Optimizing Web Search Using Social Annotations

Shenghua Bao[1*], Xiaoyuan Wu[1*], Ben Fei[2], Guirong Xue[1], Zhong Su[2], and Yong Yu[1]

[1]Shanghai JiaoTong University
Shanghai, 200240, China

{shhbao, wuxy, grxue, yyu}@apex.sjtu.edu.cn

[2]IBM China Research Lab
Beijing, 100094, China

{feiben, suzhong}@cn.ibm.com

ABSTRACT

This paper explores the use of social annotations to improve web search. Nowadays, many services, e.g. del.icio.us, have been developed for web users to organize and share their favorite web pages on line by using social annotations. We observe that the social annotations can benefit web search in two aspects: 1) the annotations are usually good summaries of corresponding web pages; 2) the count of annotations indicates the popularity of web pages. Two novel algorithms are proposed to incorporate the above information into page ranking: 1) SocialSimRank (SSR) calculates the similarity between social annotations and web queries; 2) SocialPageRank (SPR) captures the popularity of web pages. Preliminary experimental results show that SSR can find the latent semantic association between queries and annotations, while SPR successfully measures the quality (popularity) of a web page from the web users' perspective. We further evaluate the proposed methods empirically with 50 manually constructed queries and 3000 auto-generated queries on a dataset crawled from del.icio.us. Experiments show that both SSR and SPR benefit web search significantly.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval.

General Terms: Algorithms, Experimentation, Human Factors.

Keywords: Social annotation, social page rank, social similarity, web search, evaluation

1. INTRODUCTION

Over the past decade, many studies have been done on improving the quality of web search. Most of them contribute from two aspects: 1) ordering the web pages according to the query-document similarity. State-of-the-art techniques include anchor text generation [21, 28, 34], metadata extraction [37], link analysis [34], and search log mining [10]; 2) ordering the web pages according to their qualities. It is also known as query-independent ranking, or static ranking. For a long time, the static ranking is derived based on link analysis, e.g., PageRank [17], HITS [15]. Recently, the features of content layout, user click-throughs etc. are also explored, e.g., fRank[19]. Given a query, the retrieved results are ranked based on both page quality and query-page similarity.

Recently, with the rise of Web 2.0 technologies, web users with different backgrounds are creating annotations for web pages at an incredible speed. For example, the famous social bookmark web site, del.icio.us [4] (henceforth referred to as "Delicious"), has more than 1 million registered users soon after its third birthday, and the number of Delicious users have increased by more than 200% in the past nine months [13]. Social annotations are emergent useful information that can be used in various ways. Some work has been done on exploring the social annotations for folksonomy [2], visualization [18], semantic web [36], enterprise search [23] etc. However, to the best of our knowledge, little work has been done on integrating this valuable information into web search. How to utilize the annotations effectively to improve web search becomes an important problem.

In this paper, we study the problem of utilizing social annotations for better web search, which is also referred to as "social search" for simplicity. More specifically, we optimize web search by using social annotations from the following two aspects:

- **Similarity ranking**, which means the estimated similarity between a query and a web page. The annotations, provided by web users from different perspectives, are usually good summaries of the corresponding web pages. For example, the top 5 annotations of Amazon's homepage [1] in Delicious are *shopping, books, amazon, music* and *store,* which depict the page or even the whole website exactly. These annotations provide a new metadata for the similarity calculation between a query and a web page. However, for a specific web page, the annotation data may be sparse and incomplete. Therefore, a matching gap exists between the annotations and queries (e.g., between "shop" and "shopping"). How to bridge the gap remains a crucial problem in further improving the similarity ranking. We propose a new similarity estimation algorithm, SocialSimRank (SSR) to address this problem.

- **Static ranking**, the amount of annotations assigned to a page indicates its popularity and implies its quality in some sense, yet traditional static ranking algorithms such as PageRank have no way to measure this kind of quality. For example, thousands of Delicious users collect the popular pattern introduction page[2] as their favorite with a variety of annotations, but this site is given a PageRank

---

[*] Part of Shenghua Bao and Xiaoyuan Wu's work of this paper was conducted in IBM China Research Lab.

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

[2] http://37signals.com/papers/introtopatterns/

of zero by Google. Furthermore, different annotations may have different weights in indicating the popularity of web pages. In the light of the above intuitions, we propose a novel algorithm, SocialPageRank (SPR) to measure the popularity of web pages using social annotations.

For each aspect, we will show one algorithm that is guaranteed to converge. The proposed algorithms are evaluated on a Delicious corpus which consists of 1,736,268 web pages with 269,566 different social annotations. Preliminary experimental results show that SSR can calculate the annotation similarities semantically and SPR successfully depicts the quality of a web page from the web users' perspective. To evaluate their effectiveness for web search, 50 queries and the corresponding ground truths are collected from a group of CS students and 3000 queries are generated from the Open Directory Project [3] . Experiments on two query sets show that both SSR and SPR improve the quality of search results significantly. By further combining them together, the mean average precision of search results can be improved by as much as 14.80% and 25.02% on two query sets, respectively.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 proposes the social search framework with SocialSimRank and SocialPageRank in detail. Section 4 presents the experimental results. Section 5 gives some discussions. Finally, we conclude with Section 6.

2. RELATED WORK

2.1 Research on Web Search
Much work has been done on improving user experience of web search, most of which focuses on ranking search results. We briefly review the related work on similarity ranking and static ranking as follows.

Similarity ranking measures the relevance between a query and a document. Many models have been proposed to estimate the similarity between the query and the document [11]. In modern search engines, several methods have been proposed to find new information as additional metadata to enhance the performance of similarity ranking, e.g., document title [37], anchor text [21, 28, 34], and users' query logs [10]. These methods improved the performance of web search to some extent. For example, Google's search engine [28] took the anchor text as its metadata to improve the performance of search. Fortunately, recent emerging social annotations provide a new resource to calculate the query-document similarity more precisely. We propose a new method, i.e. SocialSimRank for effective use of this new resource.

Since the publication of Brin and Page's paper on PageRank [17], many studies have been conducted in the web community for the static ordering of Web pages. Recently, Richardson *et al.* proposed fRank [19] using features that are independent of the link structure of the Web. PageRank utilizes the link structure of the Web and measures the quality of a page from the page creator's point of view, while fRank utilizes content-layout and user click-though information and captures the preference of both page authors and search engine users. In this paper,

SocialPageRank is proposed to explore static ranking from social annotations and capture the preference of web annotators.

2.2 Research on Social Annotations
Existing research on social annotations includes "folksonomy" [2, 24], visualization [18], emergent semantics [25], semantic web [36], enterprise search [23] etc.

"Folksonomy", a combination of "folk" and "taxonomy", was first proposed by T. V. Wal in a mailing list [12]. Folksonomy was further divided into the narrow (e.g. flickr[4]) and the broad (Delicious) folksonomy in [33]. It provides user-created metadata rather than the professional created and author created metadata [2]. In [24], P. Merholz argued that a folksonomy could be quite useful in that it revealed the digital equivalent of "desire lines". Desire lines were the foot-worn paths that sometimes appeared in a landscape over time. [27] analyzed the structure of collaborative tagging systems as well as their dynamical aspects. Hotho *et al.* proposed Adapted PageRank and FolkRank to find communities within the folksonomy but have not applied them to web search [1]. A general introduction of folksonomy could be found in [6] by E. Quintarelli.

M. Dubinko *et al.* considered the problem of visualizing the evolution of tags [18]. They presented a new approach based on a characterization of the most interesting tags associated with a sliding time interval.

Some applications based on social annotations have also been explored. P. Mika proposed a tripartite model of actors, concepts and instances for semantic emergence [25]. X. Wu *et al.* explored machine understandable semantics from social annotations in a statistical way and applied the derived emergent semantics to discover and search shared web bookmarks [36]. Dmitriev *et al.* lightened the limitation of the amount and quality of anchor text by using user annotations to improve the quality of intranet search [23].

Different from the above work, we investigate the capability of social annotations in improving the quality of web search from the aspects of similarity ranking and static ranking within the Internet environment.

3. SEARCH WITH SOCIAL ANNOTATION
In this section, we introduce the social annotation based web search. An overview is presented in Section 3.1. We discuss SocialSimRank and SocialPageRank in Section 3.2 and 3.3, respectively. In Section 3.4, we describe the social search system utilizing both SSR and SPR.

3.1 Overview
As shown in Figure 1, there are three kinds of users related to the social search, namely *web page creators*, *web page annotators,* and *search engine users*. Obviously, these three user sets can overlap with each other. The different roles they play in web search are as follows:
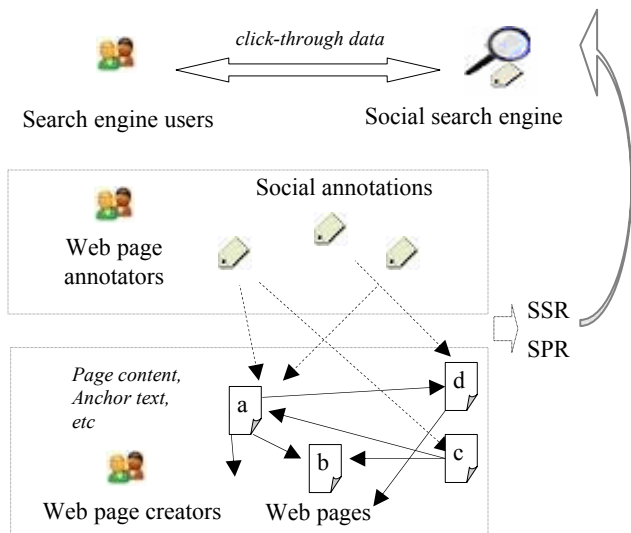

Figure 1. Illustration of social search with SocialSimRank and SocialPageRank

1) *Web page creators* create pages and link the pages with each other to make browsing easy for *web users*. They provide the basis for web search.

2) *Web page annotators* are *web users* who use annotations to organize, memorize and share their favorites online.

3) *Search engine users* use search engines to get information from the web. They may also become *web page annotators* if they save and annotate their favorites from the search results.

Previous work shows that both *web page creators* and *search engine users* contribute to web search a lot. The *web page creators* provide not only the web pages and anchor texts for similarity ranking, but also the link structure for static ranking from the *web page creators'* point of view (e.g. PageRank [17]). Meanwhile, the interaction log of *search engine users* also benefits web search by providing the click-through data, which can be used in both similarity ranking (e.g. IA [10]) and static ranking (e.g. fRank[19]). Here, we are to study how *web page annotators* can contribute to web search.

We observed that the *web page annotators* provide cleaner data which are usually good summarizations of the web pages for users' browsing. Besides, similar or closely related annotations are usually given to the same web pages. Based on this observation, SocialSimRank (SSR) is proposed to measure the similarity between the query and annotations based on their semantic relation. We also observed that the count of social annotations one page gets usually indicates its popularity from the *web page annotators'* perspective and the popularity of web pages, annotations, and annotators can be mutually enhanced. Motivated by this observation, we propose SocialPageRank (SPR) to measure the popularity of web pages from *web page annotators'* point of view.

Figure 1 illustrates the social search engine with SSR and SPR derived from the social annotations. In the following sections, we will discuss the two ranking algorithms in detail.

3.2 Similarity Ranking between the Query and Social Annotations

3.2.1 Term-Matching Based Similarity Ranking

The social annotations are usually an effective, multi-faceted summary of a web page and provide a novel metadata for similarity ranking. A direct and simple use of the annotations is to calculate the similarity based on the count of shared terms between the query and annotations. Letting $q=\{q_1,q_2,\ldots,q_n\}$ be a query which consists of n query terms and $A(p)=\{a_1,a_2,\ldots, a_m\}$ be the annotation set of web page p, Equation (1) shows the similarity calculation method based on the shared term count. Note that $sim_{TM}(q,p)$ is defined as 0 when $A(p)$ is empty.

$$sim_{TM}(q,p) = \frac{|q \cap A(p)|}{|A(p)|}, \qquad (1)$$

Similar to the similarity between query and anchor text, the term-matching based query-annotation similarity may serve as a good complement to the whole query-document similarity estimation. However, some pages' annotations are quite sparse and the term-matching based approach suffers more or less from the synonymy problem, i.e., the query and the annotation may have terms with similar meanings but in different forms. In the next section, we are to solve the synonymy problem by exploring the social annotation structures.

3.2.2 Social Similarity Ranking

Observation 1: *Similar (semantically-related) annotations are usually assigned to similar (semantically-related) web pages by users with common interests. In the social annotation environment, the similarity among annotations in various forms can further be identified by the common web pages they annotated.*
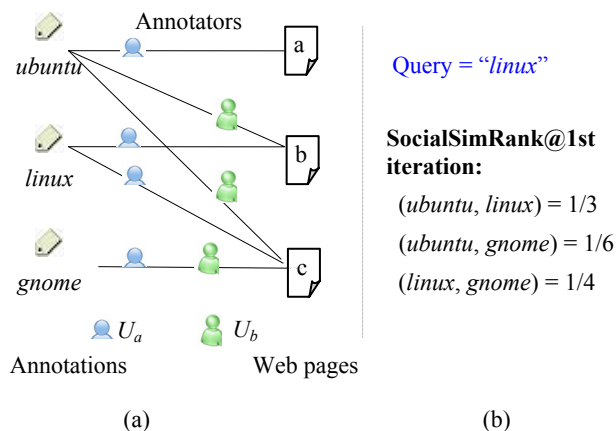

Figure 2. Illustration of SocialSimRank

Assume that there are two annotators (U_a and U_b) as illustrated in Figure 2(a). Given the ubuntu's official website *b*, U_a may prefer using the annotation "*linux*", while U_b would like "*ubuntu*". Thus, "*linux*" and "*ubuntu*" may have some semantic relations connected by their commonly associated page. As for web page *c,* both the annotation "*linux*" and "*gnome*" are given by U_a, then "*linux*" and "*gnome*" should also associate with each other to some degree.

503

In some cases, some pages may contain only the annotation "*ubuntu*" e.g., web page *a*. Then given the query containing "*linux*", the page that only has "*ubuntu*" may be filtered out improperly by simple term-matching method. Even if the page contains both annotations "*ubuntu*" and "*linux*", it is not proper to calculate the similarity between the query and the document using the keyword "*linux*" only. An exploration of similarity between "*ubuntu*" and "*linux*" may further improve the page ranking.

To explore the annotations with similar meanings, we build a bipartite-graph between social annotations and web pages with its edges indicating the user count. Assume that there are N_A annotations, N_P web pages and N_U web users. M_{AP} is the $N_A \times N_P$ association matrix between annotations and pages. $M_{AP}(a_x,p_y)$ denotes the number of users who assign annotation a_x to page p_y. Letting S_A be the $N_A \times N_A$ matrix whose element $S_A(a_i, a_j)$ indicates the similarity score between annotations a_i and a_j and S_P be the $N_P \times N_P$ matrix each of whose element stores the similarity between two web pages, we propose SocialSimRank(SSR), an iterative algorithm to quantitatively evaluate the similarity between any two annotations.

Algorithm 1: SocialSimRank (SSR)

Step 1   *Init:*   *Let* $S_A^{0}(a_i, a_j) = 1$ for each $a_i = a_j$ otherwise 0

$S_P^{0}(p_i, p_j) = 1$ for each $p_i = p_j$ otherwise 0

Step 2   *Do* {

For each annotation pair (a_i, a_j) *do*

$$S_A^{k+1}(a_i, a_j) = \frac{C_A}{|P(a_i)||P(a_j)|}$$
$$\sum_{m=1}^{|P(a_i)|}\sum_{n=1}^{|P(a_j)|} \frac{\min(M_{AP}(a_i,p_m),M_{AP}(a_j,p_n))}{\max(M_{AP}(a_i,p_m),M_{AP}(a_j,p_n))} S_P^k(P_m(a_i),P_n(a_j)) \quad , \quad (2)$$

For each page pair (p_i, p_j) *do*

$$S_P^{k+1}(p_i, p_j) = \frac{C_P}{|A(p_i)||A(p_j)|}$$
$$\sum_{m=1}^{|A(p_i)|}\sum_{n=1}^{|A(p_j)|} \frac{\min(M_{AP}(a_m,p_i),M_{AP}(a_n,p_j))}{\max(M_{AP}(a_m,p_i),M_{AP}(a_n,p_j))} S_A^{k+1}(A_m(p_i),A_n(p_j)) \quad , \quad (3)$$

}*Until* $S_A(a_i, a_j)$ converges.

Step 3   *Output:* $S_A(a_i, a_j)$

Here, C_A and C_P denote the damping factors of similarity propagation for annotations and web pages, respectively. $P(a_i)$ is the set of web pages annotated with annotation a_i and $A(p_j)$ is the set of annotations given to page p_j. $P_m(a_i)$ denotes the *m*th page annotated by a_i and $A_m(p_i)$ denotes the *m*th annotation assigned to page p_i. In our experiments, both C_A and C_P are set to 0.7.

Note that the similarity propagation rate is adjusted according to the number of users between the annotation and web page. Take Equation (2) for an example, the maximal propagation rate can be achieved only if $M_{AP}(a_i, p_m)$ is equal to $M_{AP}(a_j, p_n)$. Figure 2(b) shows the first iteration's SSR result of the sample data where C_A and C_P are set to 1.

The convergence of the algorithm can be proved in a similar way as SimRank [9]. For each iteration, the time complexity of the SSR algorithm is $O(N_A^2 N_P^2)$. Within the data set of our experiment, both the annotation and web page similarity matrices are quite sparse and the algorithm converges quickly. But if the scale of social annotations keeps growing exponentially, the speed of convergence for our algorithms may slow down. To solve this problem, we can use some optimization strategy such as incorporating the minimal count restriction [9] to make the algorithm converge more quickly.

Letting $q=\{q_1,q_2,...,q_n\}$ be a query which consists of *n* query terms and $A(p)=\{a_1,a_2,..., a_m\}$ be the annotation set of web page *p*, Equation (4) shows the similarity calculation method based on the SocialSimRank.

$$sim_{SSR}(q, p) = \sum_{i=1}^{n}\sum_{j=1}^{m} S_A(q_i,a_j), \quad (4)$$

3.3 Page Quality Estimation Using Social Annotations

Existing static ranking methods usually measure pages' quality from the *web page creators'*, or the *search engine users'* point of view. Recall that in Figure 1, the estimation of PageRank [17] is subject to *web creators*, and the fRank [19] is calculated based on both *web page creators* and *search engine users'* activities. The social annotations are the new information that can be utilized to capture the web pages' quality from *web page annotators'* perspective.

3.3.1 SocialPageRank Algorithm
Observation 2: *High quality web pages are usually popularly annotated and popular web pages, up-to-date web users and hot social annotations have the following relations: popular web pages are bookmarked by many up-to-date users and annotated by hot annotations; up-to-date users like to bookmark popular pages and use hot annotations; hot annotations are used to annotate popular web pages and used by up-to-date users.*

Based on the observation above, we propose a novel algorithm, namely SocialPageRank (SPR) to quantitatively evaluate the page quality (popularity) indicated by social annotations. The intuition behind the algorithm is the mutual enhancement relation among popular web pages, up-to-date web users and hot social annotations. Following, the popularity of web pages, the up-to-dateness of web users and the hotness of annotations are all referred to as *popularity* for simplicity.

Assume that there are N_A annotations, N_P web pages and N_U web users. Let M_{PU} be the $N_P \times N_U$ association matrix between pages and users, M_{AP} be the $N_A \times N_P$ association matrix between annotations and pages and M_{UA}, the $N_U \times N_A$ association matrix between users and annotations. Element $M_{PU}(p_i,u_j)$ is assigned with the count of annotations used by user u_j to annotate page p_i. Elements of M_{AP} and M_{UA} are initialized similarly. Let P_0 be the vector containing randomly initialized SocialPageRank scores. Details of the SocialPageRank algorithm are presented as follows.

Algorithm 2: SocialPageRank (SPR)

Step 1   *Input:*

Association matrices M_{PU}, M_{AP}, and M_{UA} and the random initial SocialPageRank score P_0

Step 2   *Do:*

$$U_i = M_{PU}^T \cdot P_i \qquad (5.1)$$

$$A_i = M_{UA}^T \cdot U_i \qquad (5.2)$$

$$P_i' = M_{AP}^T \cdot A_i \qquad (5.3) \qquad (5)$$

$$A_i' = M_{AP} \cdot P_i' \qquad (5.4)$$

$$U_i' = M_{UA} \cdot A_i' \qquad (5.5)$$

$$P_{i+1} = M_{PU} \cdot U_i' \qquad (5.6)$$

Until P_i converges.

Step 3:   *Output:*

P^*: the converged SocialPageRank score.

Step 1 does the initialization. In Step 2, P_i, U_i, and A_i denote the popularity vectors of pages, users, and annotations in the ith iteration. P_i', U_i', and A_i' are intermediate values. As illustrated in Figure 3, the intuition behind Equation (5) is that the users' popularity can be derived from the pages they annotated (5.1); the annotations' popularity can be derived from the popularity of users (5.2); similarly, the popularity is transferred from annotations to web pages (5.3), web pages to annotations (5.4), annotations to users (5.5), and then users to web pages again (5.6). Finally, we get P^* as the output of SocialPageRank (SPR) when the algorithm converges. Sample SPR values are given in the right part of Figure 3.
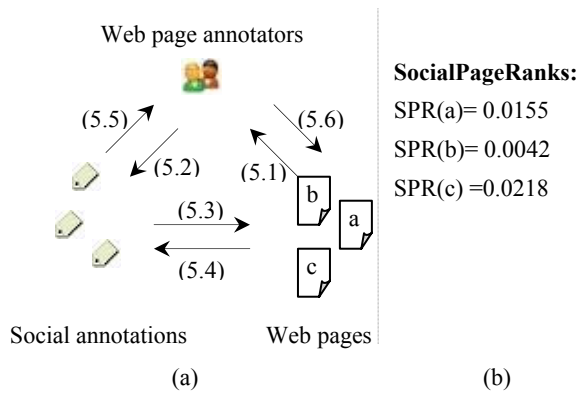

Figure 3. Illustration of quality transition between the users, annotations, and pages in the SPR algorithm

In each iteration, the time complexity of the algorithm is $O(N_U N_P + N_A N_P + N_U N_A)$. Since the adjacency matrices are very sparse in our data set, the actual time complexity is far lower. However, in Web environment, the size of data are increasing at a fast speed, and some acceleration to the algorithm (like [7] for PageRank) should be developed.

3.3.2 Convergence of SPR Algorithm
Here, we give a brief proof of the convergence of the SPR algorithm. It can be derived from the algorithm that:

$$P_{i+1} = (M \cdot M^T) \cdot P_i = (M \cdot M^T)^{i+1} \cdot P_0, \qquad (6)$$

where

$$M = M_{PU} \cdot M_{UA} \cdot M_{AP}. \qquad (7)$$

A standard result of linear algebra (e.g. [8]) states that if M_s is a symmetric matrix, and v is a vector not orthogonal to the principal eigenvector of the matrix $\omega_1(M_s)$, then the unit vector in the direction of $(M_s)^k v$ converges to $\omega_1(M_s)$ as k increases without bound. Here MM^T is a symmetric matrix and P_0 is not orthogonal to $\omega_1(MM^T)$, so the sequence P_i converges to a limit P^*, which signals the termination of the SPR algorithm.

3.4 Dynamic Ranking with Social Information

3.4.1 Dynamic Ranking Method
Due to the large number of features, modern web search engines usually rank results by learning a rank function. Many methods have been developed for automatic (or semi-automatic) tuning of specific ranking functions. Previous work estimates the weights through regression [26]. Recent work on this ranking problem attempts to directly optimize the ordering of the objects [3, 22, 32].

As discussed in [5], there are generally two ways to utilize the explored social features for dynamic ranking of web pages: (a) treating the social actions as independent evidence for ranking results, and (b) integrating the social features into the ranking algorithm. In our work, we incorporate both similarity and static features exploited from social annotations into the ranking function by using RankSVM [32].

3.4.2 Features
We divided our feature set into two mutually exclusive categories: query-page similarity features and page's static features. Table 1 describes each of these feature categories in detail.

Table 1. Features in social search

A: query-document features	
DocSimilarity	Similarity between query and page content
TermMatching (TM)	Similarity between query and annotations using the term matching method.
SocialSimRank (SSR)	Similarity between query and annotations based on SocialSimRank.
B: document static features	
GooglePageRank (PR)	The web page's PageRank obtained from the Google's toolbar API.
SocialPageRank (SPR)	The popularity score calculated based on SocialPageRank algorithm.

4. EXPERIMENTAL RESULTS

4.1 Delicious Data
There are many social bookmark tools on Web [30]. For the experiment, we use the data crawled from Delicious during May, 2006, which consists of 1,736,268 web pages and 269,566 different annotations.

Although the annotations from Delicious are easy for human users to read and understand, they are not designed for machine use. For example, users may use compound annotations in various

forms such as *java.programming* or *java/programming*. We split these annotations into standard words with the help of WordNet [35] before using them in the experiments.

4.2 Evaluation of Annotation Similarities

In our experiments, the SocialSimRank algorithm converged within 12 iterations. Table 2 shows the selected annotations from four categories, together with their top 4 semantically related annotations. With the explored similarity values, we are able to find more semantically related web pages as shown later.

Table 2. Explored similar annotations based on SocialSimRank

Technology related:	
dublin	metadata, semantic, standard, owl
debian	distribution, distro, ubuntu, linux
Economy related:	
adsense	sense, advertise, entrepreneur, money
800	number, directory, phone, business
Entertainment related:	
album	gallery, photography, panorama, photo
chat	messenger, jabber, im, macosx
Entity related:	
einstein	science, skeptic, evolution, quantum
christian	devote, faith, religion, god

4.3 Evaluation of SPR Results

We obtained each web page's SPR score using the algorithm described in section 3.3. In our experiments, the algorithm converged within 7 iterations. Each page's PageRank was also extracted from the Google's toolbar API during July, 2006. Hereafter, we use PageRank to depict the extracted Google's PageRank by default.

4.3.1 SPR vs. PageRank (Distribution Analysis)

Figure 4 shows the average counts of annotations and annotators over web pages with different PageRank values, and for the *Unique Annotation* line, the value means the count of annotations that are different with each other. From the figure, we can conclude that in most cases, the page with a higher PageRank is likely to be annotated by more users with more annotations.
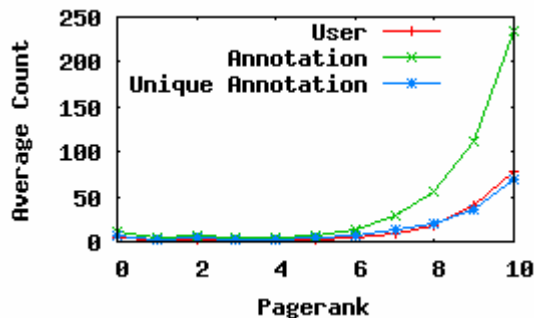

Figure 4. Average count distribution over PageRank

To further investigate the distribution, counts of annotations and annotators for all collected pages with different PageRank values are given in Figure 5(a). It is easy to see that the pages with each PageRank value diversify a lot on the number of annotations and users. Web pages with a relatively low PageRank may own more annotations and users than those who have higher PageRank. For example, some pages with PageRank 0 have more users and annotations than those who have PageRank 10.

We applied the SPR algorithm to the collected data. For easy understanding, SPR is normalized into a scale of 0-10 so that SPR and PageRank have the same number of pages in each grade from 0 to 10. Figure 5(b) shows the detailed counts of annotations and users of pages with different SocialPageRank values. It is easy to see that SocialPageRank successfully characterizes the web pages' popularity degrees among web annotators.
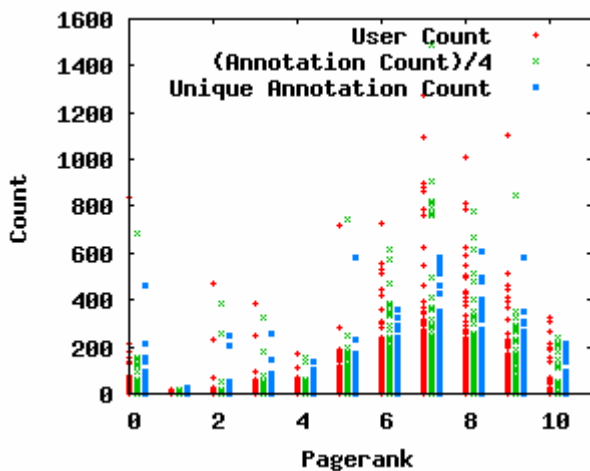
(a) Annotations and users over PageRank
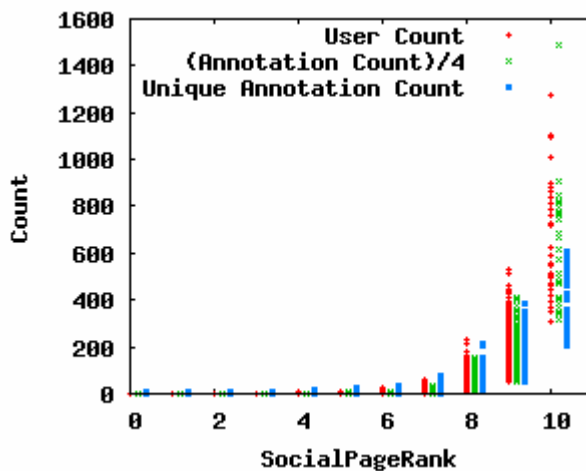
(b) Annotations and users over SocialPageRank

Figure 5. Distribution analysis of social annotations

506

4.3.2 SPR vs. PageRank (Case Study)

Table 3 shows 8 case studies of PageRank vs. SPR. Some web sites have high SocialPageRank but low PageRank, e.g., http://37signals.com/papers/introtopatterns/; or vice versa, e.g., www.lcs.mit.edu. Some web sites are both popular for web creators and web users, e.g., www.w3.org, and some are both not, e.g., www.cientologia-lisbon.org. From the case studies we conclude that the *web creators'* preferences do differ from the *web users'* (*web page annotators*) which are successfully characterized by SPR.

Table 3. Case studies of SPR vs. PageRank

Web Pages	PR	SPR
http://www.sas.calpoly.edu/asc/ssl/procrastination.html	0	10
http://37signals.com/papers/introtopatterns/	0	10
http://www.lcs.mit.edu/	10	0
http://www.macromedia.com/software/coldfusion/	10	0
http://www.w3.org/	10	10
http://www.nytimes.com/	10	10
http://www.cientologia-lisbon.org/	0	0
http://users.tpg.com.au/robotnik/	0	0

4.4 Dynamic Ranking with Social Annotation

4.4.1 Query Set

We use the data described in Section 4.1 to evaluate the effectiveness of integrating social annotations into dynamic ranking. Both manually and automatically constructed query sets are used.

1) Manual query set (**MQ**): we asked a group of CS students to help us collect 50 queries and their corresponding ground truths. Most of the 50 queries are about computer science since most of the Delicious documents we crawled are about computer science. We also selected some queries about other fields to guarantee the diversity of queries, e.g., *NBA Houston Rockets* and *Martin Luther King*. The ground truth of each query was built by browsing top 100 documents returned by Lucene search engine. Finally, the queries were associated with 304 relevant documents in total. The average length of the manual queries is 3.540.

2) Automatic query set (**AQ**): we automatically extracted 3000 queries and their corresponding ground truths from the ODP as follows. First, we merged the Delicious data with ODP and discarded ODP categories that contain no Delicious URLs. Second, we randomly sampled 3000 ODP categories, extracted the category paths as the query set and extracted the corresponding web pages as the ground truths. Note that the term *TOP* in the category path was discarded. For example, the category path *TOP/Computers/Software /Graphics* would be extracted as the query *Computers Software Graphics*. Finally, we got 3000 queries with 14233 relevant documents. The average length of automatic queries is 7.195.

4.4.2 System Setup

In our experiment, the "*DocSimilarity*" is taken as the baseline. This similarity is calculated based on the BM25 formula [29], whose term frequency component is implemented as follows:

$$TF(t,d) = \frac{k * f(t,d)}{k * ((1-b) + b * doclen/avgdoclen) + f(t,d)} , \quad (8)$$

where $f(t,d)$ means the term count of t in document d. In the experiment, k and b are set to 1 and 0.3, respectively.

To evaluate the features proposed in Table 1, we first extracted the top 100 documents returned by BM25 for each query and then created five different random splits of 40 training and 10 testing queries on MQ set, and 2,400 training and 600 testing queries on AQ set. The splits were done randomly without overlaps. Then, RankSVM is applied to learn weights for all the features described in Table 1. The default regularization parameter is set to 0.0006.

4.4.3 Evaluation Metrics

We evaluate the ranking algorithms over two popular retrieval metrics, namely *Mean Average Precision* (*MAP*), and *Normalized Discounted Cumulative Gain* (*NDCG*). Each metric focuses on one aspect of the search performance, as described below.

- **Mean Average Precision**: We mainly used MAP to evaluate search performance. It is defined as the mean of average precision over queries. The average precision for each query is defined as:

$$average\_precision = \sum_{j=1}^{M} p(j) * \Delta r(j), \quad (9)$$

where $p(j)$ denotes the precision over the top j results, and $\Delta r(j)$ is the change in recall from j-1 to j.

- **NDCG at K**: NDCG is a retrieval measure devised specifically for web search evaluation [16]. It is well suited to web search evaluation as it rewards relevant documents that are top-ranked more heavily than those ranked lower. For a given query q, the ranked results are examined in a top-down fashion, and NDCG is computed as:

$$N_q = M_q \sum_{j=1}^{K} (2^{r(j)} - 1) / \log(1 + j), \quad (10)$$

where M_q is a specially calculated normalization constant for making a perfect ordering obtain an NDCG value of 1; and each $r(j)$ is an integer relevance label (0="*Irrelevant*" and 1="*Relevant*") of the result returned at position j.

4.4.4 Dynamic Ranking Using Social Similarity

Figure 6 shows the comparison between NDCG of the term-matching and social-matching on the AQ set. We can easily find that using term-matching to utilize social annotations does improve the performance of web search. By incorporating social matching, the performance can be further improved. A similar conclusion can be drawn from Table 4, which shows the comparison of MAP on the two query sets.
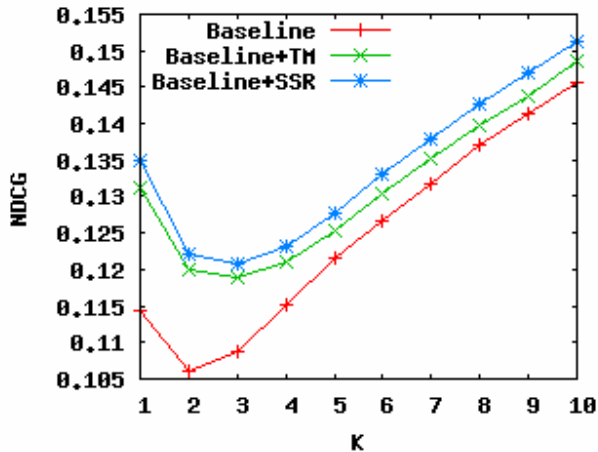
Figure 6. NDCG at K for Baseline, Baseline +TM, and Baseline +SSR for varying K

Table 4. Comparison of MAP between similarity features

Method	MQ50	AQ3000
Baseline	0.4115	0.1091
Baseline +TM	0.4341	0.1128
Baseline +SSR	**0.4697**	**0.1147**

4.4.5 Dynamic Ranking Using SPR

Figure 7 shows the comparison between NDCG of PageRank and SocialPageRank on the AQ set. Both SPR and PageRank benefit web search. The better result is achieved by SPR. Again, similar conclusion can be drawn from Table 5, which shows the comparison of MAP on the two query sets.
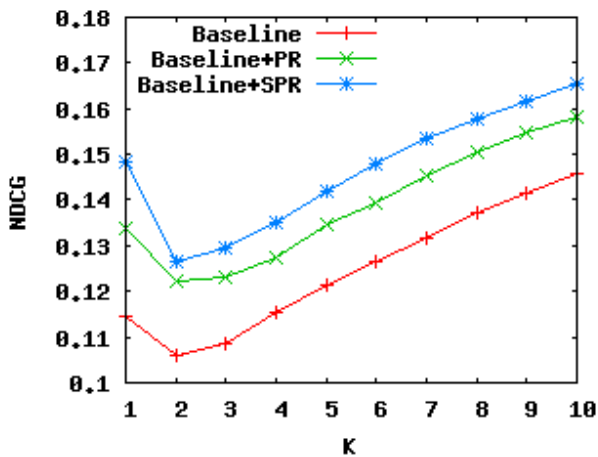

Figure 7. NDCG at *K* for BM25, BM25-PR, and BM25-SPR for varying *K*

Table 5. Comparison of MAP between static features

Method	MQ50	AQ3000
Baseline	0.4115	0.1091
Baseline +PR	0.4141	0.1166
Baseline +SPR	**0.4278**	**0.1225**

4.4.6 Dynamic Ranking Using Both SSR and SPR

By incorporating both SocialSimRank and SocialPageRank, we can achieve the best search result as shown in Table 6. T-tests on MAP show statistically significant improvements (p-value<0.05).

Table 6. Dynamic ranking with both SSR and SPR

Method	MQ50	AQ3000
Baseline	0.4115	0.1091
Baseline+SSR,SPR	**0.4724 (+14.80%)**	**0.1364 (+25.02%)**

4.4.7 Case Study

To understand how these improvements are achieved, we present a case study here. For simplicity, the similarity between the query and document content is not considered.

Given the query "*airfare*", 318 web pages are associated through the social annotations and the top-4 web pages according to SPR scores are shown in Table 7. Through the reviews of the travel sites like excellent-romantic-vacations [5], we can conclude that the social annotations are useful and SPR are really effective. For example, the site *http://www.kayak.com/* with top SPR score is also ranked first by excellent-romantic-vacations and called "Google of Travel Sites".

Table 7. Web pages associated by annotation "airfare"

URLs	SPR
http://www.kayak.com/	9
http://www.travelocity.com/	8
http://www.tripadvisor.com/	8
http://www.expedia.com/	8

Then by using SSR rank, we find the top-4 similar tags to "*airfare*" are "*ticket*", "*flight*", "*hotel*", and "*airline*". Through the analysis, we find most of the top ranked web sites in Table 7 are annotated by these similar annotations as well. Besides, these similar annotations also introduce some new web pages. Table 8 shows the top-SPR URLs that are not annotated by "*airfare*" in our corpus.

---

[5]http://www.excellent-romantic-vacations.com/best-airfare-search-engine.html

Table 8. Illustration of semantically related web pages based on SocialSimRank for query "*airfare*"

Annotation	Semantic Related Web Pages	SPR
ticket	http://www.sidestep.com/ http://www.pollstar.com/	8 8
flight	http://www.seatguru.com/ http://www.flightgear.org/	9 8
hotel	http://www.sleepinginairports.net/ http://www.world66.com/	8 8
airline	http://www.seatguru.com/ http://www.sleepinginairports.net/	9 8

From the above table, we find that most of the newly introduced web pages are relevant to "*airfare*". For example, *http://www.sleepinginairports.net/* is an interesting site that is annotated by both *ticket* and *airline*. The similar tags may also introduce some noise pages, e.g., *http://www.pollstar.com/* and *http://www.flightgear.org/* are related to concert ticket and flight simulator, respectively. However, the noise pages will not be ranked high in our setting as no other similar annotations will be assigned to it.

5. DISSCUSSION

As we have observed that the social annotations do benefit web search. There are still several problems to further address.

5.1 Annotation Coverage

First, the user submitted queries may not match any social annotation. In this case, SSR will not be applied and SPR will keep on providing the most popular web pages to the user.

Second, many web pages may have no annotations. These web pages will benefit from neither SSR nor SPR. The pages that are not annotated can be roughly divided into three categories: 1) newly emerging web pages: these pages are too fresh to be annotated or even learnt; 2) key-page-associated web pages: these pages are not annotated because they can be accessed easily via the key pages such as hub pages and homepages while users tend to annotate key pages only; 3) uninteresting web pages: these pages may interest no user. The emergence of new web pages usually does not influence the social search a lot since the social annotation systems are sensitive to new things. For example, [18] shows that popular annotations can be found over time. With the help of the sensitivity of these systems and the SSR algorithm, we can quickly discover new valuable web pages with a small amount of annotations. As for key-page-associated pages, one feasible solution is to propagate the annotations from the key pages to them. As for uninteresting pages, it is believed that the lack of annotations would not affect the social search on the whole.

5.2 Annotation Ambiguity

Annotation ambiguity is another problem concerned with SSR, i.e., SSR may find the similar terms to the query terms while fail to disambiguate terms that have more than one meanings. For example, as has been shown in the case studies, *ticket* may refer to either *airplane ticket* or *concert ticket*, and terms with these two different meanings will be mixed up. In [36], Wu *et al.* studied the problem of annotation ambiguity by using a mixture model [31]; however, it is not suitable for the web search due to its high computational complexity. Some efficient disambiguation methods may be required for further improving the performance of SSR. However, the ambiguity problem does not affect the search a lot since this problem can be lightened by query word collocation and word senses' skewed distribution [20].

5.3 Annotation Spamming

Initially, there are few ads or spams in social annotations. However, as social annotation becomes more and more popular, the amount of spam could drastically increase in the near future and spamming will become a real concern for social search [14].

Both SSR and SPR proposed in this paper take the assumption that the social annotations are good summaries of web pages, so malicious annotations have a good opportunity to harm the search quality. There are generally two ways in preventing the spam annotations. 1) Manually or semi-automatically deleting spam annotations and punishing users who abuse the social annotation system. Such work usually relies on service providers; 2) Filtering out spam annotations by using statistical and linguistic analysis before the use of SSR and SPR. This should be the main approach we will study.

6. CONCLUSION

In this paper, we studied the novel problem of integrating social annotations into web search. We observed that the fast emerging annotations provided not only a multi-faceted summary but also a good indicator of the quality of web pages. Specifically, social annotations could benefit web search in both similarity ranking and static ranking. Two novel iterative algorithms have been proposed to capture the social annotations' capability on similarity ranking and static ranking, respectively. The experimental results showed that SSR can successfully find the latent semantic relations among annotations and SPR can provide the static ranking from the web annotators' perspective. Experiments on two query sets showed that both SPR and SSR could benefit web search significantly. The main contributions can be concluded as follows:

1) The proposal to study the problem of using social annotations to improve the quality of web search.

2) The proposal of the SocialSimRank algorithm to measure the association among various annotations.

3) The proposal of the SocialPageRank algorithm to measure a web page's static ranking based on social annotations.

It is just a beginning to integrate social annotations into web search. In the future, we would optimize the proposed algorithms and explore more sophisticated social features to improve the social search quality.

8. REFERENCES

[1] A. Hotho, R. Jaschke, C. Schmitz, and G. Stumme. Information Retrieval in Folksonomies: Search and Ranking. In: *Proc. of ESWC 2006*, pp.411-426, 2006.

[2] A. Mathes. Folksonomies – Cooperative Classification and Communication through Shared Metadata. http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html, December 2004.

[3] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In: *Proc. of the ICML 2005*, Bonn, Germany, 2005.

[4] Delicious: http://del.icio.us

[5] E. Agichtein, E. Brill, and S. Dumais. Improving Web Search Ranking by Incorporating User Behavior Information. In *Proc. of SIGIR 2006*, August 6–11, 2006

[6] E. Quintarelli. Folksonomies: power to the people. *Paper presented at the ISKO Italy-UniMIB meeting.* http://www.iskoi.org/doc/folksonomies.htm, June 2005

[7] F. McSherry. A uniform approach to accelerated PageRank computation. In: *Proc. of WWW 2005*, pp.575-582, 2005.

[8] G. Golub, C.F. Van Loan, Matrix Computations, *Johns Hopkins University Press*, 1989.

[9] G. Jeh, J. Widom,. SimRank: A Measure of Structural-Context Similarity. In *Proc. of SIGKDD 2002*, pp.538-543, 2001.

[10] 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 *Proc. of CIKM 2005* pp.118-126 2005

[11] G. Salton and M. J. McGill. Introduction to Modern Information Retrieval. *McGraw-Hill*, New York, 1983.

[12] G. Smith. Atomiq: Folksonomy: social classification. http://atomiq.org/archives/2004/08/folksonomy_social_classification.html, Aug 3, 2004

[13] http://blog.del.icio.us/

[14] http://www.marketingpilgrim.com/2006/01/winks-michael-tanne-discusses-future.html

[15] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proc. of 9th Annual ACM-SIAM Symposium. Discrete Algorithms*, pp. 668-677, 1998

[16] K Jarvelin and J. Kekalainen. IR evaluation methods for retrieving highly relevant documents. In *Proc. of SIGIR 2000*, 2000

[17] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. *Technical report*, Stanford Digital Library Technologies Project, 1998.

[18] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan, A. Tomkins. Visualizing Tags over Time. In: *Proc. of WWW2006*, pp. 193-202, May 23.26, 2006

[19] M. Richardson, and A. Prakash, and E. Brill. Beyond PageRank: Machine Learning for Static Ranking. In: *Proc. of WWW2006*, May 23-26, 2006.

[20] M. Sanderson: Retrieval with good sense. *Information Retrieval* (2), pp.47-67, 2000.

[21] N. Craswell, D. Hawking, and S. Robertson Effective Site Finding using Link Anchor Information, In: *Proc. of SIGIR 2001*, pp. 250-257, New Orleans, 2001.

[22] O. Dekel, C. Manning, and Y. Singer. Log-linear models for label-ranking. In: *Advances in Neural Information Processing Systems* (16). Cambridge, MA: MIT Press, 2003.

[23] P. A. Dmitriev, N. Eiron, M. Fontoura, and E. Shekita. Using Annotations in Enterprise Search. In: *Proc. of WWW 2006* , pp. 811-817, May 23.26, 2006.

[24] P. Merholz. Metadata for the Masses. October 19, 2004. http://www.adaptivepath.com/publications/essays/archives/000361.php

[25] P. Mika Ontologies are us: a unified model of social networks and semantics. In: *Proc. of ISWC 2005*. pp. 522-536, Nov. 2005.

[26] R. Herbrich, T. Graepel, and K. Obermayer. Support vector learning for ordinal regression. In: *Proc. of the 9th International Conference on Artificial Neural Networks*, pp.97-102. 1999.

[27] S. A. Golder, and B. A. Huberman. Usage Patterns of Collaborative Tagging Systems. *Journal of Information Science*, 32(2), pp.198-208, 2006.

[28] S. Brin and L. Page, The Anatomy of a Large-Scale Hypertextual Web Search Engine, in: *Computer Networks and ISDN Systems*, 30(1-7) pp. 107-117, 1998.

[29] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull, M. Lau. Okapi at TREC. In:*Text REtrieval Conference*, pp.21–30, 1992.

[30] T. Hammond, T. Hannay, B. Lund, and J. Scott. Social book marking tools (i) - a general review. *D-Lib Magazine*, 11(4), 2005.

[31] T. Hofmann, and J. Puzicha. Statistical Models for Co-occurrence Data. *Technical report*, A.I.Memo 1635, MIT, 1998.

[32] T. Joachims. Optimizing search engines using clickthrough data. In: *Proc. of SIGKDD 2002*, pp. 133-142, 2002.

[33] T. V. Wal. Explaining and showing broad and narrow folksonomies. http://www.personalinfocloud.com/2005/02/explaining_and_.html : February 21, 2005

[34] T. Westerveld., W. Kraaij., and D. Hiemstra, Retrieving Web Pages using Content, Links, URLs and Anchors, in: *Proc. of TREC10*, 2002.

[35] WordNet: http://wordnet.princeton.edu/

[36] X. Wu, L. Zhang, and Y. Yu. Exploring Social Annotations for the Semantic Web. In: *Proc. of WWW 2006*, pp. 417-426, May 23.26, 2006

[37] Y. Hu, G. Xin, R. Song, G. Hu, S. Shi, Y. Cao, and H. Li. Title Extraction from Bodies of HTML Documents and Its Application to Web Page Retrieval. In: *Proc. of SIGIR 2005*. pp. 250-257, 2005