A Probabilistic Approach to Personalized Tag Recommendation

Meiqun Hu, Ee-Peng Lim and Jing Jiang
School of Information Systems
Singapore Management University
80 Stamford Road, Singapore 178902
Email: meiqun.hu@gmail.com, eplim@smu.edu.sg and jingjiang@smu.edu.sg

***Abstract*—In this work, we study the task of personalized tag recommendation in social tagging systems. To include candidate tags beyond the existing vocabularies of the query resource and of the query user, we examine recommendation methods that are based on personomy translation, and propose a probabilistic framework for adopting translations from similar users (neighbors). We propose to use distributional divergence to measure the similarity between users in the context of personomy translation, and examine two variations of such divergence (similarity) measures. We evaluate the proposed framework on a benchmark dataset collected from BibSonomy, and compare with two groups of baseline methods: (i) personomy translation methods based solely on the query user; and (ii) collaborative filtering. The experimental results show that our neighbor based translation methods outperform these baseline methods significantly. Moreover, we show that adopting translations from neighbors indeed helps including more relevant tags than that based solely on the query user.**

I. Introduction

Social tagging systems allow *users* to annotate Web *resources* using *tags*. While not restricted to a controlled vocabulary, tags are freeform keywords that convey meaning and interpretation from the user about the resource being annotated. The vast number of tags contributed by many users collaboratively provide rich semantic structures within the social tagging system. Social tagging offers the users the flexibility for organizing, sharing and exploring resources on the Web. Tags can also serve as metadata to facilitate resource categorization [6], [29] and Web search [22], [31]. Popular social tagging sites include Delicious[1], Flickr[2], Last.fm[3], CiteULike[4], and BibSonomy[5].

Tag recommendation mechanisms are provided at many social tagging sites. Tags are recommended at the time when a user (the *query user*) wants to annotate a resource (the *query resource*). A simple algorithm, which has been used by many social tagging sites, recommends the most frequent (popular) tags that have been assigned to the query resource. While from the system's perspective, these recommendations can help consolidate the tag vocabulary across users, from the users' perspectives, the main utility of tag recommendation is to ease the annotation process for the users. Therefore, it is important to recommend tags according to individual tagging preferences, because tagging is primarily for personal consumption [28].

Users perform tagging to store, organize and relocate Web resources they have discovered. Although synonyms are present in the tag space, *e.g.*, *web* and *internet*, users tend to be consistent in the choice of tags among synonyms for locating the resources later. For instance, if a user prefers to use *web* instead of *internet* in annotating resources, the recommendation algorithm should recommend *web* when *internet* is relevant in the context, so that the resources related to *web* and *internet* are grouped under the same tag for this user. Since information organization and consumption is highly personal, personalized tag recommendations can help the users organize the resources better, which in turn increases the utility of the recommendation service.

In social tagging sites such as Delicious, personalization in tag recommendations is performed by simply matching the popular tags of the query resource with the existing vocabulary of the query user. Such recommendations are not suitable for users who do not follow the general user population in the choice of tags. Let us consider the following three scenarios, where the intended tag of the query user differs from the popular tags assigned to the query resource:

1) When the intended tag has only been used by very few other users for annotating the same resource in the past.
2) When the intended tag has not been used for the query resource, but has been used by the query user for annotating other resources in the past.
3) When the intended tag has not been assigned to the query resource, neither has it been used by the query user herself, but it has been used by other users for annotating other resource(s) in the past.

The recommendation algorithm based solely on tag popularity fails to address all three scenarios. In the literature, collaborative filtering has been applied to tag recommendation [3], [19], which addresses scenario 1. It essentially ranks the existing tags of the query resource by considering only tags that has been assigned by the k-nearest neighbors of the query user. Such methods may be able to pick up infrequent and yet relevant tags for personalized recommendation. However,

[1]http://delicious.com/ for annotating web URLs.
[2]http://www.flickr.com/ for within-host user-contributed images.
[3]http://www.last.fm/ for annotating music profiles.
[4]http://www.citeulike.org/ for scholarly publications.
[5]http://www.bibsonomy.org/ for both scholarly publications and web URLs.

it fails to handle scenarios 2 and 3, because the intended tag has not yet been used for the query resource in these scenarios. To address scenario 2, one can translate from the existing tags of the query resource to the relevant tags in the vocabulary of the query user. For instance, Wetzker *et al.* [27], [28] explored the idea of personomy translation for personalized tag recommendation based on the observed co-occurrence of resource tags and personomy tags. Although having shown effectiveness in recommendation performance, personomy translation base solely on the query user also fails to handle scenario 3, because the intended tag has not yet been used by the query user in this scenario. For addressing scenario 3, we seek to adopt translations from other users who perform similar translations.

In this work, we propose a personomy translation based framework for personalized tag recommendation that can handle all three scenarios in a unified way. Our framework enables adopting translations from similar users. The solution we propose in this work is inspired by the observation of the multilingual composition of the users in a social tagging system. In the case of BibSonomy, for example, a significant amount of tags in German are observed besides the majority of tags in English. We also find that for tags in German, their English equivalents are also observed the in the tag set of the resource. Hence, we expect to see German-speaking population share common translation patterns, *i.e.*, German-English co-occurrences. Therefore, personomy translation performed by similar users can be borrowed to expand the set of candidate tags for recommendation.

Our research contributions in this work can be summarized as follows:

- We solve the task of personalized tag recommendation as a probabilistic ranking problem, and propose a probabilistic framework that is based on personomy translation and adopts translations from similar users.
- We propose to use distributional divergence to measure the similarity between users in the context of personomy translation. In particular, we examine the effectiveness of two such measures, namely JS-divergence and L1-norm.
- We conduct experiments on a benchmark dataset collected from BibSonomy, and compare our proposed framework with two groups of baseline methods: (i) personomy translation based solely on the query user [27], [28]; and (ii) collaborative filtering [3], [19]. The experimental results show that our neighbor based translation methods outperform these baseline methods significantly. Moreover, we show that the translations adopted from neighbors indeed help including more relevant tags than that based solely on the query user.

II. Related Work

Social tagging has brought about an emerging area of research. Trant [25] categorizes the existing works on social tagging into three broad topics: (i) on the *folksonomy* that results from the collective wisdom of users of the social tagging system; (ii) on the *tagging* behavior of users, such as the incentives and motivation for tagging; (iii) on the software aspects of the *social tagging systems*, for improving system performance and enhancing user satisfaction.

The tag recommendation task belongs to the last topic. The task can be further categorized into two types, namely *social tag prediction* and *personalized tag recommendation*. The former, also referred to as *collective tag recommendation* in some works, does not assume a query user for recommendation. It aims at enriching tags for resources that has not been tagged or inadequately tagged. In contrast, the personalized tag recommendation task recommends tags for a target user, *i.e.*, the query user. Our work in this paper belongs to the latter.

In this section, we review studies on personalized tag recommendation, and focus on approaches that are closely related to ours. Due to space limitation, we briefly sample studies on social tag prediction and other studies that consume tagging data.

A. Studies on Social Tag Prediction

Social tag prediction aims at enriching tags for Web resources that are untagged or inadequately tagged. It brings benefit to applications that consume tagging data, such as Web search [12]. The existing approaches include (i) selecting keywords from the content (for text documents) [21], (ii) inferring new tags from the existing tags of the resource [2], [7], [12], [13], and (iii) harvesting tags from other similar or linked resources [1], [18], [24].

B. Studies on Personalized Tag Recommendation

The existing approaches for personalized tag recommendation have looked into many aspects of the folksonomy for bringing relevance to both the query resource and the query user. These approaches include collaborative filtering [19], link analysis ranking [9], [5], machine learning [23], and probabilistic ranking [17], [20], [27].

Collaborative filtering techniques have been applied for personalized tag recommendation by Marinho and Schmidt-Thieme [19]. The recommendation algorithm first selects the *k-nearest neighbors* for the query user, and then recommends tags that are assigned to the query resource by the neighbors. They found that user-tag profile modeling outperforms the user-resource counterpart, suggesting that a user's tag vocabulary is a better indicator of personal preferences.

FolkRank is a random walk technique applied in folksonomies [9]. It follows the intuition and formulation of PageRank. Personalization is done by biasing the preference vector towards the query user and the query resource. Comparable to random walk technique on graphs, Guan *et al.* [5] proposed an algorithm based on heat diffusion on graphs. In their formulation, heat diffuses along the links in the multi-type graph consisting of the query resource, other linked resources and the linked tags. Personalization is done by selecting the query resource and the set of tags used by the query user as the heat sources.

There has been a number of methods following the probabilistic ranking paradigm [17], [20], [27]. Methods closely

related to ours are seen in [20] and [27], [28]. In [20], Marinho *et al.* described a relational learning approach that recommends tags from the neighborhood in a graph of related objects. In their formulation, the graph consists of all posts in the folksonomy, *i.e.*, resource-user pairs. The strength of relations between posts are exploited for estimating the probabilistic weighted average from the neighborhood. However, only simple relations were examined, *i.e.*, user-tag profiles. In [27], Wetzker *et al.* focused on user modeling, in which users are modeled as the set of probabilities for translating the resource tags to personal tags. In a later work [28], they showed improved recommendation accuracy by a similar idea. While [28] introduced a matrix-and-tensor based formulation, we provide a probabilistic view of the method in Section III-C.

C. Other Studies on Social Tagging Systems

As folksonomies become major infrastructures on the Web, applications that consume tagging data can also benefit. Tags can be used for detecting emerging trends and topics [26]. Based on the intuition that *tags reflect the interests of users*, Li *et al.* [16] studied grouping users and URLs by topics of interests mined from tagging data. Kashoob *et al.* applied LDA to model tagging on resources for discovering latent communities of users. In their work, users belong to the same community if they share common tagging vocabulary [10]. Yin *et al.* [29] utilized tagging data for bridging Web objects, and found improved performance in the classification task they studied. Recommending items to users is another promising application in folksonomies [28], [30].

III. A Probabilistic Framework to Personalized Tag Recommendation

In this work, we solve the tag recommendation task as a probabilistic ranking problem. We first introduce the basic concepts in a social tagging system and the notations used in this paper. Next, we give the probabilistic formulation on solving the tag recommendation task, and sketch a probabilistic framework that is based on personomy translation and enables adopting translations from similar users (neighbors). At last, we propose to use distributional divergence to measure the similarity (dissimilarity) between users in the context of personomy translation, and describe two variants.

A. Notations and Problem Definition

A social tagging system $\mathbb{F}$, also referred to as a *folksonomy* [4], consists of three types of entities, namely *resources*, *users* and *tags*, and the set of ternary relationships formed between these entities. Such ternary relationships are assigned by users when they annotate a resource and post the annotations to the social tagging system. Hence, a *post* may contain multiple *assignment* relationships. Formally, let R denote a resource, U denote a user, and T denote a tag. Let $A = \langle R, U, T \rangle$ denote a triplet, and $\mathbb{A}$ denote the set of ternary relationships that exist in a folksonomy. We therefore have

$$\mathbb{F} = \langle \mathbb{R}, \mathbb{U}, \mathbb{T}, \mathbb{A} \rangle, \quad (1)$$

$$\mathbb{A} \in \mathbb{R} \times \mathbb{U} \times \mathbb{T}. \quad (2)$$

For clarity and consistency, we use an uppercase letter to denote a variable and a lowercase letter to denote a particular value (instance) of a variable. We use a blackboard bold letter to denote the set of values for a variable. For instance, $r \in \mathbb{R}$.

One may project a folksonomy onto its subspaces. For example, given a user, denoted by u, the subspace on u consists of the resources annotated by u (denoted by $\mathbf{r}_u$), the set of tags used by u (denoted by $\mathbf{t}_u$), as well as the set of assignment relationships specified by u (denoted by $\mathbf{a}_u$). Formally,

$$\mathbf{r}_u = \{r \in \mathbb{R} : \langle R, U, T \rangle \in \mathbb{A}, R = r, U = u\}, \quad (3)$$

$$\mathbf{t}_u = \{t \in \mathbb{T} : \langle R, U, T \rangle \in \mathbb{A}, U = u, T = t\}, \quad (4)$$

$$\mathbf{a}_u = \{\langle R, U, T \rangle \in \mathbb{A} : U = u\}. \quad (5)$$

The subspace on u is also called the *personomy* of u [8], [27].

The tag recommendation task is to predict the assignment relationships $\langle r, u, t \rangle$. The input given to the recommender is a pair $\langle r, u \rangle_q$ (or equivalently $\langle r_q, u_q \rangle$), *i.e.*, the query resource and the query user. The expected output is the set of recommended tags that are relevant for describing the query resource by the query user, which we denote as $\{t\}_q$. Like an information retrieval task, the set of recommended tags are ranked by scores of relevance, $\delta(r_q, u_q, t)$.

B. A Probabilistic Framework

We treat the tag recommendation task as a probabilistic ranking problem. To compute the relevance score for a candidate tag, we estimate the likelihood of the tag given the pair of query resource and query user. Our main idea is that we can recommend a tag based not only on the query user's behavior but also on other similar users' behaviors. We therefore formulate our probabilistic framework in Equation 7.

$$\delta(r_q, u_q, t) = p(t|r_q, u_q) \quad (6)$$

$$= \frac{\sum_u sim(u, u_q) \times p(t|r_q, u)}{\sum_u sim(u, u_q)} \quad (7)$$

In Equation 7, the overall likelihood of a candidate tag is the weighted average of the likelihoods estimated from multiple users. u are referred to as *neighbors*, and the weight is the similarity between the neighbor and the query user u_q. The proposed framework is general and offers flexibility in three aspects. First, the framework can treat the query user as the most important neighbor. A user is always most similar to herself. Second, many existing methods proposed in the literature can be adopted here to estimate the likelihood $p(t|r_q, u)$. Finally, the measure of similarity between users can also vary, *e.g.*, cosine similarity in user-tag representation can be plugged-in here, without altering the estimation on $p(t|r_q, u)$.

In this work, for estimating the likelihood $p(t|r_q, u)$, we focus on the personomy translation methods proposed by Wetzker *et al.* [27], [28]; for measuring the similarity between users, we propose to use distributional divergence metrics in the context that users are profiled by the translations they perform. We first describe the personomy translation

methods in Section III-C. We then introduce the distributional divergence metrics for measuring the similarity between users.

C. Personomy Translation for Tag Recommendation

Wetzker *et al.* propose to solve the personalized tag recommendation task by estimating the likelihood of translating a resource tag to a personomy tag of the query user. A resource tag (denoted by t_r) is one that has been assigned to the query resource. A personomy tag (denoted by t) is one that has been used by the query user in the past. Presented in [27] and [28], Wetzker *et al.* describe two variations in estimating this likelihood, denoted by $p(t|u,t_r)$. We re-write them in Equations 9 and 10 respectively.

$$p(t|r_q,u) = \sum_{t_r \in \mathbf{t}_r} p(t|u,t_r) \times p(t_r|r_q) \quad (8)$$

$$p(t|u,t_r) = \sum_{r \in \mathbf{r}_u} p(t|r,u) \times p(r|t_r) \quad (9)$$

$$p(t|u,t_r) = \sum_{r \in \mathbf{r}_u} p(t|r,u) \times p(t_r|r) \quad (10)$$

Although [28] introduced a matrix-and-tensor based formulation, we provide a probabilistic view of the method in Equation 10. Both estimations in Equations 9 and 10 rely on tag-tag co-occurrences perceived by the query user, where the former is a personomy tag and the latter is a resource tag. Equation 8 computes the likelihood of a candidate as being translated from all current resources tags.

D. Measuring Similarity between Users

In the context of personomy translation, we argue that *users are similar to each other if they have similar translation patterns*. In other words, we say u_2 is similar to u_1, if when $p(t|u_1,t_r)$ is high, $p(t|u_2,t_r)$ is also high; and when $p(t|u_1,t_r)$ is low, $p(t|u_2,t_r)$ is also low. Based on this intuition, we propose to use *distributional divergence* to measure the similarity between users when they are profiled by their translation probabilities.

Distributional divergence is the measure of distance between distributions. In this work, we describe and examine two distributional divergence metrics, namely *JS-divergence* (Jensen-Shannon divergence) and *L1-norm* [15]. JS-divergence is the symmetrized version of KL-divergence (*Kullback-Leibler divergence*). In information theory, KL-divergence between code samples X and Y (denoted by $D_{KL}(X,Y)$) is a measure the number of extra bits needed to represent the code samples in X using the code samples from Y, as compared to using the code samples from X itself. This interpretation fits our intuition of representing the translation probability from $u1$ using the translation probabilities from $u2$. However, KL-divergence is not a symmetric measure, which makes it not a true metric. Therefore, we use JS-divergence, which is symmetric. Formally,

$$D_{JS}(X,Y) = \frac{1}{2}\left[D_{KL}(X\|M) + D_{KL}(Y\|M)\right] \quad (11)$$

$$D_{KL}(X\|Y) = \sum_i X(i) \log \frac{X(i)}{Y(i)} \quad (12)$$

$$M(i) = \frac{1}{2}(X(i) + Y(i)) \quad (13)$$

In Equation 11, M is the average of the two distributions X and Y.

The L1-norm distance metric is written in Equation 14. It is the sum of absolute distances between elements in the two distributions X and Y.

$$D_{L1}(X,Y) = \sum_i |X(i) - Y(i)| \quad (14)$$

For converting a distance measure into a similarity measure, we adopt the approach by Lee [14].

$$sim_{JS}(X,Y) = 10^{-\beta D_{JS}(X,Y)} \quad (15)$$

$$sim_{L1}(X,Y) = (2 - D_{L1}(X,Y))^{\beta} \quad (16)$$

The β in Equations 15 and 16 are not equivalent. However, they have similar effect on the resulting measurements: higher β gives less importance to the more distant neighbors. Following [14], we do not normalize the similarity scores across different metrics, even though they take different value ranges. For instance, $sim_{JS}(X,Y) \in [0,1]$ and $sim_{L1}(X,Y) \in \left[0, 2^{\beta}\right]$.

In personomy translation, each user is profiled by a set of translation probabilities, one for each t_r. If two users have translation probabilities on a common t_r, we first measure the similarity between $p(T|u_1,t_r)$ and $p(T|u_2,t_r)$ using the metrics defined above. We use $sim^{t_r}(u_1,u_2)$ to denote this intermediate similarity measure. To derive the overall similarity between two users, we take the weighted average of $sim^{t_r}(u_1,u_2)$ on different t_r, and the weight is $p(t_r|u_1)$.

$$sim^{t_r}(u_1,u_2) = sim\left(p(T|u_1,t_r), p(T|u_2,t_r)\right) \quad (17)$$

$$sim(u_1,u_2) = \frac{\sum_{t_r} p(t_r|u_1) \times sim^{t_r}(u_1,u_2)}{\sum_{t_r} p(t_r|u_1)} \quad (18)$$

We interpret $p(t_r|u_1)$ as the likelihood of u_1 having seen t_r during tagging. This likelihood can be estimated from the tags of the resources that u_1 has annotated in the past.

$$p(t_r|u) = \frac{|\{\langle R,U,T\rangle \in \mathbb{A} : R = r \in \mathbf{r}_u, T = t_r\}|}{|\{\langle R,U,T\rangle \in \mathbb{A} : R = r \in \mathbf{r}_u\}|} \quad (19)$$

IV. Experimental Setup

We conduct experiments to demonstrate the effectiveness of the proposed probabilistic framework. We evaluate if the idea of adopting translation from similar users can include more relevant tags beyond the existing tag vocabularies of the query resource and of the query user. We compare our methods with methods based solely on the translations by the query user and methods that are based on collaborative filtering [3], [19].

A. Data Collection

Our datasets are collected from BibSonomy [11]. Snapshots of BibSonomy have also been used as benchmark datasets in the PKDD ECML Discovery Challenge 2009.

We use the 2-core dataset provided in the Discovery Challenge as our training set. It is the snapshot of the BibSonomy as of January 1, 2009. The notion of 2-core indicates that every resource, user and tag appears in at least 2 posts in this training set.

TABLE I
DATASET STATISTICS

	train	validation	test
time frame	start date – 2009-JAN-01	2009-JAN-01 – 2009-JUL-01	2009-JUL-01 – 2010-JAN-01
$\lvert\mathbb{R}\rvert$	22,389	667	258
$\lvert\mathbb{U}\rvert$	1,185	136	57
$\lvert\mathbb{T}\rvert$	13,276	862	525
$\lvert\mathbb{A}\rvert$	253,615	2,604	1,262
$\lvert\mathbb{P}\rvert$	64,120	775	279
avg. posts per user	53.695	5.699	4.895
avg. tags per post	3.955	3.360	4.523
avg. dist. tags per user	61.833	13.191	14.667

We take the task2 dataset used for the Discovery Challenge as our validation set. All posts in this validation set were made between January 1, 2009 and July 1, 2009, and only those for which the resource, the user and all the tags have appeared in the training set are included.

Our test set is taken from the most recent snapshot of BibSonomy, dated on January 1, 2010. We follow the convention adopted in the ECML PKDD Discovery Challenge 2009 for removing non-alphabetic and non-digit characters in the tags and normalizing them to their lowercase NFKC[6] forms. We extract only query posts that satisfy the following three requirements:

- the post was made between July 1, 2009 and January 1, 2010;
- the user has appeared in our validation set;
- the resource and all tags in the post have appeared in our training set.

Therefore, the time order for posts in our datasets is as follows: the test set is later than the validation set, and the validation set is later than the training set. We learn the translation probabilities and the similarities between users from the training set. We tune the parameters for optimal performance using the validation set. At last, we apply the optimal parameter settings when recommending tags for the query posts in the test set. Table I shows the statistics of the three datasets.

B. Evaluation Metrics

We adopt precision-recall curve and f1@5 as the main metrics for performance comparison and optimization. f1@5 is the harmonic mean of precision and recall at the 5-th position in the ranked list of recommended tags for a query post. f1@5 is also the evaluation metric used in the ECML PKDD Discovery Challenge 2009.

To define the evaluation metrics, we use t_i to denote the tag at position i in the ranked list of recommended tags, n_q to denote the total number of truly assigned tags for the query post, and p to denote the position in the list of recommended tags at which the evaluation takes place. Hence,

$$\text{precision@p} = \frac{\sum_{i=1}^{p} I_q(t_i)}{p} \quad (20)$$

$$\text{reall@p} = \frac{\sum_{i=1}^{p} I_q(t_i)}{n_q} \quad (21)$$

$$\text{f1@p} = \frac{2 \times \text{precision@p} \times \text{recall@p}}{\text{precision@p} + \text{recall@p}} \quad (22)$$

where the function $I_q(t_i)$ returns 1 if t_i matches one of the truly assigned tags for the query post and 0 otherwise.

We compute the metrics at $p \in [1, 5]$ for each post in the test set. To gain a user-centric view of tag recommendation performance, we compare the *macro-average* performance of methods. Macro-average is the average of the per-user averages, where the average performance for each user is evaluated first and then summed up and divided by the total number of users in the test set.

C. Methods to be Compared

We evaluate our proposed probabilistic framework by including three groups of methods.

trans-n1 and trans-n2: Both methods follow our proposed probabilistic framework in estimating the likelihood $p(t|r_q, u_q)$. We use letter n to indicate the inclusion of translations from neighbors. The two variations differ in the estimation of $p(t|u, t_r)$. trans-n1 follows Equation 9, and trans-n2 follows Equation 10. We compute the similarities between users based on the estimated $p(t|u, t_r)$ for each user accordingly. When computing the similarity between users, there are two parameters to be determined: (i) β for converting the distributional divergence measure into similarity measure; (ii) k for selecting the number of nearest neighbors. For β, we search in the range $\beta \in \{1, 2, 4, 8\}$ for JS-divergence and $\beta \in \{1, 2, 4, 8, 12, 16\}$ for L1-norm. For k, we search in the range $k \in \{5, 10, 20, 50, 100, 200, 300, 400, 500\}$.

trans-u1 and trans-u2: These methods are special cases of the proposed framework. They remove other users when estimating $p(t|r_q, u)$. In other words, they rely on the translation probabilities estimated for the query user solely, but do not borrow translation from neighbors. We use letter u to indicate such distinction from the trans-n methods. For the estimation of $p(t|u, t_r)$, trans-u1 follows Equation 9, and trans-u2 follows Equation 10.

knn-ur and knn-nt: These methods are direct application of collaborative filtering to tag recommendation in folksonomies [3], [19]. They first select the k-nearest neighbors for the query user and recommend tags that have been assigned by the neighbors to the query resource. The overall relevance score of a candidate tag is the average similarity of the corresponding neighbors. The two variations differ in profiling the users for computing the similarity between users. In knn-ur, each user is represented as a vector of resources, and the vector weights are binary-valued to indicate whether the user has annotated the resource. Whereas in knn-ut, each user is

[6] NFKC stands for Normalization Form Canonical Composition.

represented as a vector of tags. The vector weights are the frequency of tags that have been used by the user[7]. The similarity between users is then computed as the cosine similarity in vector space. There is one parameter to be determined in these methods: k for selecting the number of nearest neighbors. We search k in the same range as that for trans-n methods, *i.e.*, $k \in \{5, 10, 20, 50, 100, 200, 300, 400, 500\}$.

Finally, we also include the baseline method freq-r, as shown in Equation 23. It recommends tags based on the frequency in which the tag has been assigned to the query resource. The underlying assumption is that, *the more often a tag has been assigned to the resource, the more likely it would be used again*.

$$p\left(t|r_q, u_q\right) = \frac{|\{\langle R, U, T\rangle \in \mathbb{A} : R = r_q, T = t\}|}{|\{\langle R, U, T\rangle \in \mathbb{A} : R = r_q\}|} \quad (23)$$

Although not performing personalization itself, freq-r has been reported to work well for tag recommendation tasks [3], especially when combined with methods that do perform personalization [27]. For exploring the performance space, we also combine freq-r with methods listed above. We adopt linear interpolation when calculating the interpolated likelihood of a candidate tag $p\left(t|r_q, u_q\right)$, shown in Equation 24.

$$\begin{aligned} & p_{\text{interpolated}}\left(t|r_q, u_q\right) \\ = \; & \omega \times p_{\text{freq-r}}\left(t|r_q, u_q\right) + (1-\omega) \times p\left(t|r_q, u_q\right) \end{aligned} \quad (24)$$

ω is an additional parameter need to be tuned in the interpolated estimations.

V. Experimental Results

A. Precision-Recall Curve for Top 5 Recommendations

Firstly, we examine the precision-recall curve (pr curve for short) of the six recommendation methods listed in Section IV-C, with and without freq-r. Figure 1 shows the performance on the test set, for which the corresponding parameters are determined by the validation set. *Global setting* refers to applying the same set of parameters to all users, which have been tuned to optimize the macro-average f1@5 on the validation set. *Individual setting* refers to individualized parameters that optimize the average f1@5 for each user on the validation set. L1-norm metric is used for trans-n1 and trans-n2.

Without freq-r, trans-n methods show clearly large advantage over trans-u methods. This holds for both global and individual settings. This consolidates our intuition that borrowing translations from similar users is able to help recommending tags that are relevant to the query user for the query resource. On the whole, trans-n2 performs stronger than trans-n1. trans-n2 performs the best on the test set.

knn-ur always outperforms knn-ut. This observation is consistent with those made in [19], [3]. It suggests that users who are similar in their tag vocabularies are more likely to assign same tags(s) to the same resource, than those who are similar in their collections of annotated resources.

With freq-r, all methods, except knn-ut, give largely improved performance over their non-interpolated counterparts. The performance by knn-ur is brought closer to that by knn-ut. However, the interpolated trans-u and trans-n outperform knn methods by an ample margin. This can be explained by the composition of candidate tags of knn methods. knn methods always recommend tags that have already been assigned to the query resource, in this case, by the k-nearest neighbors. In other words, the candidate tags of knn is a subset of that for freq-r. Hence, freq-r brings little additional benefit to knn-ut when the interpolation parameter ω is optimized. On the contrary, both trans-u and trans-n methods are able to bring non-existing tags to the query resource. These non-existing tags, some of which are indeed adopted by the query user to annotate the query resource, gains performance for the translation based methods over freq-r and knn methods.

Although not performing well by themselves, trans-u1 and trans-u2 methods achieve large improvement when interpolated with freq-r. The candidate set of trans-u methods includes all tags that have been used by the query user in the past, be it relevant or less relevant to the current query resource. Applying trans-u methods alone may recommend highly personal tags that are less relevant to the current query resource. However, when interpolated with freq-r, tags that are relevant to the resource can be brought back. Therefore, we observe significant lift in the performance by trans-u1 and trans-u2 when interpolated with freq-r using optimized parameter settings.

To our surprise, individual setting does not outperform global setting on the test set. Individual settings are obtained by optimizing the average f1@5 for each user on the validation set, however, not all users assign equal number of tags to resources during tagging. It remains a research question on what other optimization criteria are suitable in the context, *e.g.*, precision@1 and area under the pr-curve? This may be part of our future work.

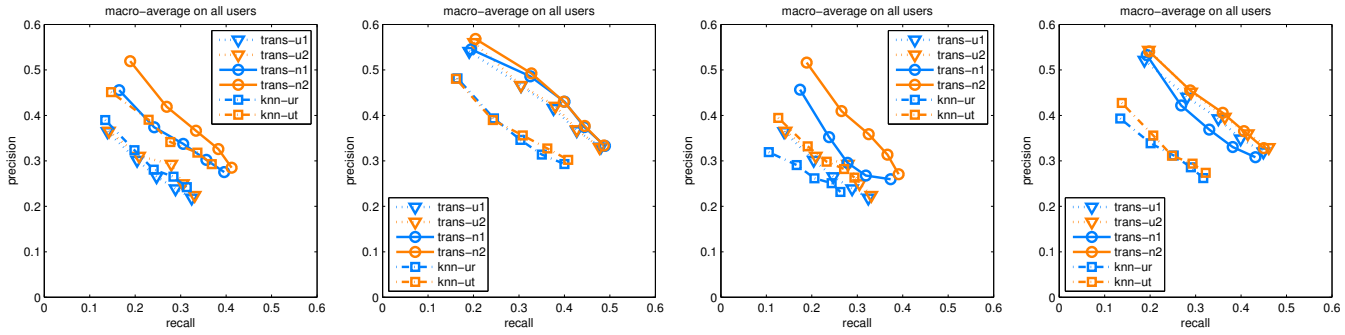
B. F1@5 on the Test Set

Next, we look at the macro-average f1@5 of the methods on the test set, shown in Table II. The best performer within each column are highlighted in boldface. We conduct paired right-tail t-test with significance level of 0.05 to test the best performer against the rest of the methods in each column. We put a * besides the macro-average f1@5 value of the *non-best-performing* method if the t-test indicates that the best performer outperforms the method significantly. Again, L1-norm metric is used in trans-n methods.

Without freq-r, trans-n2 is the best performer in both global and individual settings. It outperforms knn-ur and trans-u methods significantly. trans-n1 gives comparable performance with trans-n2.

With freq-r, the interpolated trans-n1 is the best performer under global setting, and the interpolated trans-u2 outperforms the rest under the individual setting. Under global setting, the interpolated knn methods are outperformed by the interpolation

[7]We have also tried using binary-valued weights in the user-tag representation. However, it shows similar performance with that using frequency-valued weights. Therefore, in this paper, we do not include the binary-valued variation of this method.



(a) Global Setting, without freq-r (b) Global Setting, with freq-r (c) Individual Setting, without freq-r (d) Individual Setting, with freq-r

Fig. 1. Precision-Recall Curve for Tag Recommendation Methods on the Test Set

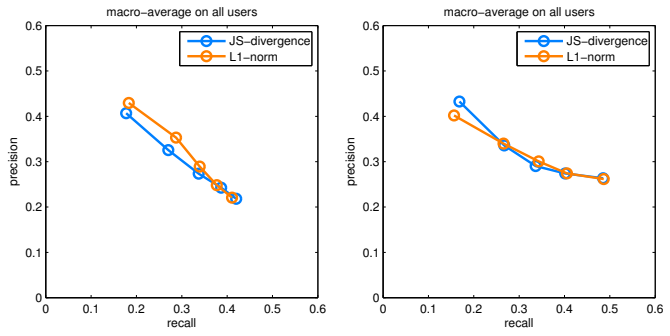
TABLE II
MACRO-AVERAGE F1@5 FOR TAG RECOMMENDATION METHODS ON THE TEST SET

	Global setting		Individual setting	
	without freq-r	with freq-r	without freq-r	with freq-r
trans-u1	*0.238	0.359	*0.238	*0.344
trans-u2	*0.244	0.358	*0.244	**0.354**
trans-n1	0.298	**0.363**	0.281	*0.330
trans-n2	**0.310**	0.362	**0.293**	0.349
knn-ur	*0.248	*0.312	*0.222	*0.260
knn-ut	0.290	*0.321	0.244	*0.263

translation methods significantly. Under individual setting, although the interpolated trans-u2 performs the best, it does not show significant advantage over the interpolated trans-n2.

C. Effect of the Divergence Metrics

Lastly, we observe little difference in the divergence metrics being used, when parameters are optimized. In Section III-D, we have introduced two divergence metrics for measuring the divergence between users in the context of personomy translation, namely JS-divergence and L1-norm. Figure 2 shows the pr-curves by trans-n2 when using these two divergence metrics. Under both global and individual settings, the performance by the two metrics are close, though L1-norm shows slight overall advantage. Similar observation can be made when trans-n1 is used. Therefore, we report the performance by trans-n1 and trans-n2 using L1-norm metric only in Figure 1 and Table II.



(a) Global Setting, without freq-r (b) Individual Setting, without freq-r

Fig. 2. Effect of Divergence Measures on the Validation Set using trans-n2

D. Case Studies

In Table III, we show a few query cases from the test set. We compare the top 5 recommendations given by trans-u and trans-n methods without freq-r.

For user 920, 4 out of the top 5 tags recommended by trans-u1 are indeed personal. However, these recommendations fail to match what the user intends to use for describing the current query resource. In contrast, trans-n1 recommends a few more suitable tags among the top 5 recommendations, but less highly personal tags. Due to the weighted average from neighbors, trans-n1 can retain the balance from recommending highly personal tags. Similar cases happen for user 1119 and user 3217 in the corresponding posts.

VI. CONCLUSION

In this work, we have proposed a probabilistic framework for solving the personalized tag recommendation task. Based on the approach of personomy translation, which translates from the resource tags to personomy tags, we propose to adopt translations from similar users (neighbors) for expanding the set of candidate tags for recommendation. Two divergence measures have been examined for measuring the similarity between users in the context of personomy translation. We found that ample improvement in the recommendation performance can be achieved when adopting translations from neighbors.

Our study in this work focused on the perspective of users. We started with the intuition that, it is due to individual's tagging habits, it makes the personalized tag recommendation difficult for some users. However, from another perspective, the difficulty may also be due to the peculiar characteristics of the resources. In the future, we plan to study the same task from the perspective of resources.

REFERENCES

[1] A. Budura, S. Michel, P. Cudré-Mauroux, and K. Aberer, "To tag or not to tag -: harvesting adjacent metadata in large-scale tagging systems," in *SIGIR '08*, pp. 733–734.

[2] M. Bundschus, S. Yu, V. Tresp, A. Rettinger, M. Dejori, and H.-P. Kriegel, "Hierarchical bayesian models for collaborative tagging systems," in *ICDM '09*, 2009, pp. 728–733.

[3] J. Gemmell, M. Ramezani, T. Schimoler, L. Christiansen, and M. Bamshad, "The impact of ambiguity and redundancy on tag recommendation in folksonomies," in *RecSys '09*, pp. 45–52.

TABLE III
CASE STUDY ON THE RECOMMENDED TAGS BY METHODS

user	item hash	tag assigned	top 5 recommendations	
			trans-u1	trans-n1
920	a45...57f	2008, bookmarking, folksonomy, social, spam, folksonomies, tagorapub, web20, 20, integpub, systems, tagger, web	diplomathesis captcha folksonomybackground closelyrelated **folksonomy**	**folksonomy** tagging **social** **web20** **web**
1119	d16...b50	it, news, technology, blog, feed, technologie	kultur online radio kunst cd	**news** web20 **blog** software **technology**
3217	467...655	annotation, ontology, knowledge, semantic	sql erd eclipse – –	tagging folksonomy **ontology** web20 **semantic**

[4] S. A. Golder and B. A. Huberman, "Usage patterns of collaborative tagging systems," *Journal of Information Science*, vol. 32, no. 2, pp. 198–208, 2006.

[5] Z. Guan, J. Bu, Q. Mei, C. Chen, and C. Wang, "Personalized tag recommendation using graph-based ranking on multi-type interrelated objects," in *SIGIR '09*, pp. 540–547.

[6] P. Heymann and H. Garcia-Molina, "Collaborative creation of communal hierarchical taxonomies in social tagging systems," Stanford InfoLab, Technical Report 2006-10, April 2006. [Online]. Available: http://ilpubs.stanford.edu:8090/775/

[7] P. Heymann, D. Ramage, and H. Garcia-Molina, "Social tag prediction," in *SIGIR '08*, pp. 531–538.

[8] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme, "Bibsonomy: A social bookmark and publication sharing system," in *Proceedings of the Conceptual Structures Tool Interoperability Workshop at the 14th International Conference on Conceptual Structures*, 2006, pp. 87–102. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.115.3646

[9] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme, "Tag recommendations in folksonomies," in *Knowledge Discovery in Databases: PKDD 2007*, 2007, pp. 506–514.

[10] S. Kashoob, J. Caverlee, and Y. Ding, "A categorical model for discovering latent structure in social annotations," in *ICWSM '09*, 2009. [Online]. Available: http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/176/412

[11] Knowledge and Data Engineering Group, "Benchmark folksonomy data from bibsonomy," online, January 2010, university of Kassel.

[12] R. Krestal, P. Fankhauser, and W. Nejdl, "Latent dirichlet allocation for tag recommendation," in *RecSys '09*, pp. 61–68.

[13] R. Krestel and P. Fankhauser, "Tag recommendation using probabilistic topic models," in *ECML PKDD Discovery Challenge '09*, vol. 497, 2009, pp. 131–141.

[14] L. Lee, "Similarity-based approaches to natural language processing," Ph.D Thesis, Harvard University, Cambridge, MA, 1997, Chapter Four.

[15] ——, "Measures of distributional similarity," in *ACL '99*, 1999, pp. 25–32.

[16] X. Li, L. Guo, and Y. E. Zhao, "Tag-based social interest discovery," in *WWW '08*, pp. 675–684.

[17] Z. Liao, M. Xie, H. Cao, and Y. Huang, "A probabilistic ranking approach for tag recommendation," in *ECML PKDD Discovery Challenge '09*, vol. 497, 2009, pp. 143–155.

[18] Y.-T. Lu, S.-I. Yu, T.-C. Chang, and J. Y.-j. Hsu, "A content-based method to enhance tag recommendation," in *IJCAI '09*, 2009, pp. 2064–2069.

[19] L. Marinho, B and L. Schmidt-Thieme, *Collaborative Tag Recommendations*, 2008, ch. Chapter 63, pp. 533–540. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-78246-9_63

[20] L. Marinho, Balby, C. Preisach, and L. Schmidt-Thieme, "Relational classification for personalized tag recommendation," in *ECML PKDD Discovery Challenge '09*, vol. 497, 2009, pp. 7–15.

[21] H. Murfi and K. Obermayer, "A two-level learning hierarchy of concept based keyword extraction for tag recommendation," in *ECML PKDD Discovery Challenge '09*, vol. 497, 2009, pp. 201–214.

[22] M. G. Noll and C. Meinel, "The metadata triumvirate: Social annotations, anchor texts and search queries," *Proceedings of WI/IAT '08*, vol. 1, pp. 640–647, 2008.

[23] S. Rendle and L. Schmidt-Thieme, "Factor models for tag recommendation in BibSonomy," in *ECML PKDD Discovery Challenge '09*, vol. 497, 2009, pp. 235–242.

[24] S. B. Subramanya and H. Liu, "SocialTagger - collaborative tagging for blogs in the long tail," in *SSM '09*.

[25] J. Trant, "Studying social tagging and folksonomy: a review and framework," *Journal of Digital Information*, vol. 10, no. 1, pp. 1–44, 2009.

[26] R. Wetzker, T. Plumbaum, A. Korth, C. Bauckhage, T. Alpcan, and F. Metze, "Detecting trends in social bookmarking systems using a probabilistic generative model and smoothing," in *ICPR '08*, 2008, pp. 1–4.

[27] R. Wetzker, A. Said, and C. Zimmermann, "Understanding the user: Personomy translation for tag recommendation," in *ECML PKDD Discovery Challenge '09*, vol. 497, 2009, pp. 275–284.

[28] R. Wetzker, C. Zimmermann, C. Bauckhage, and S. Albayrak, "I tag, you tag: translating tags for advanced user models," in *WSDM '10*, 2010, pp. 71–80.

[29] Z. Yin, R. Li, Q. Mei, and J. Han, "Exploring social tagging graph for web object classification," in *KDD '09*, pp. 957–966.

[30] Z.-K. Zhang, T. Zhou, and Y.-C. Zhang, "Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs," *Physica A: Statistical Mechanics and its Apllications*, vol. 389, no. 1, pp. 179–186, 2010.

[31] D. Zhou, J. Bian, S. Zheng, H. Zha, and C. L. Giles, "Exploring social annotations for information retrieval," in *WWW '08*, pp. 715–724.