Collective Collaborative Tagging System

Jong Y. Choi, Joshua Rosen, Siddharth Maini, Marlon E. Pierce, and Geoffrey C. Fox
Community Grids Laboratory
Indiana University, Bloomington, IN 47404, USA
{jychoi, jjrosen, smaini, marpierc, gcf}@cs.indiana.edu

***Abstract*—Currently in the Internet many collaborative tagging sites exist, but there is the need for a service to integrate the data from the multiple sites to form a large and unified set of collaborative data from which users can have more accurate and richer information than from a single site. In our paper, we have proposed a collective collaborative tagging (CCT) service architecture in which both service providers and individual users can merge folksonomy data (in the form of keyword tags) stored in different sources to build a larger, unified repository. We have also examined a range of algorithms that can be applied to different problems in folksonomy analysis and information discovery. These algorithms address several common problems for online systems: searching, getting recommendations, finding communities of similar users, and finding interesting new information by trends. Our contributions are to a) systematically examine the available public algorithms' application to tag-based folksonomies, and b) to propose a service architecture that can provide these algorithms as online capabilities.**

I. Introduction

In recent years, the number of virtual on-line communities has grown rapidly, and the quantity of information and knowledge produced in those on-line communities is immense. The interesting aspect of this trend is that the knowledge in the Internet is not only produced by a small number of experts, but also they are produced by the normal Internet users. Ratings, recommendations, and tagging are typical examples. Such keyword tagging systems create what is commonly termed a "folksonomy", in which the classification or description of a particular Web object emerges from the community. This is in contrast to more structured taxonomies and ontologies in which knowledge representation is modeled by experts.

Many on-line systems have been developed to support such user activities. Among them, collaborative tagging systems, also known as social bookmarking systems, are one of the most popular systems designed to utilize the power of peoples knowledge and provide efficient ways of searching information. The core of collaborative tagging systems is to provide a simple and easy interface to collaboratively annotate Internet objects – mostly URLs but not restricted to documents, Internet media, and so on – by tags or keywords. In this way, the system can easily collect people's knowledge and help users to easily access such collections. Delicious, Connotea, and CiteULike, to name a few, are well known for collaborative tagging systems. These keyword tags are often displayed as"tag clouds" that use fonts to indicate the relative importance of various terms.

Although tagging systems can be used by individual users to manage URL collections, the idea of collaborative tagging systems can be easily applied to develop a community-oriented on-line system. For an example, our Minority Serving Institution-Cyberinfrastructure Empowerment Coalition (hereafter MSI-CIEC for short) portal has been developed to support researchers at Minority Serving Institution to connect with each other and with the education, outreach, and training services that are designed to serve them, expanding their participation in cyberinfrastructure research efforts [1]. In the MSI-CIEC portal, users can create public profiles to describe their research interests and annotate them with tags, create bookmarks of URLs with tags, and search information by using own tags or tags created by others. The home page of MSI-CIEC is shown in Fig 1.

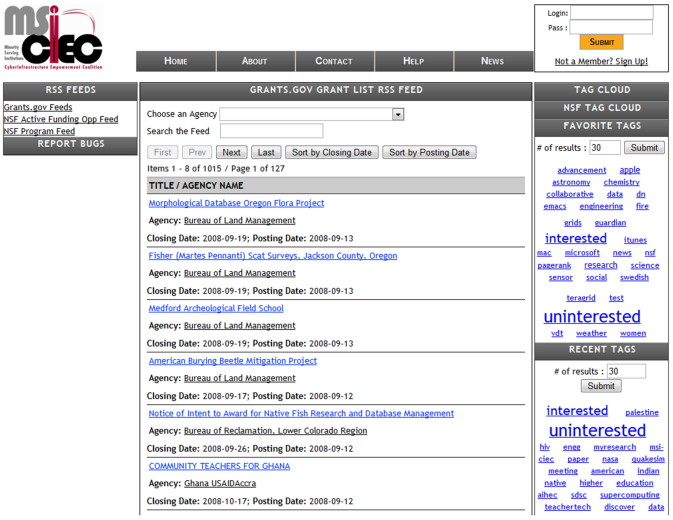


Fig. 1. The MSI-CIEC Social Networking Web Portal uses tagging annotations to manage URLs to online program announcements and funding solicitations.

Although the collaborative tagging systems are common in many on-line communities, we have observed the following deficiencies:

- The collaborative tagging systems are to collect knowledge from the people and the quality of knowledge users can get will increase as the quantity of data people provided grows. Currently in the Internet many collaborative tagging sites exist, but there is the need for a service to integrate the data from the multiple sites to form a large and unified set of collaborative data from which users can have more accurate and richer information than from a single site.

- Many Information Retrieval (IR) algorithms have been well studied and open to public. Although most of the collaborative tagging sites provide various searching services, their algorithms are closed to public and unknown to the users. Furthermore, most of them provide only one type of searching algorithm and the users have no opportunity to apply various other IR algorithms to find the best information available from the data. Using the same data set with various different searching algorithms, users can have more chances to discover hidden information varied in the data set.

Motivated by the above considerations, the purpose of this paper is two fold: i) to propose a new collaborative tagging system service architecture that can collect tag data from other repositories and merge them in order to provide better quality of knowledge, and ii) to compare commonly used algorithms for the folksonomy analysis. We envision a system that allow users to try different algorithms. The details of the algorithms are irrelevant to the user, but the quality of results (searching, recommendations, etc) can be readily judged through the Web interface.

Although the IR services are general purpose, we are particularly interested in their integration with science gateways and portals. Gateways have tended to focus on the ability to interact with remote resources for scientific data, information, and application management, but we believe these systems will naturally gravitate toward more social, Web 2.0 like online communities. Important taggable Web objects (URLs) for gateways include online data sets, experiments, workflows, journal articles, presentations, and funding announcements. Tag-supported online communities can greatly expand the capabilities of traditional science portals.

II. Related Work

Many researches on collaborative tagging systems have been conducted to develop efficient searching or recommendation schemes by using folksonomy data. Related researches can be found in FolkRank [2], Flickr tag recommendations [3], and probabilistic models for information retrieval [4].

However, only a few researches have been performed to study about the impact of merging folksonomy data. Among them GiveALink [5] is worth mentioning. The GiveALink system is a social bookmarking system to enable users to share their bookmarks with others and provides rich and personalized searching and recommendation services based on the analysis of collected bookmarks users uploaded. Although the core idea to utilize peoples knowledge is similar with our motivation, our proposed system will focus on socialized and collaborative tagging activities.

Metasearch engines have been developed to improve quality of search results by using multiple search engines [6]. For a given query, the metasearch engines will query to multiple search engines and aggregate the results from the different sources. Aggregating external information from various sources shares the similarity with our approaches.

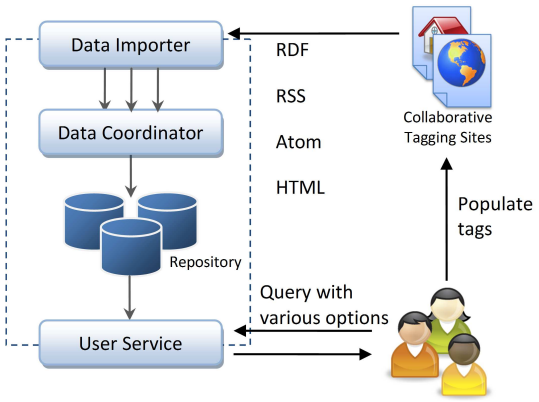


Fig. 2. Overview of Collective Collaborative Tagging (CCT) System

III. A New System

In this section, we propose a *collective collaborative tagging* (CCT for short) Web service that can provide various collaborative tagging services in a uniform way. Our service architecture may be used both service providers (i.e. the MSI-CIEC portal) and users. Our CCT system is designed to provide the following key functions.

- Importing data from multiple sources to build a large and unified tag repository
- Query services with options to run various IR algorithms
- Query services with options to run with different data sources and various parameter settings

A. Proposed Web Architecture

The system will consist of the three main components; *data importer*, *data coordinator*, and *user service* (Fig 2).

Details of main three components are as follow.

- **Data Importer**: Importing tagging data with machine readable format such as RDF, RSS, Atom or Web APIs from number of different collaborative tagging sites. Importing can be done asynchronously or synchronously.
- **Data coordinator**: Merging data from different sources and storing them into an uniform repository. The coordinator will resolve possible format conflicts and duplication problem which may exist in multiple sites.
- **User service**: Providing various machine learning based searching algorithms and options users can choose as a form of Web service APIs. The queries will be performed against unified repository which stores tagging data collected from different sources collaborative tagging systems

B. Service Type

Various kinds of user requests to extract information from the folksonomy data can exist in collaborative tagging systems; for example, searching items by using tags, getting personalized recommendations based on user's profiles or past activities, discovering group of users or communities sharing similar interests, just to name a few. Those demands can be generally categorized into 4 classes and our CCT system

TABLE I
GENERAL TYPES OF SERVICE IN COLLABORATIVE TAGGING SYSTEMS. SEE TEXT FOR DEFINITION OF TERMS

Type	Services	Candidate Algorithms
I	Searching	LSI, FolkRank, Tag Graph
II	Recommendation	LSI, FolkRank, Tag Graph
III	Clustering	K-Means, Deterministic Annealing Clustering, Pairwise Deterministic Annealing
IV	Trend detection	Time Series Analysis (HMM and other techniques)

will provide services to support those requests. The following classification is not exclusive but rather overlapping in some sense.

Type I – Searching by tags : For a given set of tags as an input, searching the most relevant objects with the input tags is an essential function in the collaborative tagging system. Generally the objects can be either documents, items, users, or anything annotated by tags in the system. Results will be returned to users in an ordered fashion based on some computed scores.

Type II – Recommendation : With no explicit input of tags, the system will return a recommendation list of objects. While the input tags used in searching by tags should be explicitly defined by a user, in recommendation those are generated implicitly by the system, based on user's previous activities, preferences, or profiles. For an example, the system can give to a user a recommendation list of documents which haven't been discovered by the user, based on the user's past tagging activities. Also, recommendation of tags is possible when a user wants to annotate a document for the first time, the system can recommend other co-used tags with his initial input.

Type III – Clustering : This is so called community discovery. Not only searching for the most relevant objects, it is also useful finding a group or a community which shares more common interests expressed by tags within the group members than with others.

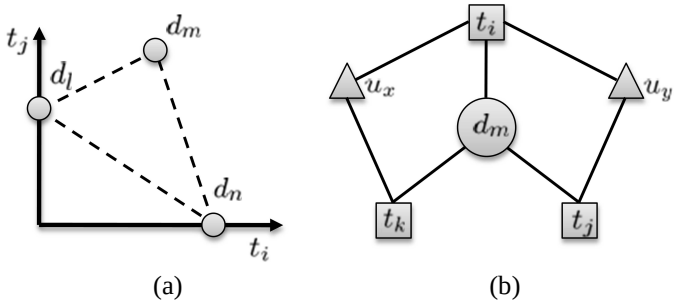
Type IV – Trend Detection : The system analyzes the tagging activities in time-series manner and detect interesting patterns of tagging or abnormality among the tag data set.

More specific examples of service types or information users can get for each category are summarized in Table I.

Conventional Web service system design and tools are well known, and we will make use of these. Our contribution is to implement a suite of these services that encapsulate various machine learning algorithms for folksonomy analysis. We discuss these algorithms and their applications to tagging systems in the following sections.

IV. MODELS FOR TAG ANALYSIS

A collaborative tagging system is designed to utilize the power of peoples knowledge and provide an efficient way of searching information from the collaboratively annotated data set. In this way, the system can help users to find the information with more efficiency and discover unexposed or hidden information buried under piles of information. Thus, developing efficient models and algorithms for searching is the key step for building a successful collaborative tagging system. In this section, we discuss the models for developing folksonomy searching engines and various algorithms for searching and tag analysis.



(a) (b)

Fig. 3. Examples of folksonomy models. (a) the vector space model and (b) the graph-based model.

A. Models

For building an efficient searching engine for folksonomies, the way to represent folksonomy data is an important issue. In the field of Information Retrieval (hereafter IR for short), two models – *the vector space model* and *the graph model* – have been widely used and they are both well applicable in folksonomy indexing.

Although both models are sharing many similar aspects, they are distinct in many practical points of views. As examples, the Latent Semantic Indexing (hereafter LSI for short) (we will discuss details of this algorithm later) is using the vector space model for indexing and measuring pairwise similarities between objects, and the famous ranking algorithm PageRank used by Google and its variant FolkRank for folksonomy searching are based on the graph model. While the vector space model has been widely used in many areas due to its simplicity, not many researches have been conducted for the use of the graph model so far.

1) Vector space model: In the vector space model, also known as bag-of-words model, URLs are represented as an unordered collection of tags and by using mathematical notation a vector is used. I.e., a URL d_j can be represented as a q-dimensional column vector where q equals the total number of distinct tags in the system and its i-th element is a weight of the occurrence of the tag t_i (We will discuss various weight schemes shortly). Thus, the whole collection of n URLs can be represented as a tag-URL matrix $T \in \mathbb{R}^{q \times n}$ where each column corresponds to a URL $d_j (1 \leq j \leq n)$. In the vector space model, it is often convenient to consider a URL as a point in a q-dimension coordinate system. An example is shown in Fig 3 (a).

2) Graph-based model: Although the vector space model is simple and easy-to-use, sometimes it lacks the ability to describe URL-URL relationships, which is more easier in the graph model. In the graph model, folksonomies can be represented as a network of connections, also known as *tag*

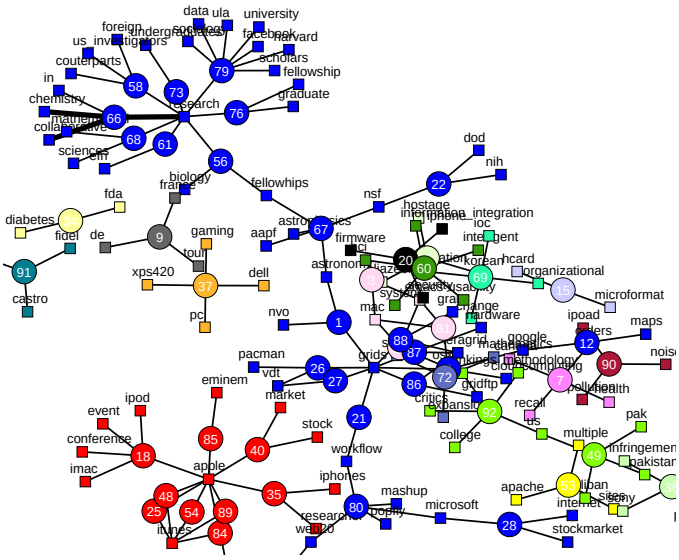


Fig. 4. An example of tag graph. The data used in this figure obtained from our in-house collaborative tagging system, MSI-CIEC portal. In the tag graph, URLs and tags are represented as circles and squares respectively and edges are drawn to show tagging relationship between them.

graph, which consists of URLs as nodes and connections between them as edges. An example is shown in Fig 3 (b) and Fig 4.

More specifically, a tag graph is a undirected tripartite graph $G = (V, E)$ where nodes in V are one of URLs in disjoint subsets of three entities – URLs, tags, and users – and edges exist only between three entities. Each edge will be added for each single transaction, i.e., annotating an URL with a set of tags by a user.

B. Similarity Measurement

Measuring similarity between two objects is a key step in folksonomy analysis and it is directly related to the performance of the system. Although it is possible in folksonomy analysis to measure various types of similarities such as URL-URL, URL-tag, URL-user, user-tag, and user-user, in this paper we only consider URL-URL similarity for brevity. The other measurements can be easily estimated by using the same manner.

1) Weight Measurement: Weight measurement is a scheme to quantify the importance of tag t_i used for annotating URL d_j. A simple minded approach is to count the occurrence of the tag t_i for the URL d_j, which is known as Term Frequency (TF for short). As observed in many IR researches, however, this approach has an disadvantage to utilize the low frequency terms or tags. Tag distributions in folksonomies usually follows the Zipf's power law where a few majority tags govern the most of distributions [7] and thus minor tags can lost their importance in many searching algorithms. Thus, some normalization scheme can be used to avoid this problem and to collect more variety information by exploiting minor tags in folksonomies.

Various schemes have been suggested in many IR literature but the most popular scheme is Term Frequency-Inverse Document Frequency (TF-IDF for short) which is the multiplication of TF and IDF. In a nutshell, term frequency tf_{ij} is the number of occurrence of tag t_i for document d_j and the document frequency df_i is the number of documents having the same tag t_i. IDF is computed by $\log \frac{n}{df_i}$ for the total number of document n and thus TF-IDF equals $tf_{ij} \times \log \frac{n}{df_i}$. Formulas are summarized in Table II.

2) Similarity Measurement: Similarity measurement is to measure a degree of likeness between two tagged objects in folksonomies. In the vector space model, various similarity measurement schemes have been developed in the field of IR and in practice three similarity measurement schemes are the most popular among them: Cosine, Jaccard, and Pearson [8] (summarized in Table II).

While in the vector space model such similarities are measured by geometric characteristics (such as cosine angles) or statistical ways (such as Jaccard and Pearson), similarities in the graph model can be measured by graph theoretic properties, such as hop distances, shortest paths, maximum flows, and so on.

Pairwise similarity is also an important measurement for using in finding groups or communities. Note, however, measuring pairwise similarity is also different in both models. In vector space mode, all URL-URL similarities can be directly computed from the tag-URL matrix A; I.e, in the vector space model, we can compute a pairwise similarity matrix $D = [\delta_{ij}] \in \mathbb{R}^{n \times n}$ and its entries δ_{ij} by computing the similarity between any two URLs d_j and d_k among total n URLs. Thus, the computation cost to build $n \times n$ pairwise similarity matrix D is $\mathcal{O}(n^2)$

However, in the graph model we cannot compute pairwise similarities directly from the matrix A but, instead, we can have this iteratively; Firstly, compute only similarities of directly connected URLs, i.e., URLs sharing at least one common tag between them, and then, measure similarities of the others, which have no direct connections, by means of discovering paths between them. Path discoveries can be done by using the algorithms for finding the shortest path. Floyd-Warshall algorithm [9] is well known for this problem and this requires generally $\mathcal{O}(n^3)$ computations. This computation burdens can be removed by using parallel computing techniques in clusters or multi-core systems.

V. Algorithms

Currently numerous algorithm have been studied for supporting various types of services in collaborative tagging systems and this is also very active research area. In this section, we focus on core algorithms which can successfully support our service classification as shown in Table I.

A. Latent Semantic Indexing

The Latent Semantic Indexing (hereafter LSI for short) has been widely used for indexing the Web pages or documents in

TABLE II
EQUATIONS USED FOR MEASURE WEIGHTS AND DISSIMILARITIES. SLIGHTLY MODIFIED FROM ORIGINAL EQUATIONS.

Abbr	Name	Definition
TF tf_{ij}	Term Frequency	The number of tagged term t_j for document d_i
DF df_j	Document Frequency	The number of documents having the same tag t_j
TF-IDF $tfidf_{ij}$	TF-Inverse DF	$tf_{ij} \times \log \frac{n}{df_j}$ where n is the total number of d_i
COS(d_i, d_j)	Cosine	$\sum_k w_{ik}w_{jk} / \sqrt{\sum_k w_{ik}^2 \sum_k w_{jk}^2}$
JAC(d_i, d_j)	Jaccard	$\sum_k w_{ik}w_{jk} / \left(\sum_k w_{ik}^2 + \sum_k w_{jk}^2 - \sum_k w_{ik}w_{jk}\right)$
PEA(d_i, d_j)	Pearson	$\frac{\left(\sum_k w_{ik}w_{jk} - \frac{1}{q}\sum_k w_{ik}\sum_k w_{jk}\right)}{\sqrt{\left(\sum_k w_{ik}^2 - \frac{1}{q}(\sum_k w_{ik})^2\right)\left(\sum_k w_{jk}^2 - \frac{1}{q}(\sum_k w_{jk})^2\right)}}$

libraries and served as one of the most popular searching algorithms based on the vector space model. The LSI algorithm can be also used in folksonomies as a searching engine to support the Type-I service in the vector space model. Using the tag-URL matrix representing data in the system as an input, the LSI algorithm can help to recover underlying or latent structures of folksonomies, often obscured by noisy data, and enable to find the true relationship between tags and URLs without noises based on the statistical information.

The core idea of LSI algorithm is that since the dimension of the raw or untreated tag-URL matrix is usually too high to find the concise relationships between tags and objects, the dimension should be reduced to recover latent structures of the input matrix. Thus, the algorithm projects the tag-URL matrix $A = [a_{ij}] \in \mathbb{R}^{q \times n}$ in the q-dimension space onto a lower dimension space d such that $d \ll q$ in order to remove "noisy" information and recover the true relationships. In this sense, the LSI algorithm can be considered as a dimension reduction algorithm changing dimension from q to d.

For dimension reduction processing, the LSI uses the Singular Value Decomposition (SVD) method to find the best lower dimension matrix $\hat{A}$ of the raw matrix A as an input in a way to make the 2-norm difference $||A - \hat{A}||_2$ minimized.

B. FolkRank

Inspired from the PageRank algorithm which exploits the network structures of Web pages, the FolkRank algorithm has been developed as a folksonomy search engine by using the graph model [2]. The FolkRank algorithm can be used to provide Type-I service by using the graph model.

The PageRank algorithm starts with building the network of the hyperlinked Web pages as a directed graph, in which a Web page can have inlinks (or incoming edges) or outlinks (or outgoing edges) or both. Given the graph of Web pages, the next step is to spread out the weight of importance, which is known as rank, of each Web page from the inlinks to the outlinks until the weights are converged. The intuition is that a Web page is getting more important and having higher rank, if it has more inlinks from the higher ranked pages.

The FolkRank algorithm adopted the same weight spreading approaches as in the PageRank. The main difference, however, lies in the graph. In the FolkRank, the graph of tags has no direction, while the PageRank uses directed graphs. More details of FolkRank algorithms can be found in [2]

C. Clustering

Clustering algorithms can be used to discover hidden group structures in folksonomy data. Among numerous clustering algorithms, k-means and deterministic annealing algorithms [10] can be used for the folksonomy data in the vector space model and pairwise deterministic annealing [11] can be used for the graph-based model.

VI. EXPERIMENTS

For the experiments in this paper, we collected tagging data from the Connotea [1]. The Connotea data set was collected in January 2008 and extracted 1131 URLs and 6071 tags from its popular URL list.

In this experiment, we have applied LSI schemes and generated the tag graph of the Connotea data set to show how those algorithms can be used in the collaborative tagging system and our CCT system.

A. Latent Semantic Indexing

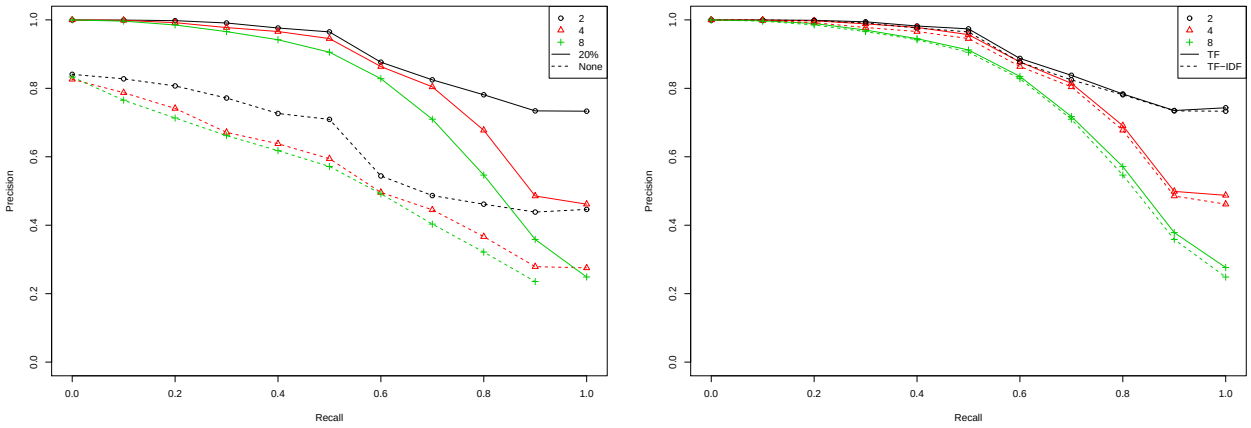
To study how the LSI scheme can be used with folksonomy data, we have applied the LSI scheme to the Connotea data set with randomly generated queries having various lengths. In our experiment, we generated total 6000 queries which evenly consisting of queries of 2, 4, and 8 term length and measured precisions and recalls with two different dimension reduction rate of the LSI scheme: 20% variance-based dimension reduction versus 0% (i.e., no dimension reduction).

To evaluate the system, we have generated precision-recall graphs which are traditionally used in the field of IR [12]. Precision is the fraction of retrieved relevant documents in a query result and recall is the fraction of retrieved relevant documents in the system. I.e, for the returned URLs by the LSI scheme, precision and recall can be defined by

$$\begin{aligned} \text{Precision} &= D_r/D_q, \\ \text{Recall} &= D_r/N_r \end{aligned}$$

where D_r is the number of relevant URLs included in the answer set, D_q is the number of returned URL, and N_r is the total number of relevant URLs in the system. In general,

[1] http://www.connotea.org/



(a) (b)

Fig. 5. The precision-recall graph by using the LSI scheme. In (a), different dimension reduction rates are used. 20% variance-based reduction (solid line) and 0% reduction (dashed line) are compared. 20% reduction outperforms 0% reduction. In (b), TF(solid line) and TF-IDF (dashed line) compared. TF performs better slightly than TF-IDF but no significant difference.

higher precisions at the lower recall levels indicate the better performance.

In precision and recall measurement, it is crucial to know relevant URLs for each query to simulate. For simplicity, we defined all URLs tagged by query tags as the relevant URLs.

As shown in the result (Fig 5 (a)), 20% dimension reduction outperforms 0% dimension reduction.

As for the second experiment, we have investigated how weight schemes effect searching performance. For this end, we have implemented the LSI schemes by using two different weight schemes – TF and TF-IDF. As shown in the result (Fig 5 (b)), TF(solid line) performs slightly better than TF-IDF (dashed line) but with no significant difference.

B. Graph-based analysis

We have investigated how the graph-based model can be used for folksonomy analysis by characterizing the network structures of tag graphs. For this end, we have constructed a tag graph by using Connotea data set and created a pairwise URL-URL similarity matrix.

In Fig 6, the URL-URL graph is visualized by using the Classical Multi-Dimensional Scaling(CMDS) algorithms.

Applying more sophisticated graph-based analysis technique to improve searching performance will be our next work.

VII. Conclusion

We have proposed a collective collaborative tagging (CCT) service architecture in which both service providers and individual users can merge folksonomy data (in the form of keyword tags) stored in different sources to build a larger, unified repository. We have examined a range of algorithms that can be applied to different problems in folksonomy analysis and information discovery. These algorithms address several common problems for online systems: searching, getting recommendations, finding communities of similar users, and finding interesting new information by trends. These capabilities are available on many Web community sites in some form or another, but the algorithm details are not public. This introduces a range of possibilities, from ad hoc techniques to very sophisticated proprietary algorithms (such as Netflex's recommendation system). Our contributions are to a) systematically examine the available public algorithms' application to tag-based folksonomies, and b) to propose a service architecture that can provide these algorithms as online capabilities.

References

[1] M. Pierce, G. Fox, J. Rosen, S. Maini, and J. Choi, "Social networking for scientists using tagging and shared bookmarks: a Web 2.0 application," *Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on*, pp. 257–266, 2008.

[2] A. Hotho, R. Jaschke, C. Schmitz, and G. Stumme, "Information Retrieval in Folksonomies: Search and Ranking," *LECTURE NOTES IN COMPUTER SCIENCE*, vol. 4011, p. 411, 2006.

[3] B. Sigurbjörnsson and R. van Zwol, "Flickr tag recommendation based on collective knowledge," 2008.

[4] D. Zhou, J. Bian, S. Zheng, H. Zha, and C. Giles, "Exploring social annotations for information retrieval," 2008.

[5] L. Stoilova, T. Holloway, B. Markines, A. Maguitman, and F. Menczer, "GiveALink: mining a semantic network of bookmarks for web search and recommendation," *Proceedings of the 3rd international workshop on Link discovery*, pp. 66–73, 2005.

[6] J. Aslam and M. Montague, "Models for metasearch," *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 276–284, 2001.

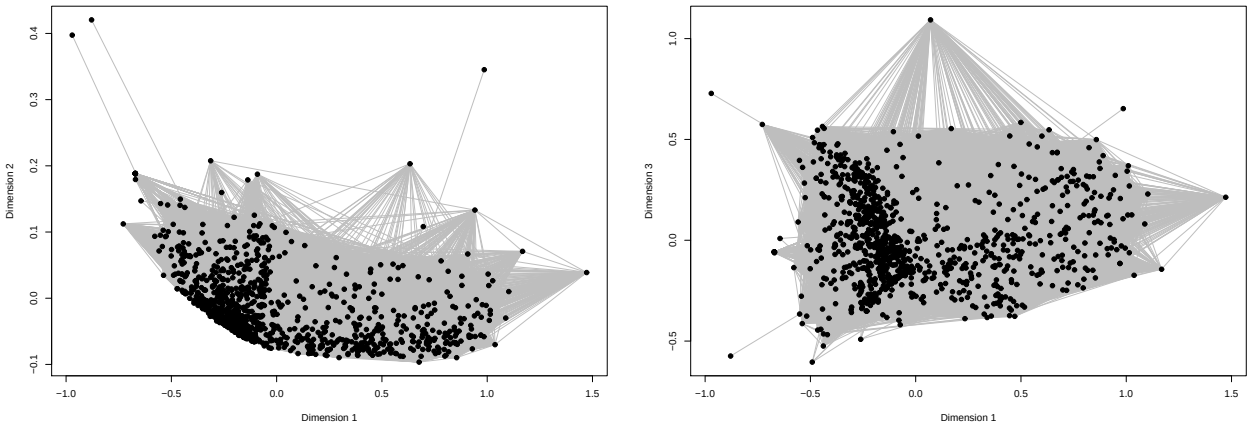
[7] H. Halpin, V. Robu, and H. Shepherd, "The complex dynamics of collaborative tagging," *Proceedings of the 16th international conference on World Wide Web*, pp. 211–220, 2007.

[8] K. Boyack, R. Klavans, and K. Börner, "Mapping the backbone of science," *Scientometrics*, vol. 64, no. 3, pp. 351–374, 2005.

[9] R. Floyd, "Algorithm 97: Shortest path," *Communications of the ACM*, vol. 5, no. 6, p. 345, 1962.

[10] K. Rose, "Deterministic annealing for clustering, compression, classification, regression, and related optimization problems," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2210–2239, 1998.

[11] J. Thomas, "Pairwise Data Clustering by Deterministic Annealing," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, pp. 1–14, 1997.



(a) (b)

Fig. 6. 2-D visualization of Connotea tag graph. The graph is generated from pairwise dissimilarity measurement by using CMDS. (a) 1st-2nd and (b) 1st-3rd principal dimensions are shown.

[12] R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval*. Addison-Wesley Harlow, England, 1999.