Exploiting Social Annotation for Automatic Resource Discovery

Anon Plangprasopchok and Kristina Lerman
USC Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292, USA
{plangpra,lerman}@isi.edu

Abstract

Information integration applications, such as mediators or mashups, that require access to information resources currently rely on users manually discovering and integrating them in the application. Manual resource discovery is a slow process, requiring the user to sift through results obtained via keyword-based search. Although search methods have advanced to include evidence from document contents, its metadata and the contents and link structure of the referring pages, they still do not adequately cover information sources — often called "the hidden Web"— that dynamically generate documents in response to a query. The recently popular social bookmarking sites, which allow users to annotate and share metadata about various information sources, provide rich evidence for resource discovery. In this paper, we describe a probabilistic model of the user annotation process in a social bookmarking system *del.icio.us*. We then use the model to automatically find resources relevant to a particular information domain. Our experimental results on data obtained from *del.icio.us* show this approach as a promising method for helping automate the resource discovery task.

Introduction

As the Web matures, an increasing number of dynamic information sources and services come online. Unlike static Web pages, these resources generate their contents dynamically in response to a query. They can be HTML-based, searching the site via an HTML form, or be a Web service. Proliferation of such resources has led to a number of novel applications, including Web-based mashups, such as Google maps and Yahoo pipes, information integration applications (Thakkar, Ambite, & Knoblock 2005) and intelligent office assistants (Lerman, Plangprasopchok, & Knoblock 2007) that compose information resources within the tasks they perform. In all these applications, however, the user must discover and model the relevant resources. Manual resource discovery is a very time consuming and laborious process. The user usually queries a Web search engine with appropriate keywords and additional parameters (e.g., asking for .kml or .wsdl files), and then must examine every resource returned by the search engine to evaluate whether it has the desired functionality. Often, it is desirable to have not one but several resources with an equivalent functionality to ensure robustness of information integration applications in the face of resource failure. Identifying several equivalent resources makes manual resource discovery even more time consuming.

The majority of the research in this area of information integration has focused on automating modeling resources — i.e., understanding semantics of data they use (Heß & Kushmerick 2003; Lerman, Plangprasopchok, & Knoblock 2006) and the functionality they provide (Carman & Knoblock 2007). In comparison, the resource discovery problem has received much less attention. Note that traditional search engines, which index resources by their contents — the words or terms they contain — are not likely to be useful in this domain, since the contents is dynamically generated. At best, they rely on the metadata supplied by the resource authors or the anchor text in the pages that link to the resource. Woogle (Dong *et al.* 2004) is one of the few search engines to index Web services based on the syntactic metadata provided in the WSDL files. It allows a user to search for services with a similar functionality or that accept the same inputs as another services.

Recently, a new generation of Web sites has rapidly gained popularity. Dubbed "social media," these sites allow users to share documents, including bookmarks, photos, or videos, and to *tag* the content with free-form keywords. While the initial purpose of tagging was to help users organize and manage their own documents, it has since been proposed that collective tagging of common documents can be used to organize information via an informal classification system dubbed a "folksonomy" (Mathes 2004). Consider, for example, *http://geocoder.us*, a geocoding service that takes an input as address and returns its latitude and longitude. On the social bookmarking site *del.icio.us*[1], this resource has been tagged by more than $1,000$ people. The most common tags associated by users with this resource are "map," "geocoding," "gps," "address," "latitude," and "longitude." This example suggests that although there is generally no controlled vocabulary in a social annotation system, tags can be used to categorize resources by their functionality.

We claim that social tagging can be used for information resource discovery. We explore three probabilistic gener-

[1] http://del.icio.us

ative models that can be used to describe the tagging process on *del.icio.us*. The first model is the probabilistic Latent Semantic model (Hofmann 1999) which ignores individual user by integrating bookmarking behaviors from all users. The second model, the three-way aspect model, was proposed (Wu, Zhang, & Yu 2006) to model *del.icio.us* users' annotations. The model assumes that there exists a global conceptual space that generates the observed values for users, resources and tags independently. We propose an alternative third model, motivated by the Author-Topic model (Rosen-Zvi *et al.* 2004), which maintains that latent topics that are of interest to the author generate the words in the documents. Since a single resource on *del.icio.us* could be tagged differently by different users, we separate "topics", as defined in Author-Topic model, into "(user) interests" and "(resource) topics". Together user interests and resource topics generate tags for one resource. In order to use the models for resource discovery, we describe each resource by a topic distribution and then compare this distribution with that of all other resources in order to identify relevant resources.

The paper is organized as follows. In the next section, we describe how tagging data is used in resource discovery. Subsequently we present the probabilistic model we have developed to aid in the resource discovery task. The section also describes two earlier related models. We then compares the performance of the three models on the datasets obtained from *del.icio.us*. We review prior work and finally present conclusions and future research directions.

Problem Definition

Suppose a user needs to find resources that provide some functionality: e.g., a service that returns current weather conditions, or latitude and longitude of a given address. In order to improve robustness and data coverage of an application, we often want more than one resource with the necessary functionality. In this paper, for simplicity, we assume that the user provides an example resource, that we call a *seed*, and wants to find more resources with the same functionality. By "same" we mean a resource that will accept the same input data types as the seed, and will return the same data types as the seed after applying the same operation to them. Note that we could have a more stringent requirement that the resource return the same data as the seed for the same input, but we don't want to exclude resources that may have different coverage.

We claim that users in a social bookmarking system such as *del.icio.us* annotate resources according to their functionality or topic (category). Although *del.icio.us* and similar systems provide different means for users to annotate document, such as notes and tags, in this paper we focus on utilizing the tags only. Thus, the variables in our model are resources R, users U and tags T. A bookmark i of resource r by user u can be formalized as a tuple $\langle r, u, \{t_1, t_2, \ldots\}\rangle_i$, which can be further broken down into a co-occurrence of a triple of a resource, a user and a tag: $\langle r, u, t\rangle$.

We collect these triples by crawling *del.icio.us*. The system provides three types of pages: a tag page — listing all resources that are tagged with a particular keyword; a user page — listing all resources that have been bookmarked by a particular user; and a resource page — listing all the tags the users have associated with that resource. *del.icio.us* also provides a method for navigating back and forth between these pages, allowing us to crawl the site. Given the seed, we get what *del.icio.us* shows as the most popular tags assigned by the users to it. Next we collect other resources annotated with these tags. For each of these we collect the resource-user-tag triples. We use these data to discover resources with the same functionality as the seed, as described below.

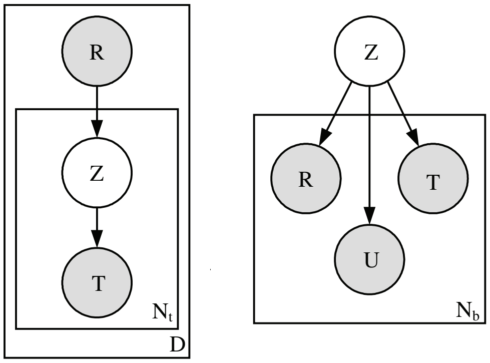

Figure 1: Graphical representations of the probabilistic Latent Semantic Model (left) and Multi-way Aspect Model (right) R, U, T and Z denote variables "Resource", "User", "Tag" and "Topic" repectively. N_t represents a number of tag occurrences for a particular resource; D represents a number of resources. Meanwhile, N_b represents a number of all resource-user-tag co-occurrences in the social annotation system. Note that filled circles represent observed variables.

Approach

We use probabilistic models in order to find a compressed description of the collected resources in terms of topic descriptions. This description is a vector of probabilities of how a particular resource is likely to be described by different topics. The topic distribution of the resource is subsequently used to compute similarity between resources using Jensen-Shannon divergence (Lin 1991). For the rest of this section, we describe the probabilistic models. We first briefly describe two existing models: the probabilistic Latent Semantic Analysis (pLSA) model and the Three-Way Aspect model (MWA). We then introduce a new model that explicitly takes into account users' interests and resources' topics. We compare performance of these models on the three *del.icio.us* datasets.

Probabilistic Latent Semantic Model (pLSA)

Hoffman (Hofmann 1999) proposed a probabilistic latent semantic model for associating word-document co-occurrences. The model hypothesized that a particular document is composed of a set of conceptual themes or topics Z. Words in a document were generated by these topics with some probability. We adapted the model to the context of social annotation by claiming that all users have common agreement on annotating a particular resource. All bookmarks from all users associated with a given resource were aggregated into a single corpus. Figure 1 shows the graphical representation of this model. Co-occurrences of a particular resource-tag pair were computed by summing resource-user-tag triples $\langle r, u, t \rangle$ over all users. The joint distribution over resource and tag is

$$p(r,t) = \sum_z p(t|z)p(z|r)p(r) \tag{1}$$

In order to estimate parameters $p(t|z)$, $p(z|r)$, $p(r)$ we define log likelihood L, which measures how the estimated parameters fit the observed data

$$L = \sum_{r,t} n(r,t) log(p(r,t)) \tag{2}$$

where $n(r,t)$ is a number of resource-tag co-occurrences. The EM algorithm (Dempster, Laird, & Rubin 1977) was applied to estimate those parameters that maximize L.

Three-way Aspect Model (MWA)

The three-way aspect model (or multi-way aspect model, MWA) was originally applied to document recommendation systems (Popescul *et al.* 2001), involving 3 entities: user, document and word. The model takes into account both user interest (pure collaborative filtering) and document content (content-based). Recently, the three-way aspect model was applied on social annotation data in order to demonstrate emergent semantics in a social annotation system and to use these semantics for information retrieval (Wu, Zhang, & Yu 2006). In this model, conceptual space was introduced as a latent variable, Z, which independently generated occurrences of resources, users and tags for a particular triple $\langle r, u, t \rangle$ (see Figure 1). The joint distribution over resource, user, and tag was defined as follows

$$p(r,u,t) = \sum_z p(r|z)p(u|z)p(t|z)p(z) \tag{3}$$

Similar to pLSA, the parameters $p(r|z)$, $p(u|z)$, $p(t|z)$ and $p(z)$ were estimated by maximizing the log likelihood objective function, $L = \sum_{r,u,t} n(r,u,t) log(p(r,u,t))$. EM algorithm was again applied to estimate those parameters.

Interest-Topic Model (ITM)

The motivation to implement the model proposed in this paper comes from the observation that users in a social annotation system have very broad interests. A set of tags in a particular bookmark could reflect both users' interests and resources' topics. As in the three-way aspect model, using a single latent variable to represent both "interests" and "topics" may not be appropriate, as intermixing between these two may skew the final similarity scores computed from the topic distribution over resources.

Instead, we propose to explicitly separate the latent variables into two: one representing user interests, I; another representing resource topics, Z. According to the proposed model, the process of resource-user-tag co-occurrence could be described as a stochastic process:

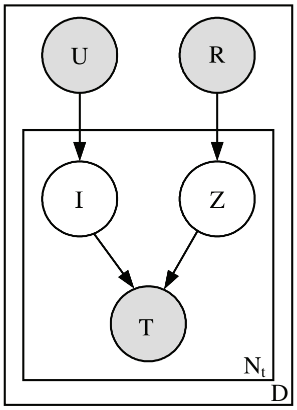

Figure 2: Graphical representation on the proposed model. R, U, T, I and Z denote variables "Resource", "User", "Tag", "Interest" and "Topic" repectively. N_t represents a number of tag occurrences for a one bookmark (by a particular user to a particular resource); D represents a number of all bookmarks in social annotation system.

- User u finds a resource r interesting and she would like to bookmark it
- User u has her own interest profile i; meanwhile the resource has a set of topics z.
- Tag t is then chosen based on users's interest and resource's topic

The process is depicted in a graphical form in Figure 2. From the process described above, the joint probability of resource, user and tag is written as

$$P(r,u,t) = \sum_{i,z} p(t|i,z)p(i|u)p(z|r)p(u)p(r) \tag{4}$$

Log likelihood L is used as the objective function to estimate all parameters. Note that $p(u)$ and $p(r)$ could be obtained directly from observed data – the estimation thus involves three parameters $p(t|i,z)$, $p(i|u)$ and $p(z|r)$. L is defined as

$$L = \sum_{r,u,t} n(r,u,t) log(p(r,u,t)) \tag{5}$$

EM algorithm is applied to estimate these parameters. In the expectation step, the joint probability of hidden variables I and Z given all observations is computed as

$$p(i,z|u,r,t) = \frac{p(t|i,z)p(i|u)p(z|r)}{\sum_{i,z} p(t|i,z)p(i|u)p(z|r)} \tag{6}$$

Subsequently, each parameter is re-estimated using $p(i,z|u,r,t)$ we just computed from the E step

$$p(t|i,z) = \frac{\sum_{r,u} n(r,u,t)p(i,z|u,r,t)}{\sum_{r,u,t} n(r,u,t)p(i,z|u,r,t)} \tag{7}$$

$$p(i|u) = \frac{\sum_{r,t} n(r,u,t) \sum_z p(i,z|u,r,t)}{n(u)} \quad (8)$$

$$p(z|r) = \frac{\sum_{u,t} n(r,u,t) \sum_i p(i,z|u,r,t)}{n(r)} \quad (9)$$

The algorithm iterates between E and M step until the log likelihood or all parameter values converges.

Once all the models are learned, we use the distribution of topics of a resource $p(z|r)$ to compute similarity between resources and the seed using Jensen-Shannon divergence.

Empirical Validation

To evaluate our approach, we collected data for three seed resources: *flytecomm*[2] *geocoder*[3] and *wunderground*[4]. The first resource allows users to track flights given the airline and flight number or departure and arrival airports; the second resource returns coordinates of a given address; while, the third resource supplies weather information for a particular location (given by zipcode, city and state, or airport). Our goal is to find other resources that provide flight tracking, geocoding and weather information. Our approach is to crawl *del.icio.us* to gather resources possibly related to the seed; apply the probabilistic models to find the topic distribution of the resources; then rank all gathered resources based on how similar their topic distribution is to the seed's topic distribution. The crawling strategy is defined as follows: for each seed

- Retrieve the 20 most popular tags that users have applied to that resource
- For each of the tags, retrieve other resources that have been annotated with that tag
- For each resource, collect all bookmarks that have been created for it (i.e., resource-user-tag triples)

We wrote special-purpose Web page scrapers to extract this information from *del.icio.us*. In principle, we could continue to expand the collection of resources by gathering tags and retrieving more resources that have been tagged with those tags, but in practice, even after the small traversal we do, we obtain more than 10 million triples for the *wunderground* seed.

We obtained the datasets for the seeds *flytecomm* and *geocoder* in May 2006 and for the seed *wunderground* in January 2007. We reduced the dataset by omitting low (fewer than ten) and high (more than ten thousand) frequency tags and all the triples associated with those tags. After this reduction, we were left with (a) 2,284,308 triples with 3,562 unique resources; 14,297 unique tags; 34,594 unique users for the *flytecomm* seed; (b) 3,775,832 triples with 5,572 unique resources; 16,887 unique tags and 46,764 unique users for the *geocoder* seed; (c) 6,327,211 triples with 7,176 unique resources; 77,056 unique tags and 45,852 unique users for the *wunderground* seed.

Next, we trained all three models on the data: pLSA, MWA and ITM. We then used the learned topic distributions to compute the similarity of the resources in each dataset to the seed, and ranked the resources by similarity. We evaluated the performance of each model by manually checking the top 100 resources produced by the model according to the criteria below:

- *same*: the resource has the same functionality if it provides an input form that takes the same type of data as the seed and returns the same type of output data: e.g., a flight tracker takes a flight number and returns flight status
- *link-to*: the resource contains a link to a page with the same functionality as the seed (see criteria above). We can easily automate the step that check the links for the right functionality.

Although evaluation is performed manually now, we plan to automate this process in the future by using the form's metadata to predict semantic types of inputs (Heß & Kushmerick 2003), automatically query the source, extract data from it and classify it using the tools described in (Gazen & Minton 2005; Lerman, Plangprasopchok, & Knoblock 2006). We will then be able to validate that the resource has functionality similar to the seed by comparing its input and output data with that of the seed (Carman & Knoblock 2007). Note that since each step in the automatic query and data extraction process has some probability of failure, we will need to identify many more relevant resources than required in order to guarantee that we will be able to automatically verify some of them.

Figure 3 shows the performance of different models trained with either 40 or 100 topics (and interests) on the three datasets. The figure shows the number of resources within the top 100 that had the same functionality as the seed or contained a link to a resource with the same functionality. The Interest-Topic model performed slightly better than pLSA, while both ITM and pLSA significantly outperformed the MWA model. Increasing the dimensionality of the latent variable Z from 40 to 100 generally improved the results, although sometimes only slightly. Google's find "Similar pages" functionality returned 28, 29 and 15 resources respectively for the three seeds *flytecomm*, *geocoder* and *wunderground*, out of which 5, 6, and 13 had the same functionality as the seed and 3, 0, 0 had a link to a resource with the same functionality. The ITM model, in comparison, returned three to five times as many relevant results.

Table 1 provides another view of performance of different resource discovery methods. It shows how many of the method's predictions have to be examined before ten resources with correct functionality are identified. Since the ITM model ranks the relevant resources highest, fewer Web sites have to be examined and verified (either manually or automatically); thus, ITM is the most efficient model.

One possible reason why ITM performs slightly better than pLSA might be because in the datasets we collected, there is low variance of user interest. The resources were gathered starting from a seed and following related tag links; therefore, we did not obtain any resources that were annotated with different tags than the seed, even if they are tagged

[2] http://www.flytecomm.com/cgi-bin/trackflight/

[3] http://geocoder.us

[4] http://www.wunderground.com/

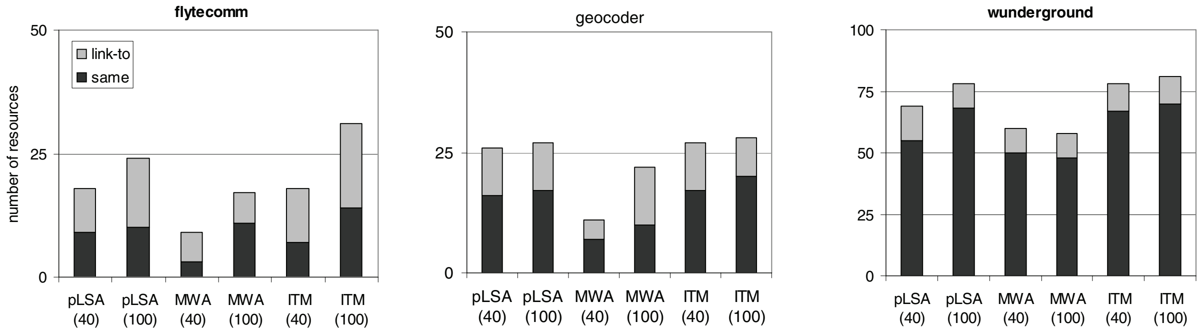

Figure 3: Performance of different models on the three datasets. Each model was trained with 40 or 100 topics. For ITM, we fix interest to 20 interests across all different datasets. The bars show the number of resources within the top 100 returned by each model that had the same functionality as the seed or contained a link to a resource with the same functionality as the seed.

by the same user who bookmarks the seed. Hence user-resource co-occurrences are incomplete: they are limited by a certain tag set. pLSA and ITM would perform similarly if all users had the same interests. We believe that ITM would perform significantly better than pLSA when variation of user interest is high. We plan to gather more complete data to weigh ITM behavior in more detail.

Although performances pLSA and ITM are only slightly different, pLSA is much better than ITM in terms of efficiency since the former ignores user information and thus reduces iterations required in its training process. However, for some applications, such as personalized resource discovery, it may be important to retain user information. For such applications the ITM model, which retains this information, may be preferred over pLSA.

Previous Research

Popular methods for finding documents relevant to a user query rely on analysis of word occurrences (including metadata) in the document and across the document collection. Information sources that generate their contents dynamically in response to a query cannot be adequately indexed by conventional search engines. Since they have sparse metadata, the user has to find the correct search terms in order to get results.

A recent research (Dong *et al.* 2004) proposed to utilize metadata in the Web services' WSDL and UDDI files in order to find Web services offering similar operations in an unsupervised fashion. The work is established on a heuristic that similar operations tend to be described by similar terms in service description, operation name and input and output names. The method uses clustering techniques using cohesion and correlation scores (distances) computed from co-occurrence of observed terms to cluster Web service operations. In this approach, a given operation can only belong to a single cluster. Meanwhile, our approach is grounded on a probabilistic topic model, allowing a particular resource to be generated by several topics, which is more intuitive and robust. In addition, it yields a method to determine how the resource similar to others in certain aspects.

	PLSA	MWA	ITM	GOOGLE*
flytecomm	23	65	15	> 28
geocoder	14	44	16	> 29
wunderground	10	14	10	10

Table 1: The number of top predictions that have to be examined before the system finds ten resources with the desired functionality (the same or link-to). Each model was trained with 100 topics. For ITM, we fixed the number of interests at 20. *Note that Google returns only 8 and 6 positive resources out of 28 and 29 retrieved resources for flytecomm and geocoder dataset respectively.

Although our objective is similar, instead of words or metadata created by the *authors* of online resources, our approach utilizes the much denser descriptive metadata generated in a social bookmarking system by the *readers* or *users* of these resources. One issue to be considered is the metadata cannot be directly used for categorizing resources since they come from different user views, interests and writing styles. One needs algorithms to detect patterns in these data, find hidden topics which, when known, will help to correctly group similar resources together. We apply and extend the probabilistic topic model, in particular pLSA (Hofmann 1999) to address such issue.

Our model is conceptually motivated by the Author-Topic model (Rosen-Zvi *et al.* 2004), where we can view a user who annotate a resource as an author who composes a document. The aim in that approach is to learn topic distribution for a particular author; while our goal is to learn the topic distribution for a certain resource. Gibbs sampling was used in parameter estimation for that model; meanwhile, we use the generic EM algorithm to estimate parameters, since it is analytically straightforward and ready to be implemented.

The most relevant work, (Wu, Zhang, & Yu 2006), utilizes multi-way aspect model on social annotation data in *del.icio.us*. The model doesn't explicitly separate user interests and resources topics as our model does. Moreover, the work focuses on emergence of semantic and personalized resource search, and is evaluated by demonstrating that it can alleviate a problem of tag sparseness and synonymy in a task of searching for resources by a tag. In our work, on the other hand, our model is applied to search for resources similar to a given resource.

There is another line of researches on resource discovery that exploits social network information of the web graph. Google (Brin & Page 1998) uses visitation rate obtained from resources' connectivity to measure their popularity. HITS (Kleinberg 1999) also use web graph to rate relevant resources by measuring their authority and hub values. Meanwhile, ARC (Chakrabarti *et al.* 1998) extends HITS by including content information of resource hyperlinks to improve system performance. Although the objective is somewhat similar, our work instead exploits resource metadata generated by community to compute resources' relevance score.

Conclusion

We have presented a probabilistic model that models social annotation process and described an approach to utilize the model in the resource discovery task. Although we cannot compare to performance to state-of-the-art search engine directly, the experimental results show the method to be promising.

There remain many issues to pursue. First, we would like to study the output of the models, in particular, what the user interests tell us. We would also like to automate the source modeling process by identifying the resource's HTML form and extracting its metadata. We will then use techniques described in (Heß & Kushmerick 2003) to predict the semantic types of the resource's input parameters. This will enable us to automatically query the resource and classify the returned data using tools described in (Gazen & Minton 2005; Lerman, Plangprasopchok, & Knoblock 2006). We will then be able to validate that the resource has the same functionality as the seed by comparing its input and output data with that of the seed (Carman & Knoblock 2007). This will allow agents to fully exploit our system for integrating information across different resources without human intervention.

Our next goal is to generalize the resource discovery process so that instead of starting with a seed, a user can start with a query or some description of the information need. We will investigate different methods for translating the query into tags that can be used to harvest data from *del.icio.us*. In addition, there is other evidence potentially useful for resource categorization such as user comments, content and input fields in the resource. We plan to extend the present work to unify evidence both from annotation and resources' content to improve the accuracy of resource discovery.

Acknowledgements This research is based by work supported in part by the NSF under Award No. CNS-0615412 and in part by DARPA under Contract No. NBCHD030010.

References

Brin, S., and Page, L. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems* 30(1–7):107–117.

Carman, M. J., and Knoblock, C. A. 2007. Learning semantic descriptions of web information sources. In *Proc. of IJCAI.*

Chakrabarti, S.; Dom, B.; Gibson, D.; Kleinberg, J.; Raghavan, P.; and Rajagopalan, S. 1998. Automatic resource list compilation by analyzing hyperlink structure and associated text. In *Proceedings of the 7th International World Wide Web Conference*.

Dempster, A. P.; Laird, N. M.; and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 39(1):1–38.

Dong, X.; Halevy, A. Y.; Madhavan, J.; Nemes, E.; and Zhang, J. 2004. Simlarity search for web services. In *Proc. of VLDB*, 372–383.

Gazen, B. C., and Minton, S. N. 2005. Autofeed: an unsupervised learning system for generating webfeeds. In *Proc. of K-CAP 2005*, 3–10.

Heß, A., and Kushmerick, N. 2003. Learning to attach semantic metadata to web services. In *International Semantic Web Conference*, 258–273.

Hofmann, T. 1999. Probabilistic latent semantic analysis. In *Proc. of UAI*, 289–296.

Kleinberg, J. M. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM* 46(5):604–632.

Lerman, K.; Plangprasopchok, A.; and Knoblock, C. A. 2006. Automatically labeling the inputs and outputs of web services. In *Proc. of AAAI.*

Lerman, K.; Plangprasopchok, A.; and Knoblock, C. A. 2007. Semantic labeling of online information sources. *International Journal on Semantic Web and Information Systems, Special Issue on Ontology Matching*.

Lin, J. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory* 37(1):145–151.

Mathes, A. 2004. Folksonomies: cooperative classification and communication through shared metadata.

Popescul, A.; Ungar, L.; Pennock, D.; and Lawrence, S. 2001. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *17th Conference on Uncertainty in Artificial Intelligence*, 437–444.

Rosen-Zvi, M.; Griffiths, T.; Steyvers, M.; and Smyth, P. 2004. The author-topic model for authors and documents. In *AUAI '04: Proceedings of the 20th conference on Uncertainty in artificial intelligence*, 487–494. Arlington, Virginia, United States: AUAI Press.

Thakkar, S.; Ambite, J. L.; and Knoblock, C. A. 2005. Composing, optimizing, and executing plans for bioinformatics web services. *VLDB Journal* 14(3):330–353.

Wu, X.; Zhang, L.; and Yu, Y. 2006. Exploring social annotations for the semantic web. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, 417–426. New York, NY, USA: ACM Press.