) p_{\mathrm{ML}}(t), \tag{20}$$

where

$$\lambda_u = \Big(\frac{\sum_t n(u,t)}{\mu + \sum_t n(u,t)}\Big), \; p_{\mathrm{ML}}(t|u) = \frac{n(u,t)}{\sum_t n(u,t)} \tag{21}$$

The result adapts linear interpolation smoothing with $p(t)$, the term probability estimated from a background model. Fixing the background influence as $\lambda_u = \lambda$ results in *Jelinek-Mercer* smoothing [32].