Personalized Documents Ranking With Social Contextualization

Mohamed Reda Bouadjenek[1*], Hakim Hacid[2**], and Mokrane Bouzeghoub[1]

[1] PRiSM Laboratory, Versailles University
{reda.bouadjenek, mokrane.bouzeghoub}@prism.uvsq.fr
[2] SideTrade, 114 Rue Gallieni, 92100 Boulogne-Billancourt, France
hhacid@sidetrade.com

Abstract. We present in this paper a contribution to IR modeling by proposing a new ranking function for documents while considering the social dimension of the Web. This social dimension is any social information that surrounds documents along with the social context of users. Currently, our approach relies on folksonomies for extracting these social contexts, but it can be extended to use any social meta-data, e.g. comments, ratings, tweets, etc. The evaluation performed on our approach shows its benefits for personalized search with respect to the closest state of the art methods.

1 Introduction

Nowadays, the Web is becoming more and more complex with the socialization and interaction between individuals and objects. This evolution is known as social Web, which includes linking people through the World Wide Web. This is mainly done through platforms such as *Facebook*, *Twitter*, or *YouTube*, where users can comment, spread, share and tag information and resources. The social Web leads to facilitate the implication of users in the enrichment of the social context of web pages. Especially, it allows users to freely tag web pages with annotations. These annotations can be easily used to get an intuition about the content of web pages to which they are related. Hence, several research works ([21,6,8,4]) reported that adding tags to the content of a document enhances the search quality, as they are good summaries for documents. In particular, tags are useful for documents that contain few terms where a simple indexing strategy is not expected to provide a good retrieval performances (e.g. the *Google homepage*[3]).

---

[*] This work has been mainly done when the author was a PhD student at Bell Labs France, Centre de Villarceaux.

[**] This work has been mainly done when the author was a research scientist at Bell Labs France, Centre de Villarceaux.

[3] http://www.google.com/ There are only few terms on the page itself but a thousands of annotations available on *delicious* are associated to it. Eventually, the social information of the *Google* homepage is more useful for indexing.

In such a context, classic model of Information Retrieval (IR) should be adapted by considering (i) the social context that surrounds web pages and resources, e.g. their annotations, their associated comments, their ratings, etc. and (ii) the social context of users, e.g. their used tags, their comments, their trustworthiness, etc. Exploiting social information has a number of advantages (for IR in particular). First, feedback information in social networks is provided directly by the user. Hence, accurate information about the user interest can be learned because people actively express their opinions on social platforms. Second, exploring published information doesn't violate user privacy, since the primary goal for most of people is to share information. Finally, social resources are often publicly accessible, as most of social networks provide APIs to access their data (even if often, a contract must be established before any use).

In this paper, we are interested in improving the IR model. Especially, we propose a new ranking function for ranking documents while considering the social context of the Web. The approach we are proposing relies on social annotations as a source of social information, which are associated to documents in bookmarking systems.

1.1 Background

Social bookmarking websites are based on the techniques of *social tagging or collaborative tagging*. The principle behind social bookmarking platforms is to provide the user with a means to annotate resources on the Web, e.g. URIs in *delicious*, videos in *youtube*, images in *flickr*, or academic papers in *CiteULike*. These annotations (also called tags) can be shared with others. This unstructured (or better, free structured) approach to classification with users assigning their own labels is often referred to as a *folksonomy [9]*. A folksonomy is based on the notion of bookmark, which is formally defined as follow:

Definition 1. *Let U, T, R be respectively the set of Users, Tags and Resources. A bookmark is a triplet (u,t,r) such as $u \in U, t \in T, r \in R$, which represents the fact that the user u has annotated the resource r with the tag t.*

Then, a folksonomy is formally defined as follow:

Definition 2. *Let U, T, R be respectively the set of Users, Tags and Resources. A folksonomy $\mathbb{F}(U, T, R)$ is a subset of the Cartesian product $U \times T \times R$ such that each triple $(u, t, r) \in \mathbb{F}$ is a bookmark.*

A folksonomy can be represented by a tripartite-graph where each ternary edge represents a bookmark. In particular, the graph representation of the folksonomy $\mathbb{F}$ is defined as a tripartite graph $\mathcal{G}(V, E)$ where $V = U \cup T \cup R$ and $E = \{(u, t, r) | (u, t, r) \in \mathbb{F}\}$. Figure 1 shows example of a folksonomy with seven bookmarks.

1.2 Problem definition

The problem we are addressing can be formalized as follows: Let consider a folksonomy $\mathbb{F}(U, T, R)$ whose a user $u \in U$ submits a query q to a search engine.

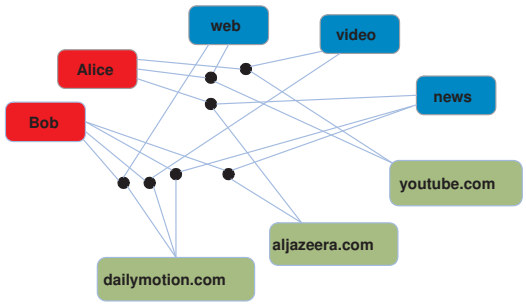


Fig. 1: Example of a folksonomy.

We would like to re-rank the set of documents $R_q \subseteq R$ (or resources) that match q, such that relevant documents for u are highlighted and pushed to the top for maximizing his satisfaction and personalizing the search results. The ranking follows an ordering $\tau = [r_1 \geq r_2 \geq \cdots \geq r_k]$ in which $r_k \in R$ and the ordering relation is defined by $r_i \geq r_j \Leftrightarrow Rank(r_i, q, u) \geq Rank(r_j, q, u)$, where $Rank(r, q, u)$ is a ranking function that quantify similarity between the query and the resource w.r.t the user [16].

1.3 Contributions and paper organization

In this context of social Web, we propose the following contributions: (1) A ranking function that leverages the social context of the Web. (2) Two methods for weighing user profiles and the social representations of documents. (3) An intensive evaluation of our approach and a comparison with the closest works on a large public dataset.

The rest of this paper is organized as follows: in Section 2 we present the related works and we position our method consequently. Section 3 introduces our approach for ranking documents. The different experiments are discussed in Section 4. Finally, we conclude and provide some future directions in Section 5.

2 Related Work

We distinguished two categories for social results re-ranking that differ in the way they use social information. The first category uses social information by adding a social relevance to documents while the second use it for personalization.

2.1 Re-ranking using social relevance

Several approaches have been proposed to improve document re-ranking using social relevance. Social relevance refers to information socially created that characterizes a document from a point of view of its interest, i.e. its general interest, its popularity, etc. Two formal models for folksonomies and ranking algorithm

called *folkRank and Social PageRank* are defined in [10] and [1] respectively. Both are an extension of the well-known *PageRank* algorithm adapted for the generation of rankings of entities within folksonomies. In the same spirit, Takahashi et al. [15] propose *S-BIT* and *FS-BIT*, an extension of the well-known HITS [11] approach. Finally, Yanbe et al. [20] proposed *SBRank*, which indicates how many users bookmarked a page, and use the estimation of *SBRank* as an indicator of web search.

2.2 Personalized re-ranking

In general, users have different interests, different profiles, and different habits. Hence, in an IR system, providing the same documents sorted in the same way is not really suitable since relevance judgment is user-dependent [14]. Therefore, a personalized function to sort documents differently according to the each user is expected to improve search results.

Several approaches have been proposed to personalize ranking of search results using social information [7,16,17,19]. Almost all these approaches are in the context of folksonomy and follow the same idea that the ranking score of a document d retrieved when a user u submits a query q is driven by: (i) a term matching process, which calculates the similarity between q and the textual content of d to generate a user unrelated ranking score; and (ii) an interest matching process, which calculates the similarity between u and d to generate a user related ranking score. Then a merge operation is performed to generate a final ranking score based on the two previous ranking score.

The approach we are proposing is part of this initiative. However, we enhance the ranking process by considering a new aspect, which is *the social matching score*. It measures the similarity between the query and the social representation of documents. Details of our ranking function are given in the next section.

3 A ranking function for personalized search

In this Section, we first define our ranking function, then we present the methods used for modeling and weighting the social representation of documents and user profiles.

3.1 Ranking for personalized search

On the one hand, we believe that a matching score between a document d and a query q should be based on (i) a textual matching score, and (ii) a social matching score. The textual matching score expresses the similarity between the textual content of d and q. The social matching score expresses how similar the social representation of d is, for q. This social representation is based on the annotations associated to d modeled and weighted as described in Section 3.2. More formally, in this paper, we consider this two ranking scores as an independent evidence,

and we propose to merge them using the *Weighted Borda Fuse*. This merge is summarized in Equation 1:

$$Score(q,d) = \beta \times Sim(\overrightarrow{q}, \overrightarrow{d}) + (1-\beta) \times Sim(\overrightarrow{q}, \overrightarrow{S_d}) \quad (1)$$

where β is a weight that satisfies $0 \leq \beta \leq 1$, $Sim(\overrightarrow{q}, \overrightarrow{d})$ denotes the textual matching score between d and q (computed using the *Apache Lucene*[4] search engine in our implementation), $\overrightarrow{S_d}$ is the vector that models the social representation of the document d, and $Sim(\overrightarrow{q}, \overrightarrow{S_d})$ denotes the social matching score between d and q. Inspired by the Vectorial Space Model (VSM), we compute this similarity using the cosine measure as follows:

$$Sim(\overrightarrow{q}, \overrightarrow{S_d}) = \frac{\overrightarrow{q} \bullet \overrightarrow{S_d}}{|\overrightarrow{q}| \times |\overrightarrow{S_d}|} \quad (2)$$

On the other hand, in the non-personalized search engines, the relevance between a query and a document is assumed to be only based on the textual content of the document. However, as relevance is actually relative for each user [14], considering only a matching between a query and documents is not enough to generate satisfactory search results. Thus, we propose to estimate the interest of a user u to a document d by computing a similarity between the profile of u and the social representation of d. Then, we propose to merge this interest value to the previous ranking score computed in Equation 1 for computing the matching score of a document to a query with respect to a user. Formally, the ranking score of a document d that potentially match the query q issued by a user u is computed as follows:

$$Rank(d,q,u) = \gamma \times Sim(\overrightarrow{p_u}, \overrightarrow{S_d}) + (1-\gamma) \times \left[\beta \times Sim(\overrightarrow{q}, \overrightarrow{d}) + (1-\beta) \times Sim(\overrightarrow{q}, \overrightarrow{S_d})\right] \quad (3)$$

where, γ is the weight that satisfies $0 \leq \gamma \leq 1$, and $Sim(\overrightarrow{p_u}, \overrightarrow{S_d})$ is the similarity between the profile of u and the social representation of d. This similarity quantifies the interest of u to d and is computed using the cosine measure as follows:

$$Sim(\overrightarrow{p_u}, \overrightarrow{S_d}) = \frac{\overrightarrow{p_u} \bullet \overrightarrow{S_d}}{|\overrightarrow{p_u}| \times |\overrightarrow{S_d}|} \quad (4)$$

At the end of this process, we obtain a list of re-ranked documents according to: (i) a textual content matching score of documents and the query, (ii) a social matching score of documents and the query, and (iii) the social interest score of the user to documents. Finally, the top ranked documents are formatted for presentation to the user.

In the next two subsections, we present two methods to weight and estimate the social document representation and the user interest vectors.

[4] http://lucene.apache.org/

3.2 Social document modeling

In this paper, the social representations of documents are estimated by their social annotations and modeled as in the VSM. Hence, if we consider web pages as documents and annotations as terms, the above setting is right for the VSM. Even if the VSM has been developed a long time ago, it has shown its effectiveness for IR and remains very competitive and challenging. One of the key points in the VSM is the weighting of terms. Hence, we first propose to simply weight annotations using the *tf-idf* measure as follows:

$$w_t = tf_t \times log(\frac{\mid R \mid}{\mid R_t \mid}) \tag{5}$$

where tf_t denotes the tag frequency, $\mid R \mid$ denotes the total number of web pages in the whole collection and $\mid R_t \mid$ denotes the number of web page tagged with t.

Beside this, the BM25 weighting scheme is a more sophisticated alternative, which represents state-of-the-art weighting functions used in IR. It is computed as follows:

$$w_t = log(\frac{\mid R \mid - \mid R_t \mid + 0.5}{\mid R_t \mid + 0.5}) \times \frac{tf_t \times (k_1 + 1)}{tf_t + k_1 \times (1 - b + b \times \frac{dl}{avgdl})} \tag{6}$$

where k_1 and b are free parameters set to 2 and 0.75 respectively, dl denotes number of annotations associated to the web page and $avgdl$ denotes the average number of annotations associated to web pages the collection.

3.3 User modeling

Folksonomies have proven to be a valuable knowledge for user profiling [7,13,16,19]. Personalization allows discriminating between individuals by emphasizing on their specific domains of interest and their preferences. Several techniques exist to provide personalized services among which the user profiling. The user profile is a collection of personal information associated to a specific user that enables to capture his interests. In this paper and in the context of folksonomies, we define a user profile as follow:

Definition 3. *Let U, T, R be respectively the set of Users, Tags and Resources of a folksonomy $\mathbb{F}(U, T, R)$. A profile assigned to a user $u \in U$, is modeled as a weighted vector $\overrightarrow{p_u}$ of m dimensions, where each dimension represents a tag the user employed in his tagging actions. More formally, $\overrightarrow{p_u} = \{w_{t_1}, w_{t_2}, ..., w_{t_m}\}$ such that $t_m \in T \wedge (\exists r \in R \mid (u, t_m, r) \in \mathbb{F})$, and w_{t_m} is the weight of t_m.*

At this point, the main challenge is *how to define the weight of each dimension in the user profile?* Hence, we fist propose to use an adaptation of the well-known *tf-idf* measure to estimate this weight. Formally, we define the weight w_{t_i} of

the term t_i in a user profile as the *user term frequency, inverse user frequency (utf-iuf)*, which is computed as follows:

$$w_t = utf_t \times log\left(\frac{\mid U \mid}{\mid U_t \mid}\right) \tag{7}$$

where utf_u is the user term frequency, i.e. the number of time the user u used the tag t, $\mid U \mid$ is the total number of users in the folksonomy, and $\mid U_t \mid$ is the number of users who have used the term t_i.
Similarly, we can adapt the BM25 weighting scheme to weight the user profiles. It is computed as follows:

$$w_t = log(\frac{\mid U \mid - \mid U_t \mid +0.5}{\mid U_t \mid +0.5}) \times \frac{utf_t \times (k_1+1)}{utf_t + k_1 \times (1—b + b \times \frac{dl_u}{avgdl_u})} \tag{8}$$

where k_1 and b are free parameters set to 2 and 0.75 respectively, dl_u denotes number of annotations used by u and $avgdl_u$ denotes the average number of annotations used by users in the collection.

In summary, our ranking function for ranking documents that match a query with respect to a user takes into account: (i) the textual content of documents, (ii) their social context, and (iii) the social context of the user by defining a profile and estimating his interest. The social representations of documents and the user profiles are modeled as vectors, and we proposed two methods for weighting these vectors based on state of the art weighting schemes, i.e. *tf-idf* and *BM25*.

4 Evaluation

In this section, we describe the dataset we used, the evaluation methodology and the evaluations we have performed.

4.1 Dataset

We have selected a *delicious* dataset to perform an off-line evaluation, which is public, described and analyzed in [18][5]. Before the experiments, we performed five data preprocessing tasks: (1) We remove manually several annotations that are too personal or meaningless, e.g. “toread”, “Imported IE Fa-vorites”, “system:imported”, etc. (2) Although the annotations from delicious are easy for users to read and understand, they are not designed for machine use. For example, some users may concatenate several words to form an annotation such as “java.programming” or “java/programming”. We tokenize this kind of annotations before using them in the experiments. (3) The list of terms undergoes a stemming by means of the Porter’s algorithm in such a way to eliminate the differences between terms having the same root. (4) We downloaded all the available

[5] http://data.dai-labor.de/corpus/delicious/

web pages while removing those which are no longer available using the *cURL* command line tool. (5) Finally, we removed all the non-english web pages using *Apache Tika* toolkit. Table 1 gives a description of the resulted dataset after our cleansing:

Table 1: Details of the delicious dataset

Bookmarks	Users	Tags	Web pages	Unique terms
9 675 294	318 769	425 183	1 321 039	12 015 123

The resulted dataset still has the same properties, i.e. it is very sparse and follows a long tail distribution [18].

4.2 Evaluation methodology

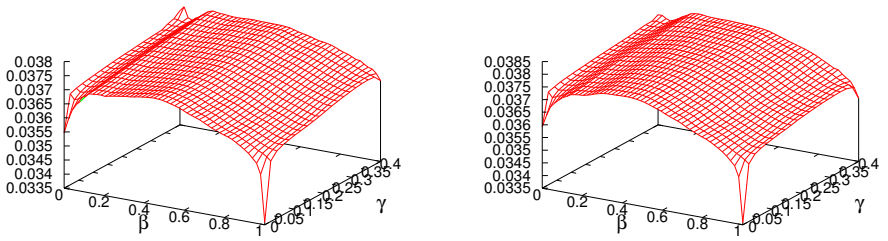
Making evaluations for personalized search is a challenge since relevance judgments can only be assessed by end-users themselves [7]. This is difficult to achieve at a large scale. However, different efforts [12,3,4] state that the tagging behavior of a user of a folksonomy closely reflects his behavior of search on the Web. In other words, if a user tags a document d with a tag t, he will choose to access the document d if it appears in the result obtained by submitting t as query to the search engine. Thus, we can easily state that any bookmark (u, t, r) that represents a user u who bookmarked a resource r with tag t, can be used as a test query for evaluations. The main idea of these experiments is based on the following assumption:

Assumption 1 *For a personalized query $q = \{t\}$ issued by user u with query term t, the relevant documents are those tagged by u with t.*

Hence, for each evaluation, we randomly select 2000 pairs (u, t), which are considered to form a personalized query set. For each corresponding pair (u, t), we remove all the bookmarks $(u, t, r) \in \mathbb{F}, \forall r \in R$ in order to not promote the resource r (or document) in the results obtained by submitting t as a query in our algorithm and the considered baselines. By removing these bookmarks, the results should not be biased in favor of documents that simply tend to return tagged documents and making comparisons to the baseline uninformative. For each pair, the user u sends the query $q = \{t\}$ to the system. Then, we retrieve and rank all the documents that match this query using our approach or a specific baseline, where documents are indexed based on their textual content using the *Apache Lucene.* Finally, according to the previous assumption, we compute the Mean Average Precision (MAP) and the Mean Reciprocal Rank (MRR) over the 2000 queries. The random selection was carried out 10 times independently, and we report the average results.

4.3 Evaluation of the parameters

In this Section, we propose a parameter estimation that aims to provide insights regarding the different values of the parameters used in our approach as well as their potential impact on the system. Our approach has two parameters that can be tuned (γ and β) and two weighting models. Note that each time, we use either the *tf-idf* weighting model for weighting both the social representations of documents and the user profiles or the *BM25* weighting model, i.e. we do not merge the two weighting models. Figure 2 shows the MAP obtained for different values of γ and β and our two weighting models. We vary γ from 0 to 0.4 to better show the impact of β, i.e. for high values of γ, β has a very low impact according to our ranking function of Equation 3.



(a) tf-idf based ranking

(b) BM25 based ranking

Fig. 2: MAP for different values of β and γ using the different weighting models.

First, according to Figure 2, the optimal performance is achieved for $\beta \in [0.2, 0.6]$ for the different values of γ. This shows that both the textual matching score part and the social matching score part are important and are complementary. Second, Figure 2 shows that the behavior of our ranking function seems to be the same for our two ranking models while varying γ and β. Finally, even if the *BM25* weighting model improves better the performance than the *tf-idf* weighting model, our ranking function still doesn't depends on the weighing model.

In the next section, we present the results of the comparison of our approach with several state of the art approaches.

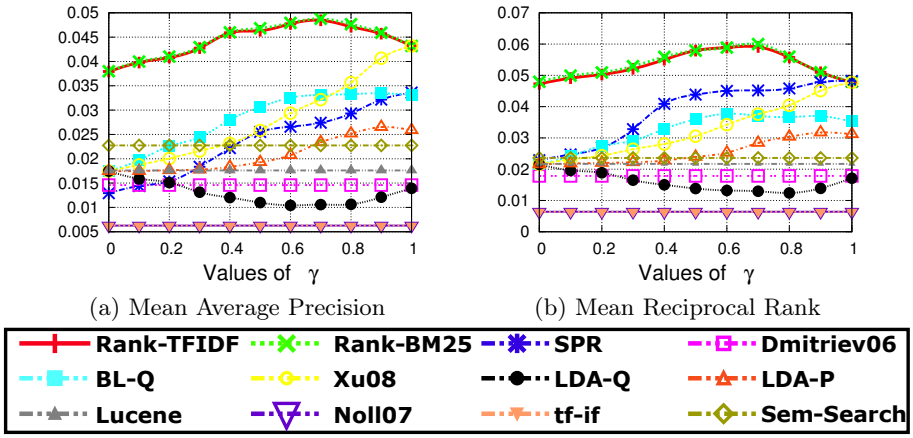
4.4 Comparison with baselines

We compare our approach to several baselines, in which the social enhancement score is merged with the textual based matching score using the *Weighted Borda Fuse (WBF)* with a γ parameter. The baselines are summarized and described in Table 2.

Table 2: Summary of the baselines.

		Baseline	Description
Non-personalized approaches	1	**SPR [1]**	SocialPageRank (SPR), which captures the popularity (quality) of web pages using folksonomies.
	2	**Dmitriev06 [8]**	Combine the annotations with the content of documents to produce a new index.
	3	***B*L-Q**	This approach use a query based ranking function where a similarity between a document and a query is computed by merging the textual based matching score and a social based matching score only. The social representation of each document is based on all its annotations weighted using the *tf-idf* measure.
	4	**Lucene**	This approach represents the Lucene naive score.
	5	**LDA-Q**	Using LDA [5], we model queries and documents. Then, for each document that match a query, we compute a similarity between its topic and the topic of the query using the cosine measure. The obtained value is then merged with the textual ranking score.
Personalized approaches	6	**Xu08 [19]**	This approach use a profile based ranking function, where documents are weighted using the *tf-idf*.
	7	**Noll07 [13]**	The approach considers only a user interest matching between a user and a document. It does not make use of the user and document length normalization factors, and only uses the user tag frequency values. The authors normalize all document tag frequencies to 1, since they want to give more importance to the user profile.
	8	**tf-if [16]**	This approach is an adaptation of **[13].** The main difference is that tf-if incorporate both the user and document tag distribution global importance factors, following the VSM principle.
	9	**Semantic Search [2]**	This approach ranks documents by considering users that hold similar content to the query, i.e., users who used at least one of the query terms in describing their content.
	10	**LDA-P**	Using LDA, we model users and documents. Then, for each document that match a query, we compute a similarity between its topic and the topic of the user profile using the cosine measure. The obtained value is then merged with the textual ranking score.

The obtained results are illustrated in Figure 3, while varying γ. The results show that our approach is much more efficient than all the baselines for our two weighting models and for all the values of γ. Especially, our approach significantly outperform the Xu08 and LDA-P approaches, which we consider as the closest works to our. Hence, we conclude that the personalization efforts introduced by our ranking function bring a considerable improvement to the search quality. We also notice that most of the approach decease their performance for high values



(a) Mean Average Precision (b) Mean Reciprocal Rank

Fig. 3: Comparison with the baseline while varying γ and using the optimal values of the parameters.

of γ. This is certainly due to the fact that they are not designed for personalized search, since these approaches fail in discriminating between users in spite of their preferences.

Finally, we note that the better performances are obtained for $\gamma \in [0.6, 0.8]$, a compromise between the user interest matching score and the query affinity matching score. Although its simplicity, our ranking function is very efficient compared to other state of the art approaches. However, these results should be reinforced using an on-line evaluation to give a better overview of the performance, which is an ongoing work.

5 Conclusion and future work

This paper discusses a contribution to the area of IR modeling while leveraging the social dimension of the web. We proposed a new documents ranking function, which uses social information to enhance and improve web search. The experiments performed show the benefit of our approach while comparing it to the closest works. This method can be improved in different way. First, the temporal dimension of social users' behavior has not been deeply investigated yet in the literature. Considering this dimension is a part of our future work, e.g. considering the evolution of the taste of users in the ranking function. Second, considering a social relevance score factor, which characterizes documents from a point of view of interest, is a possible improvement of our ranking function, e.g. their popularities. Finally, performing an on-line user evaluation in order to validate our results is also an ongoing work.

References

1. S. Bao, G. Xue, X. Wu, Y. Yu, B. Fei, and Z. Su. Optimizing web search using social annotations. In *WWW*, 2007.
2. M. Bender, T. Crecelius, M. Kacimi, S. Michel, T. Neumann, J. X. Parreira, R. Schenkel, and G. Weikum. Exploiting social relations for query expansion and result ranking. In *ICDE Workshops*, 2008.
3. D. Benz, A. Hotho, R. Jäschke, B. Krause, and G. Stumme. Query logs as folksonomies. *Datenbank-Spektrum*, 10:15–24, 2010.
4. K. Bischoff, C. S. Firan, W. Nejdl, and R. Paiu. Can all tags be used for search? In *CIKM*, 2008.
5. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.
6. D. Carmel, H. Roitman, and E. Yom-Tov. Social bookmark weighting for search and recommendation. *The VLDB Journal*, 2010.
7. D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har'el, I. Ronen, E. Uziel, S. Yogev, and S. Chernov. Personalized social search based on the user's social network. In *CIKM*, 2009.
8. P. A. Dmitriev, N. Eiron, M. Fontoura, and E. Shekita. Using annotations in enterprise search. In *WWW*, 2006.
9. T. Hammond, T. Hannay, B. Lund, and J. Scott. Social bookmarking tools : A general review. *D-Lib Magazine*, 11(4), April 2005.
10. A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In Y. Sure and J. Domingue, editors, *The Semantic Web: Research and Applications*, 2006.
11. J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.
12. B. Krause, A. Hotho, and G. Stumme. A comparison of social bookmarking with traditional search. In *ECIR*, 2008.
13. M. G. Noll and C. Meinel. Web search personalization via social bookmarking and tagging. In *ISWC'07/ASWC'07*, 2007.
14. J. Pitkow, H. Schütze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. Personalized search. *Commun. ACM*, 2002.
15. T. Takahashi and H. Kitagawa. A ranking method for web search using social bookmarks. In *DASFAA*, 2009.
16. D. Vallet, I. Cantador, and J. M. Jose. Personalizing web search with folksonomy-based user and document profiles. In *ECIR*, 2010.
17. Q. Wang and H. Jin. Exploring online social activities for adaptive search personalization. In *CIKM*, 2010.
18. R. Wetzker, C. Zimmermann, and C. Bauckhage. Analyzing social bookmarking systems: A del.icio.us cookbook. In *ECAI*, 2008.
19. S. Xu, S. Bao, B. Fei, Z. Su, and Y. Yu. Exploring folksonomy for personalized search. In *SIGIR*, 2008.
20. Y. Yanbe, A. Jatowt, S. Nakamura, and K. Tanaka. Towards improving web search by utilizing social bookmarks. In *ICWE*, 2007.
21. X. Zhang, L. Yang, X. Wu, H. Guo, Z. Guo, S. Bao, Y. Yu, and Z. Su. sdoc: exploring social wisdom for document enhancement in web mining. In *CIKM*, 2009.