World Wide Web (2009) 12:421–440
DOI 10.1007/s11280-009-0069-1

The Effectiveness of Latent Semantic Analysis for Building Up a Bottom-up Taxonomy from Folksonomy Tags

Takeharu Eda · Masatoshi Yoshikawa · Toshio Uchiyama · Tadasu Uchiyama

Received: 16 December 2008 / Revised: 6 April 2009 /
Accepted: 24 July 2009 / Published online: 12 August 2009

Abstract In this paper, we evaluate the effectiveness of a semantic smoothing technique to organize folksonomy tags. Folksonomy tags have no explicit relations and vary because they form uncontrolled vocabulary. We discriminates so-called subjective tags like "cool" and "fun" from folksonomy tags without any extra knowledge other than folksonomy triples and use the level of tag generalization to form the objective tags into a hierarchy. We verify that entropy of folksonomy tags is an effective measure for discriminating subjective folksonomy tags. Our hierarchical tag allocation method guarantees the number of children nodes and increases the number of available paths to a target node compared to an existing tree allocation method for folksonomy tags.

Keywords folksonomy · tag · LSI · bottom-up taxonomy · classification

1 Introduction

Web services like blogs and wiki have become very popular. In these services, many entries are updated frequently and many links are generated automatically, degrading the value of web links. Web hyperlink graphs have been an effective

T. Eda (✉) · T. Uchiyama · T. Uchiyama
NTT Cyber Solution Laboratories, NTT Corporation, Kanagawa, Japan
e-mail: eda.takeharu@lab.ntt.co.jp

T. Uchiyama
e-mail: uchiyama.toshio@lab.ntt.co.jp

T. Uchiyama
e-mail: uchiyama.tadasu@lab.ntt.co.jp

M. Yoshikawa
Kyoto University, Kyoto, Japan
e-mail: yoshikawa@i.kyoto-u.ac.jp

sources for weighting web pages for search enhancement since web page owners choose the links in their page by themselves [1]. Splogs, which are created for the purpose of raising the PageRank of target sites, make link analysis and weighting of web pages difficult [12].

Schemes for overcoming these drawbacks and creating the next web page weighting technique [9, 28]), social bookmark services (SBMs) (e.g. del.icio.us,[1] BibSonomy,[2] Hatena Bookmark[3]) have attracted a lot of attention. They use the bottom-up taxonomy system called *folksonomy*. SBMs offer bookmarking functions to the user and store bookmark entries on a server. Users can attach tags with varying numbers of keywords to their entries and set comments on entries. Compiling bookmark entries is a promising way for ranking fresh user-chosen web pages [28], because the number of times an entry is bookmarked by users can be considered as a measure of the popularity of the entry. Users are indirectly connected and make implicit communities by bookmarking the same entries or using the same tags [18].

Folksonomy can be considered as one way for end users to participate in the classification of web resources (URLs). Tags are viewpoints selected by users and are intended to be used as clues to find resources [17]. Bookmarking URLs is such a common activity that for almost all net surfing users have participated at some time. In many SBMs, interfaces to participate in them are provided as web browser plug-ins and have almost the same usability as the default bookmarking functions of browsers. Because the gap between users and SBM systems is small, users can easily start using SBMs and tagging URLs. Such flexibility is one of the reasons why folksonomy has become the largest and most diverse resource annotating system yet invented [9]. We believe that, in the future, the emergence of folksonomy will considered to be an epoch-making system for annotating the huge volume of information released by the information explosion.

Since the folksonomy idea is simple and has much potential, a lot of researchers have paid it close attention. One tried to disambiguate tags [30], another tried to allocate folksonomy tags into a tree [9] and another utilized it for better information retrieval [31]. The problem tackled in this paper is a better approach to organize the huge number of tags in folksonomy by automatically allocating them into a hierarchy and estimating tag objectivity. Tag hierarchy composed of semantically related tags is useful in many applications like trend analysis, improved navigation tools for popular articles and so on. So-called subjective tags, like "cool" and "fun" should be handled differently from objective tags like "web", "mac" which would be accepted as classifying objects by many users.

In order to handle the explosion in the number of folksonomy tags and utilize them more effectively, we propose here a organizing method based on Probabilistic Latent Semantic Indexing(PLSI). Based on the hidden semantics identified by PLSI, we discriminate subjective tags from all folksonomy tags. We then form the objective folksonomy tags into a hierarchy by extracting the relationships among them. We compute the feature vectors from the co-occurrence data of folksonomy and measure

[1] http://delicious.com

[2] http://www.bibsonomy.org/

[3] http://b.hatena.ne.jp

the generalization differences between them. Our method enables users to intuitively understand the connections among tags. Our contributions are as follows.

- We propose a novel method that can set folksonomy tags into a DAG (Directed Acyclic Graph). We calculate feature vectors using Probabilistic Latent Semantic Indexing (PLSI), quantify them using probabilistic vectors, and estimate abstract upper or lower level differences (generalization) among tags by comparing the entropy values of the tags.
- We introduce a simple criterion, entropy, for discriminating subjective tags from users' classification decisions without recourse to the contents of the tagged objects.
- Our experiments show that entropy, especially that of PLSI vectors is a good criterion for identifying subjective tags from folksonomy tags. Furthermore, we compared our hierarchy assignment method to the existing tree assignment method based on a sparse folksonomy vector model, where not only the hierarchical structure but also the vector models are evaluated. As a whole, the statistical method (PLSI) can grasp the hidden semantics underlying the observed bookmarks and can be used as the basis for effectively organizing folksonomy tags.

The paper is organized as follows. In Section 2, we survey related work on folksonomy and define our research problems in Section 3. In Section 4 we propose our tag organization method which is composed of subjective tag discrimination and hierarchical assignment. Section 5 reports the results of qualitative experiments. Finally, we describe future work and conclude the paper in Section 6.

2 Related work

The social web services that support user online activities and friendships continue to increase in number. Folksonomy is attracting a lot of attention as a way of accumulating users' viewpoints on resources and is expanding its role from tagging URLs to tagging pictures, movies, papers, and so on. Some of these services offer user scores and comments as well as tags.

A lot of studies have been done on folksonomy [22, 26]. Brooks and Montanez [2] proposed automatic tagging based on content and discussed the hierarchical assignment of tags. Since it is possible to consider a tag hierarchy as an ontology, one of their conclusions that such a technique is useful in real world applications, motivates our research. However, they also noted that tree structures are too restrictive to be used easily as a user navigation map.

We can understand the structure and usage of folksonomy systems from the several case studies on folksonomy data [9, 15]. Golder and Huberman [7] analyzed the distributions of tags, users, and resources in del.icio.us and showed that different users act differently. One of their discoveries, that there are seven kinds of tags used in real folksonomy systems, is a key part of our research. We divide their seven kinds into two subjective and objective, and propose a method for estimating tag subjectivity in this paper. Sen et al. [24] did a case study of MovieLens and found that each user tended to be careful with tag assignment. Niwa et al. [19] consider folksonomy to be a method of making web page recommendations. They insisted

that synonym and ambiguity problems can be solved by clustering tags, but they did not consider the applicability of tag-based classification systems. Dubinko et al. [4] visualize frequently used tags in a time series. Since SBMs keep gathering fresh bookmark entries, users can see and browse the history of popular URLs by using such timeline visualizations.

To solve the problem of synonyms and ambiguity in tags, Wu et al. [27] proposed a probabilistic indexing model for folksonomy triples (user, tag, and resource). One of the main advantages of their method is the ability to index all users, tags and resources at the same time. Another method for tag disambiguation is based on bipartite network analysis [30]. The focus of their research can be considered as realizing our goal of constructing a bottom-up taxonomy.

Tagcloud used in many folksonomy services can be considered as one simple way of organizing a huge number of tags by picking popular tags and ordering them according to their size (popularity). Sinclair and Cardew-Hall [25] found tagclouds are not enough to help users navigate. Rivadeneira et al. [21] performed evaluation studies on impression formation. As for goal-orientated tasks, simple alphabetical word lists are preferred over tagclouds [16]. Our organization method proposed in this paper has possibilities to enhance the tagcloud interface using rich information of tag hierarchy network and subjectivity estimation [5].

If we consider that the problem is the automatic organization of tags by computers, the obvious solution is to use flat (hierarchical) clustering algorithms [2, 11]. However, flat (hierarchical) clustering raises the unclear problem of determining the labels for each cluster (inner node).

Heymann and Garcia-Molina [8] proposed a method of assigning tags in a tree based on the *folksonomy vector model*.[4] In this model, tag vectors are composed of the number of people who tagged each URL. The dimension of the tag vector is the number of tagged URLs in the data set. After extracting tag vectors from folksonomy triples, their method uses cosine similarities to measure the distances among tags and calculates the centrality of each tag by considering whole tag sets as a network. They use betweeness centrality, which is easy to approximate. They proposed the extensible greedy algorithm where tags are sorted in this order of centrality, and the algorithm starts from the most central tag and builds a tree in top-down manner. However, their method has several problems. One is the problem of determining the position of a tag to another tag as a child. That is, a node must always have one parent node. Consider real classification systems like Open Directory[5] and Google Directory:[6] we find it natural that several nodes have several parents, since this makes it easier for users to find resources. Another problem is the sparseness of folksonomy vectors, which causes mis-allocation of nodes near the leaf. Figure 1 shows a Heymann tree that is rooted at "mac".[7] We see that several tags around the leaves are far from the topic of "mac".

[4] Later we followed and extended their research [6].

[5] http://www.dmoz.org

[6] http://directory.google.com/

[7] This was drawn in the setting described in Section 5.

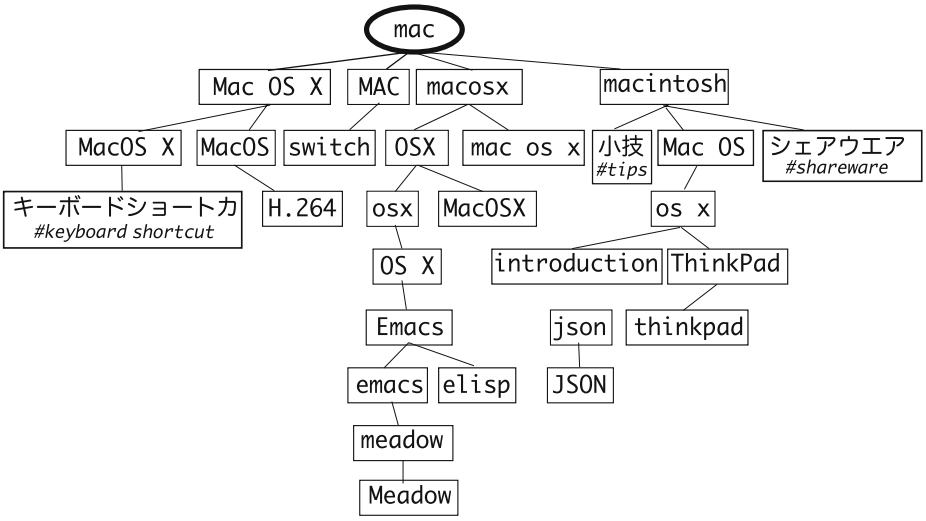

Figure 1 Heymann's tree (**EXT** approach), built by the extensible greedy algorithm.

3 Problem definition

Folksonomy data has a flat structure. One URL is classified by a user with a tag, yielding a triple composed of user, tag, and URL. If a user sets two tags, two triples are created by the same user for the same URL. A set of folksonomy triples collected in a social bookmark service forms a tripartite graph [18].

Users tag URLs using their vocabulary; they can use "apple", "intel mac", and "cool" for the same URL according to their context or feeling. That is, folksonomy tags are inherently personal and subjective. However, viewing a large number tags entered over some period of time is expected common agreements with regard to the tags set for URLs by diverse users. We call these tags objective in this study. Examples of objective tags are "apple" and "intel mac" while "cool" is a subjective tag.

By considering the number of times of one URL is given the same tag, we can define a distance (or similarity) between tags. Such a distance is used in *related tag* interfaces.[8] A related tag interface is a first step towards better navigation via folksonomy tags. This study assumes that dividing related tags into two classes, general or concrete tags, will be beneficial for users.

By organizing folksonomy tags effectively, users can receive many benefits including better browsing of popular entries, and easier retrieval to desired articles. The two organization approaches proposed in this paper are the identification of subjective tags and the automatic creation of tag hierarchy from folksonomy tags.

[8] In fact, `delicious.com` offers related tag links in the tag page.

3.1 Estimation of tag subjectivity

Golder et al. reported there are seven kinds of tags in real folksonomy systems [7].

1 Identifying what(or who) it is about. (e.g. `web, mac, programming`)
2 Identifying what it is. (e.g. `blog, journal, article`)
3 Identifying who owns it. (e.g. `Mike, Tim`)
4 Identifying qualities or characteristics. (e.g. `fun, cool, ***`)
5 Refining Categories. (e.g. `2008, 11`)
6 Self Reference. (e.g. `mystuff`)
7 Task Organizing. (e.g. `todo, toread, toblog`)

It is impossible to know a tag's kind from just the stored tag itself without any user context. We cannot determine whether a URL tagged with "blog" is a blog itself, or a discussion group on blog. However, we understand the difference between the first three tags and the last four tags. The former are related to the URL's object itself, while the latter are mainly related to the user. The usage of the former is expected to achieve consensus by many users but the usage of latter will vary with the user. Our assumption is that the first (second) group provides more objective (subjective) information.

To say that one group is more important than the other does not lie within the scope of this paper. However we can imagine several application scenarios by discriminating subjective/objective tags. If we want *personal* organization, where tags are organized adapted to each users tagging behaviour considering the global tagging tendency, or *emotional* organization, where sentimental tags are keys for organizing tags, the importance of subjective tags (especially 4. and 5.) will dominate. For the purpose of traditional classification problem context, just objective tags are important and subjective tags are noisy. Thus, we tackle the problem defined below in this paper.

Problem 1 *Discriminate* (*not eliminate*[9]) *subjective tags from folksonomy tags.*

Note that our approach does not utilize the content pointed to by the URLs.

3.2 Hierarchical allocation of folksonomy tags

By grouping semantically-related tags, we get a bird-eyed view of the enormous number of tags and more choices related to the tag that the user is currently interested in. Considering more sophisticated related tag interface, it is reasonable to measure the semantic relationships of tags and use the relationships to form a tag hierarchy. We define this problem as follows.

Problem 2 *Create a hierarchy of folksonomy tags that reflects users' classification decisions.*

[9] As one motivation for using subjective tags for creating an *emotional* hierarchy, we show a DAG with subjective tags in Section 5.

The semantics of the hierarchy is not explicitly defined but we can say that ambiguous tags, which often co-occur with other tags, are more general than unambiguously clustered tags in all of the tags. We show some example hierarchies created by our method in Section 5.

The relationship of the above two problems and the processing flow for organizing folksonomy tags is shown in Figure 2 in a comparison to related work by Heymann et al. At first, we discriminate subjective tags by solving Problem 1, and the discriminated objective tags are used to build a tag hierarchy.

4 Proposed method

We propose a novel tag organizing method which is composed of the identification of subjective tags and hierarchical allocation into DAG. The overall processing flow is compared in Figure 2 to Heymann et al.'s approach [8]. Heymann et al. does not discuss the subjective tag discrimination step but we consider the utilization of this step for creating a better organized bottom-up hierarchy.

DAG (Directed Acyclic Graph) is a graph structure where a node can have multiple parents in contrast to trees that allow each node to have only one parent. We employ DAG as the classification hierarchy since it is usual for categories to belong to several parent categories in many real classification systems. We show an example DAG where a tag belongs to multiple more general tags in Section 5.

To form the folksonomy tags into a hierarchy, both proposals consist of two steps: tag vectorization and hierarchical allocation of the vectors. In the following, our approach, the PLSI vectors and the DAG allocation algorithm, is referred to as **DAG** and Heymann's approach, the folksonomy vectors and the extensible greedy algorithm, is referred to as **EXT**. At first, we apply PLSI to calculate feature vectors(PLSI vectors). PLSI can expose the hidden relationships among item sets in a statistical way, thus yielding better precision than naive folksonomy vectors.

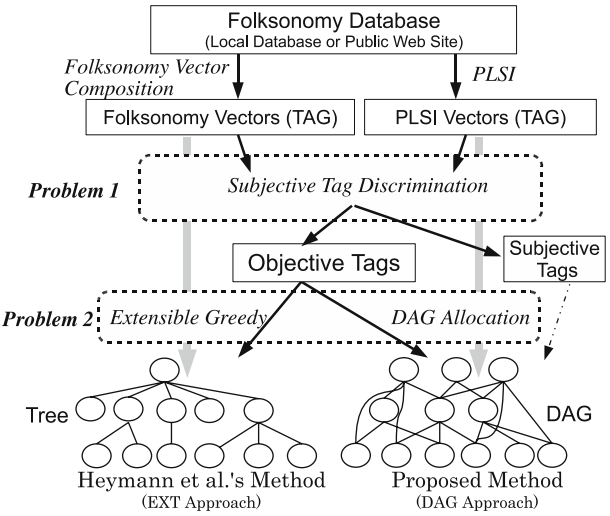

Figure 2 Processing flow of tag organization method consists of subjective tag discrimination and hierarchical allocation.

PLSI vectors are allocated in DAG by determining children of each PLSI vector. We find k nearest neighbors of each vector and set each neighbor as a child if its level of generalization is lower than that of the parent. We estimate the level of generalization from the entropy value of each PLSI vector. We use three-mode PLSI [27] to handle the 3-tuple co-occurrence data consisting of user, tag, and resource. Although PLSI vectors for user, tag, and resource are calculated at the same time, we use only PLSI tag vectors.

We start by describing the preparatory steps. A full explanation of how our method solves the two problems is then given.

4.1 Preparation

4.1.1 Tag vector (folksonomy vector model)

In the folksonomy vector model [8], each tag t_i is represented as a vector V_{t_i} where each value N_{t_i,r_j} is the number of people who assigned the tag to URL r_i.

$$V_{t_i}^f = \left(N_{t_i,r_1}, N_{t_i,r_2}, N_{t_i,r_3}, \cdots, N_{t_i,r_{|R|}}\right)$$

Since the length of folksonomy vectors $|R|$ equals the number of URLs in folksonomy systems, the vector model is very sparse, like document vector model [23]. Heymann and Garcia-Molina [8] used the *cosine* similarity measure since it is a practical approach to assessing the similarity between vectors.

$$sim(V_{t_i}^f, V_{t_j}^f) = \sum_{k=1}^{|R|} N_{t_i,r_k} \cdot N_{t_j,r_k}$$

4.1.2 Indexing tags using PLSI

PLSI enables us to index co-occurrence data (subsets of the direct product space of several item sets) with given dimension feature vectors. This can be understood as mapping each set into a semantic vector space. Another view is the clustering of both documents and terms via *soft* membership functions.

Although PLSI was first developed for two-mode co-occurrence data (document×term) as a probabilistic variant of LSI [10], it has been extended to three-modes in several studies [14, 27]. By calculating distances among PLSI vectors, we can index the closeness of items. Since PLSI maps item sets into the semantic vector space, the closeness among PLSI vectors is based on their closeness in the semantic space.

Figure 3 shows the model for PLSI used in our method. U, R, and T stand for user set, resource set, and tag set, respectively. The dimension of a semantic space is D. Occurrence probability from $\alpha(\in [1, 2, \ldots, D])$ to $u_i \in U$, $r_j \in R$, $t_k \in T$ is denoted by $p(u_i|\alpha)$, $p(r_j|\alpha)$, $p(t_k|\alpha)$. We assume that occurrence probability of α is $p(\alpha)$. The equation below holds for u_i, r_j, t_k, and the semantic space.[10]

$$p(u_i, r_j, t_k) = \sum_{\alpha=1}^{D} p(\alpha)\, p(u_i|\alpha)\, p(r_j|\alpha)\, p(t_k|\alpha)$$

[10] $p(u_i, r_j, t_k)$ stands for occurrence probability of the triple.

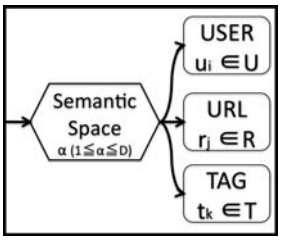

Figure 3 Probabilistic model (Wu et al. [27]).

The left-hand side can be observed from the data set, and probabilities in the right-hand side can be estimated by using the EM algorithms with appropriate initial values [27]. EM algorithm consists of E (Expectation) step and M (Maximization) step. These steps are alternated repeatedly until convergence is achieved. EM algorithm is assured to increase the log likelihood, but it does not necessarily converge to global optimum. In our implementation, a tempered EM algorithm is used to avoid local optimums, where temperature parameter is used to avoid overfitting [3].

Using Bayes' rule, we can calculate the conditional probabilities from items to the semantic space.

$$p(\alpha|t_i) = \frac{p(t_i|\alpha) \cdot p(\alpha)}{p(t_i)}$$

By considering this probability as an α dimension value in a vector space, all tags are vectors in a D dimensional vector space.

$$V_{t_i}^{p} = \Big(p(1|t_i),\, p(2|t_i),\, p(3|t_i), \cdots,\, p(D|t_i)\Big)$$

Examples of calculated PLSI vectors Figure 4 shows examples of PLSI vectors where the dimension of semantic space D equals 80. The horizontal and vertical axes show dimension and probability, respectively. All tags in Figure 4 related to "mac os x" have high probability around the 75-th dimension. Semantically-related tags are understood visually as tags which have high probability at one or more common dimensions. Through these semantic relationships, PLSI can better identify the hidden relationships between tags.

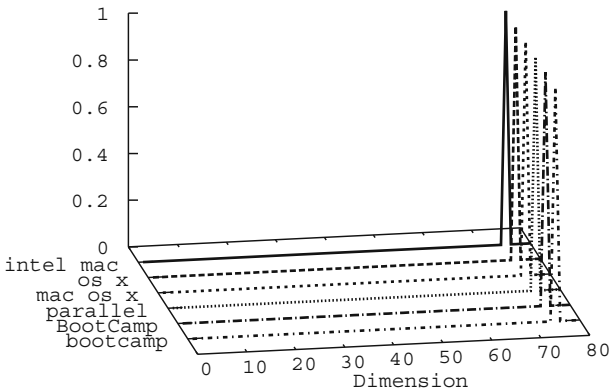

Figure 4 Example of low entropy PLSI vectors; they have high probability at 75th dimension.

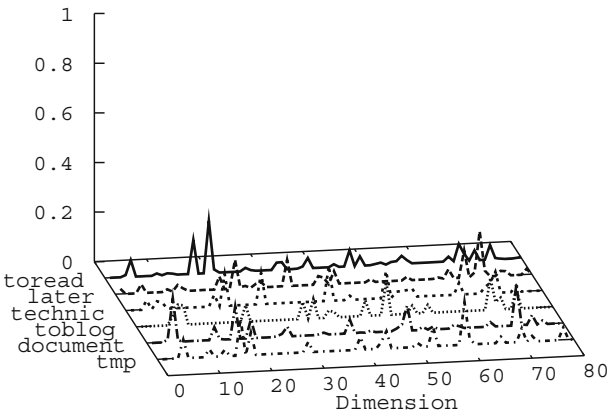

Figure 5 Example of high entropy PLSI vectors; they are spread over many dimensions.

Distances among PLSI vectors In order to measure tag similarity in PLSI vector space, we need the distance among PLSI vectors. In this study, we employ JS divergence, which is a distance between probabilistic distributions and is considered to offer better accuracy in information retrieval tasks [13]. JS divergence is calculated as follows.[11]

$$distance(V_{t_k}^p, V_{t_l}^p) = D_{js}(t_k, t_l) = \frac{1}{2}\left[D(t_k||avg(t_k, t_l)) + D(t_l||avg(t_k, t_l))\right]$$

4.1.3 Entropy of vectors

Entropy is a measure of the uncertainty of the information sources. For both vector models, we define entropy as follows.

$$\textbf{[Tag Entropy]} \quad H_{tag}(t_i) = -\sum_{j=1}^{|R|} \frac{N_{t_i,r_j}}{\sum_{k=1}^{|R|} N_{t_i,r_k}} log \frac{N_{t_i,r_j}}{\sum_{k=1}^{|R|} N_{t_i,r_k}}$$

$$\textbf{[PLSI Entropy]} \quad H_{plsi}(t_i) = -\sum_{\alpha=1}^{D} p(\alpha|t_i) \cdot log(p(\alpha|t_i))$$

The minimum entropy value is 0 and occurs when the vector has only one peak; the value increases as the distribution approaches uniformity. Entropy is considered to be a measure of uncertainty of the probabilistic distribution. Figure 5 shows an example of PLSI vectors that have high entropy values, while Figure 4 shows an example of low entropy vectors. Low entropy PLSI vectors have high membership probability at just a few dimensions; high entropy PLSI vectors, on the other hand, are spread over many dimensions.

[11] $D(q||r)$ is KL divergence between q and r.

4.2 Estimation of tag subjectivity

Our assumption on subjective tags' usage is that it depends on each user. In other words, subjective tags are not attached to a set of similar URLs but to diverse URLs; this means the tag is ambiguous.

A subjective tag is not necessarily an unpopular nor personal tag and is ambiguous regardless of the number times it is used. Examples of such popular but subjective tags are "cool", "fun", and so on, which are used often but remain very subjective. This assumption leads to the simple, but reasonable approach of using entropy as a measure of tag subjectivity.

In PLSI vector model, we also use its entropy as a measure of tag subjectivity. However, the assumption is slightly different. A subjective tags should be diffuse in many semantic dimensions.

Our strategy for estimating tag subjectivity is as follows.

Subjective Tag Estimation Strategy 1 *We treat a folksonomy tag as a subjective tag if its entropy is high in either of the vector models.*

4.3 DAG allocation

DAG is a directed graph with no directed cycle. Any node leads to an terminal node if the number of nodes in DAG is finite. DAG is intuitively considered to exhibit node flow or order of nodes. Figure 6 shows a DAG constructed from objective tags by our method. "mac" is the most general tag in the DAG and "mac os x", "boot camp", and so on are more specific than "mac".

We use the entropy value of PLSI vectors to create this flow by allocating tags to a DAG. Table 1 shows the DAG allocation algorithm, which is quite simple. For all nodes, we set children and parents with `divide_neighbors`, in which, we first search for the nearest neighbors and designate the neighbor node as a child if its

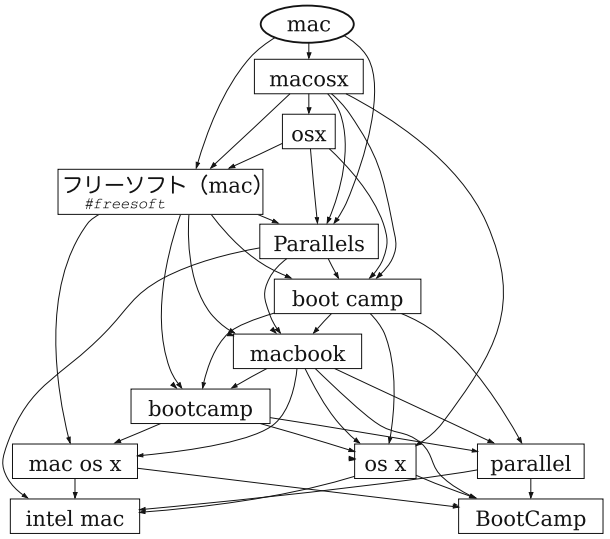

Figure 6 A DAG rooted at "mac" (**DAG** approach).

Table 1 DAG allocation algorithm.

input: PLSI vector list. $\{v_i\}_{i=1}^{i=m}$
output: DAG where PLSI vectors are allocated in.
1 **for**$(i = 1; i < m + 1; i+ = 1)$
2 `divide_neighbors`(v_i, k, n)
3 **endfor**

Function: `divide_neighbors`
input: Node: v, Threshold: k, Number of Chidren: n
1 $\{w_i\}_{i=1}^{i=n}$: Get all n nodes near to v of the distance within k
2 **for**$(i = 1; i < n + 1; i+ = 1)$
3 if$(w_i.entropy < v.entropy)$ **then**
4 Set w_i as a child of v.
5 **else**
6 Set w_i as a parent of v.
7 **endif**
8 **endfor**

entropy is less than the node's entropy. If the entropy is more than that of the node, the neighbor node is designated as a parent. If they are the same, special handling is required. In our setup, we used the alphabetic order of tags.[12] By our algorithm, entropy values decrease from root to leaf, since the entropy of a child is always less than that of its parent. In the data structure constructed by our algorithm, it is impossible to go back to the starting node when navigating directed edges from parent to child. That means that the constructed data structure is a DAG.

One advantage of DAG is that the user can never return to an already visited node. The allocation algorithm in Table 1 chooses neighbors whose distances are less than a certain threshold. However, sometimes no children are found. In that case, the condition can be extended to the k nearest neighbor nodes depending on the distribution of JS divergence values. We discuss the distribution of distances in Section 5.

The time complexity of the DAG allocation algorithm is $O(n^2)$, which matches that of the extensible greedy algorithm [8]. Though the extensible greedy algorithm is an offline algorithm, which needs to construct the whole hierarchical structure beforehand, our algorithm is an online algorithm that enables us to expand children locally around the required node.

5 Evaluation

We conducted several experiments to assess the performance of our approach in the following areas.

1. Correlation and difference between cosine similarity and JS divergence.
2. Precision of subjectivity estimation by entropy.
3. Precision of children nodes calculated by DAG and the extensible greedy algorithm.

[12] In practice, there are few cases where entropy values coincide.

We compared our method with the extensible greedy algorithm using the folksonomy vector model [8]. Both approaches are described in Table 2.

As the data set, we crawled popular entries at Hatena Bookmark in October 2006. We got a list of popular tags from `http://b.hatena.ne.jp/t` and retrieved a set of URLs that were linked from the site. HTML files were scraped into folksonomy triples, which were then stored in a RDBMS. Before applying PLSI, we eliminated the users, tags, and resources that appeared less than five times (and thus triples containing these items), for the purpose of noise reduction. The statistics of data set is shown in Table 3. 4019 resources are bookmarked by 11713 users with 5496 tags. We count the number of unique triples composed of an user, a tag and an URL. 51.6% of tags are in Japanese and others are in English. This means that it is common for Japanese users to classify URLs with English keywords.

All the parameters of the algorithms were chosen empirically as shown in Table 4. EM iterations were chosen with the same setting used in [27]. We tried several dimensions for the semantic space and chose the most accurate dimension by checking the kNN of tags sampled by users. **DAG**'s thresholds k and n were determined in an ad-hoc manner since **DAG** has the ability to expand children locally. In the case of **EXT**, we tried several thresholds and set the threshold to avoid the very shallow tree.[13]

5.1 Differences in distances

Figure 7 shows the distribution of distances between 300 randomly chosen tags;[14] the horizontal axis represents cosine similarity and the vertical axis represents JS divergence. Although, there is no clear association between cosine and JS divergence, we notice that there were several points whose cosines equaled 0 but whose JS divergence varied over a wide range. To verify this point, we calculated the cases wherein the two tags were maximally distant from each other, that is, the cosine similarity equals 0 and the JS divergence equals 1. The result is that almost half (45%) of the folksonomy vector pairs in the data set were maximally distant from each other. In contrast, only a few (7%) of the PLSI vectors were maximally far from each other. This means that the statistical method (PLSI) enables us to identify the tags that are most widely separated in the folksonomy vector model.

Revealed Connections Table 5 shows tag pairs whose cosine similarity equals 0 and whose JS divergence is near 0. Tag pairs are randomly chosen, and only English words are shown.[15] Using **EXT** approach we were unable to detect the relationships shown in Table 5, that were detected by our **DAG** approach.

5.2 Tag subjectivity estimation

Figure 8 shows the subjective tags observed in the data set.[16] Answer set were created by user. All tags were sorted in decreasing number of times used (`TF`), tag entropy

[13] If we increase the threshold of **EXT**, many nodes become children of the root.

[14] 300 were chosen just for the graphic drawing.

[15] Since Hatena bookmark is a Japanese bookmark service, many tags are in Japanese.

[16] All Japanese tags are translated into English and shown like (J) "tag".

Table 2 **DAG** and **EXT** approaches.

	DAG	**EXT**
Vector	PLSI vector	Folksonomy vector
Distance	JS divergence	Cosine similarity
Data structure	DAG	Tree

Table 3 Data set.

Resources	4019
Tags	5496
Users	11713
Triples	1190504

Table 4 Parameters.

DAG	Dimension of semantic space	80
	EM iterations	80
	Threshold: k	0.2
	Num. of children: n	11
EXT	Threshold: k	0.00001

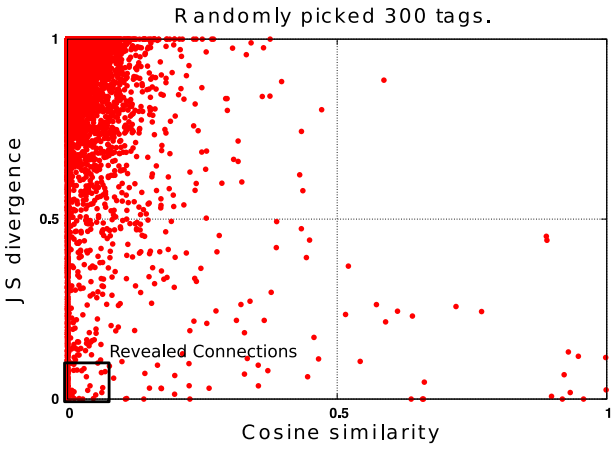

Figure 7 Distribution of Cosine and JS Divergence.

Table 5 Revealed connections.

CRC	partition	recover	nLite
CSS3	xhtml	VB.NET	Trac
KDDI	Softbank	Winny	Privacy
Attention economy	nagios	KDDI	WILLCOM
mod_proxy_balancer	rrdtool	3DCG	flash
j2ee	BDD	prototype	wysywig
ASP.NET	Trac	WinFS	foldershare
Rails	memcached	Plagger	LINUX
GoogleEarth	google mars	trac	unison

(Example pairs of tags cosine similarity equals 0 and JS divergence is near 0.) For example, KDDI, Softbank and WILLCOM are Japanese telecommunication companies. attention economy does not seem to related to nagios, which is a software program. However, one function of nagios is to monitor computer servers, and so has a hidden relationship to attention. These trends are applicable to tags in Japanese.

Figure 8 Observed subjective tags.

KIND	TAG
4	*** interesting useful (J) great cool ('A') *#Japanese smiley* ...
5	1,2,3,4,5 ...
6	memo ...
7	todo toread (J) to read later ...

(Entropy), and PLSI entropy (PLSI Entropy) and then checked as to whether it was in the answer set or not.

The precision-recall curve is shown in Figure 9. From the results, the number of times a tag is used has no relationship to whether it is subjective or objective. Both entropy values provide a positive result with regard to tag subjectivity estimation. In particular, PLSI entropy is the best criterion for identifying subjective tags.

Spearman's rank correlation coefficient between both entropy ranks was 0.77, which means that there is a correlation between tag ranking score in tag and PLSI entropy. Of the top 100 tags, 37% and 44% were determined according to tag entropy and PLSI entropy, respectively, as subjective. Table 6 lists some examples of objective tags that have high entropy as defined by tag entropy. PLSI entropy effectively lowers the rank of these tags.

Figure 9 Precision-recall curve of subjective tag identification.

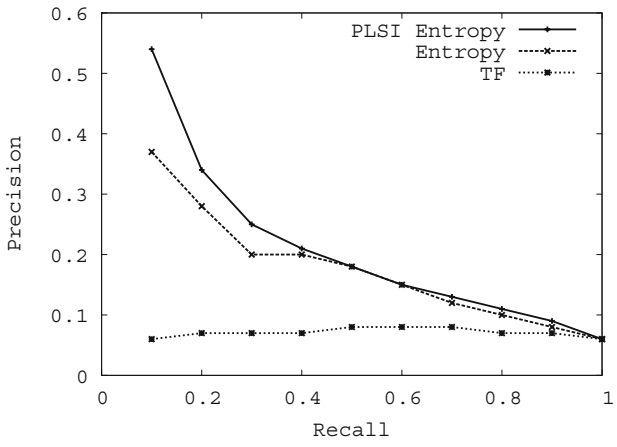

Table 6 High entropy objective tags.

	Tag entropy rank	PLSI entropy rank
list	4	226
(J) information	5	28
japan	6	61
site	7	131
(J) foreign countries	8	23
(J) reference	10	105

5.3 Comparison of local structures

We randomly sampled tags and compared the number of children nodes rooted at the tag. For all sampled tags, we counted the number of child nodes in both methods and evaluated the precision of child nodes by hand. Since neither **DAG** or **EXT** aim to extract parent-child relationships like *is-a* and *part-of*, the correct answers were judged from the points of relatedness and ancestor-descendant relationship.[17]

The average number of child nodes and the average precision are shown in Figure 10. **DAG**'s average precision is 11% less than that of **EXT**. However, **DAG**'s average number of child nodes is eight. This means that the **DAG** approach has 2.7 times as many child nodes for just an 11% decrease in precision compared to **EXT**. Furthermore, **DAG**'s support of *local expansion* is confirmed by the results in Figure 11, which shows that the relatedness from a tag to its children does not decrease significantly as the number of children increases.

5.4 Example DAG structures

Figure 12 shows a DAG expanded from "HDD" upward[18]. "HDD" has 5 parents; "(J)*trouble*", "(J)*customize*", "(J)*backup*", "PC", and "CD", which are not necessarily parents in their semantics, but we understand these context because all parents are often discussed with "HDD". Furthermore, we notice there are three general topics; "(J)*internet*, "computer", and "(J)*goods* above "HDD".

As an example of the motivation to use subjective tags in a hierarchy, Figure 13 shows the upper structure from "2.0" with the nearest subjective tag "(J) *interesting*". Tag "cnet japan" is a Japanese news site that translated a famous article "*What Is Web* 2.0" by Tim O'Reilly. Subjective tags have possibility of allowing users popularity/reputation to be included in a tag hierarchy.

These hierarchies are bottom-up taxonomies built from only users' classification activities.

5.5 Summary

The PLSI vector model uses the JS divergence distance to reveal the hidden relationships among tags. Several software programs which run on Apple's "mac" do

[17] If a child was synonym, we considered it correct in both methods.

[18] We used graphviz (http://graphviz.org) to visualize all the graphs in this paper.

(average)	Num. of Children	Precision
DAG	8.00	0.64
EXT	3.00	0.75

Figure 10 Average number of children and precision.

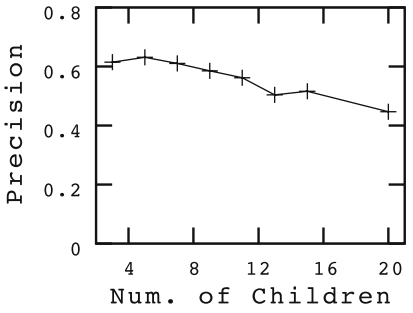

Figure 11 Precision-num. of children curve.

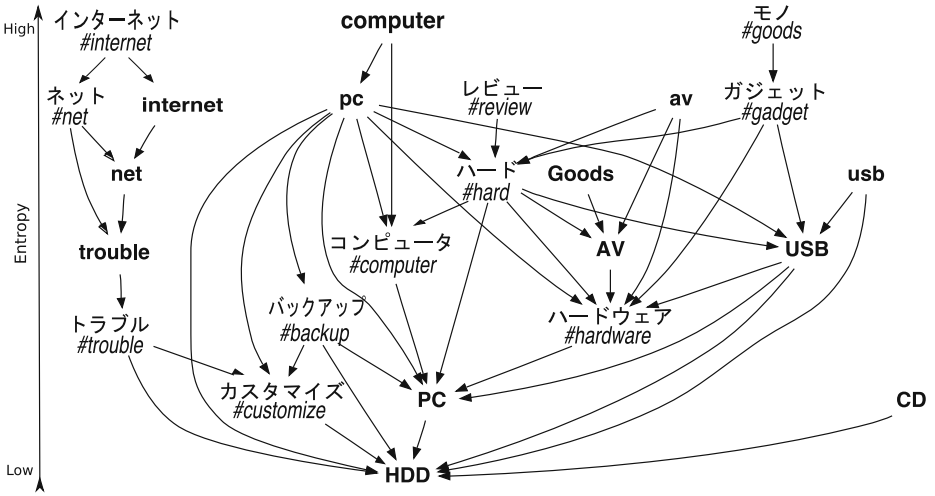

Figure 12 A DAG expanded upward from “HDD”.

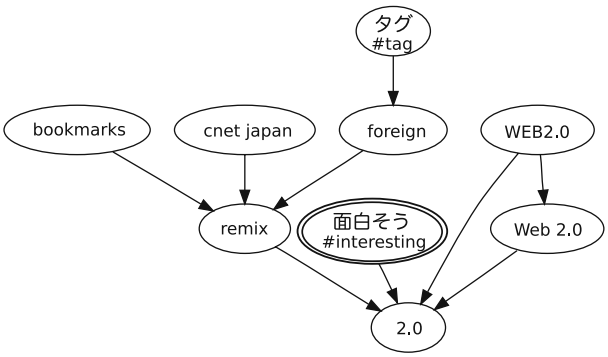

Figure 13 A DAG expanded upward from “2.0” including the nearest subjective tag (J)*interesting*.

not appear in Figure 1 but appear in Figure 6. This property is the main advantage of applying the statistical method for indexing co-occurrence data like folksonomy triples.

Entropy of either vector model showed a positive result in estimating tag subjectivity. The reason why PLSI entropy has better precision seems to be that the dimension-reduced semantic space acquires richer information than the sparse vectors.

DAG has 2.7 times as many child nodes as the **EXT** with 11% less precision when the parameters are statically determined. However, **EXT** cannot expand children locally while **DAG** enables us to expand the number of children with only a small drop in precision. **DAG** can be used in applications such as user navigation tool bars to give users more navigation choices instantly upon demand.

6 Conclusions

We proposed a novel tag organizing method which is composed of the identification of subjective tags and hierarchical allocation into DAG. Based on the hidden semantics identified by PLSI, we discriminate subjective tags from all folksonomy tags. We then form the objective folksonomy tags into a hierarchy by extracting the relationships among them.

Our experiments show that entropy, especially that of PLSI vectors is a good criterion for identifying subjective tags from folksonomy tags. Furthermore, we compared our hierarchy assignment method to the existing tree assignment method based on a sparse folksonomy vector model, where not only the hierarchical structure but also the vector models are evaluated.

As a whole, the statistical method (PLSI) can grasp the hidden semantics underlying the observed bookmarks and can be used as the basis for effectively organizing folksonomy tags.

Future work

To exploit the locally expandable tag hierarchy, threshold adjustment is an interesting future work. Increasing the thresholds means that tag islands are grouped into larger islands. By creating DAGs with several thresholds, we might be able to identify hierarchical clusters that lie above the DAGs. Since folksonomy systems continue to generate new entries, visualization of DAG in a time series is another future work.

Here we estimated tag subjectivity from just entropy value. Therefore, it is a straightforward to refine the task using sophisticated supervised learning techniques [20].

From the viewpoint of user navigation, facet estimation of folksonomy tags using machine learning is a promising avenue. By using estimated facets and hierarchy structure of tags, existing faceted search techniques can be applied [29] in folksonomy navigation systems.

References

1. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst. **30**(1–7), 107–117 (1998)
2. Brooks, C.H., Montanez, N.: Improved annotation of the blogosphere via autotagging and hierarchical clustering. In: Proc. International World Wide Web Conference (WWW), pp. 625–632. ACM, New York (2006)
3. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. **41**(6), 391–407 (1999)
4. Dubinko, M., Kumar, R., Magnani, J., Novak, J., Raghavan, P., Tomkins, A.: Visualizing tags over time. In: Proc. International World Wide Web Conference, pp. 193–202. ACM, New York (2006)
5. Eda, T., Uchiyama, T., Uchiyama, T., Yoshikawa, M.: Signaling emotion in tagclouds. In: Proc. International World Wide Web Conference (WWW), pp. 1199–1200. ACM, New York (2009)
6. Eda, T., Yoshikawa, M., Yamamuro, M.: Locally expandable allocation of folksonomy tags in a directed acyclic graph. In: Proc. Web Information Systems Engineering (WISE), pp. 151–162 (2008)
7. Golder, S., Huberman, B.A.: Usage patterns of collaborative tagging systems. J. Inf. Sci. **32**, 198–208 (2006)
8. Heymann, P., Garcia-Molina, H.: Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical report, Stanford InfoLab. (2006)
9. Heymann, P., Koutrika, G., Garcia-Molina, H.: Can social bookmarking improve web search? In: Proc. International Conference on Web Search and Data Mining (WSDM), pp. 195–206 (2008)
10. Hofmann, T.: Probabilistic latent semantic analysis. In: Proc. Uncertainty in Artificial Intelligence, pp. 289–296, Stockholm (1999)
11. Khy, S., Ishikawa, Y., Kitagawa, H.: A novelty-based clustering method for on-line documents. World Wide Web **11**(1), 1–37 (2008)
12. Kolari, P., Java, A., Finin, T.: Characterizing the splogosphere. In: Proc. Workshop on the Weblogging Ecosystem, Edinburgh (2006)
13. Lee, L.: On the effectiveness of the skew divergence for statistical language analysis. In: Proc. International Workshop on Artificial Intelligence and Statistics, pp. 65–77 (2001)
14. Lin, C., Xue, G.-R., Zeng, H.-J., Yu, Y.: Using probabilistic latent semantic analysis for personalized web search. In: Proc. Asia Pacific Web Conference (APWeb). Lecture Notes in Computer Science, vol. 3399, pp. 707–717. Springer, New York (2005)
15. Lux, M., Granitzer, M., Kern, R.: Aspects of broad folksonomies. In: Proc. International Conference on Database and Expert Systems Applications (DEXA), pp. 283–287 (2007)
16. Martin, H., Keane, M.T.: An assessment of tag presentation techniques. In: Proc. International World Wide Web Conference (WWW), pp. 1313–1314 (2007)
17. Mathes, A.: Folksonomies—cooperative classification and communication through shared metadata. http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html (2004)
18. Mika, P.: Ontologies are us: a unified model of social network and semantics. In: Proc. International Semantic Web Conference (ISWC), pp. 522–536 (2005)
19. Niwa, S., Doi, T., Honiden, S.: Web page recommender system based on folksonomy mining. In: Proc. International Conference on Information Technology: New Generations, pp. 388–393 (2006)
20. Ramamohanarao, K., Fan, H.: Patterns based classifiers. World Wide Web **10**(1), 71–83 (2007)
21. Rivadeneira, A.W., Gruen, D.M., Muller, M.J., Millen, D.R.: Getting our head in the clouds: towards evaluation studies of tagclouds. In: Proc. Conference on Human Factors in Computing Systems (CHI), pp. 995–998 (2007)
22. Sabou, M., Gracia, J., Angeletou, S., dAquin1, M., Motta, E.: Evaluating the semantic web: a task-based approach. In: Proc. International Semantic Web Conference and Asian Semantic Web Conference (ISWC/ASWC). Lecture Notes in Computer Science, vol. 4825/2008, pp. 423–427 (2007)
23. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
24. Sen, S., Lam, S.K., Rashid, A.M., Cosley, D., Frankowski, D., Osterhouse, J., Harper, M.F., Riedl, J.: Tagging, communities, vocabulary, evolution. In: Proc. Conference on Computer Supported Cooperative Work (CSCW), pp. 181–190, New York (2006)

25. Sinclair, J., Cardew-Hall, M.: The folksonomy tag cloud: when is it useful? J. Inf. Sci. **34**, 15–29 (2007)
26. Voss, J.: Tagging, folksonomy & co-renaissance of manual indexing? http://arxiv.org/abs/cs/0701072v2 (2007)
27. Wu, X., Zhang, L., Yu, Y.: Exploring social annotations for the semantic web. In: Proc. International World Wide Web Conference (WWW), pp. 417–426 (2006)
28. Yanbe, Y., Jatowt, A., Nakamura, S., Tanaka, K.: Can social bookmarking enhance search in the web? In: Proc. ACM/IEEE Joint Conference on Digital Libraries (JCDL), pp. 107–116 (2007)
29. Yee, K.-P., Swearingen, K., Li, K., Hearst, M.: Faceted metadata for image search and browsing. In: Conference on Human Factors in Computing Systems (CHI), pp. 401–408. ACM, New York (2003)
30. Yeung, C.-m.A., Gibbins, N., Shadbolt, N.: Tag meaning disambiguation through analysis of tripartite structure of folksonomies. In: Proc. International Conferences on Web Intelligence and Intelligent Agent Technology (WI/IAT), pp. 3–6 (2007)
31. Zhou, D., Bian, J., Zheng, S., Zha, H., Giles, C.L.: Exploring social annotations for information retrieval. In: Proc. International World Wide Web Conference (WWW), pp. 715–724 (2008)