2009 International Conference on Computational Science and Engineering

Tag Sense Disambiguation for Clarifying the Vocabulary of Social Tags

Kangpyo Lee
School of Computer Science & Engineering
Seoul National University
Seoul, South Korea
kplee@idb.snu.ac.kr

Hyunwoo Kim
School of Computer Science & Engineering
Seoul National University
Seoul, South Korea
hwkim@idb.snu.ac.kr

Hyopil Shin
Linguistics Department
Seoul National University
Seoul, South Korea
hpshin@snu.ac.kr

Hyoung-Joo Kim
School of Computer Science & Engineering
Seoul National University
Seoul, South Korea
hjk@snu.ac.kr

Abstract—**Tagging is one of the most popular services in Web 2.0. As a special form of tagging, social tagging is done collaboratively by many users, which forms a so-called folksonomy. As tagging has become widespread on the Web, the tag vocabulary is now very informal, uncontrolled, and personalized. For this reason, many tags are unfamiliar and ambiguous to users so that they fail to understand the meaning of each tag. In this paper, we propose a tag sense disambiguating method, called Tag Sense Disambiguation (TSD), which works in the social tagging environment. TSD can be applied to the vocabulary of social tags, thereby enabling users to understand the meaning of each tag through Wikipedia. To find the correct mappings from del.icio.us tags to Wikipedia articles, we define the Local Neighbor tags, the Global Neighbor tags, and finally the Neighbor tags that would be the useful keywords for disambiguating the sense of each tag based on the tag co-occurrences. The automatically built mappings are reasonable in most cases. The experiment shows that TSD can find the correct mappings with high accuracy.**

Keywords- social tagging; folksonomy; vocabulary; Wikipedia; word sense disambiguation

I. INTRODUCTION

In recent years, tagging has become one of the most popular services in Web 2.0. Websites providing tagging services, such as del.icio.us [1] for bookmarks, Flickr [2] for images, and YouTube [3] for videos, have achieved a great success. Tagging is the act of assigning a series of relevant keywords (i.e. tags) to annotate various resources on the Web. Especially, social tagging (also known as collaborative tagging) is done collaboratively by many users, which forms a so-called folksonomy. Del.icio.us is said to be the true reflection of social tagging and folksonomies. It provides an online social bookmarking service that enables users to register their own bookmarks and share them with others. Each user assigns his or her own tags to a URL of interest, and the whole set of tags (i.e. the folksonomy) created for that URL is shown in the form of a posting history or a tag cloud.

Originally, the tag vocabulary was formal, rather than informal, since it actually was the set of "keywords" that help describe a resource. In [4], the authors gave a good summary on various kinds of del.icio.us tags. However, as tagging has become widespread on the Web, they are now very informal. There is no regulation on tags only if the direction about whitespaces is followed, and users thus can use any words as tags. This informal, uncontrolled, and personalized vocabulary of tags makes general users who see the tags feel uncomfortable since many tags are not familiar to them. In [5], Mathes mentioned the problems inherent in an uncontrolled vocabulary in folksonomies. The problems are the ambiguity, spaces and multiple words, and synonyms. If we use tags as Web metadata to understand the Web resources, these problems can be considerably serious. They can no longer act as the metadata if we cannot understand the meaning of each tag.

Given this situation, if we are able to get the right information about the meaning of each tag, it can be a great help for users to understand the tag and we can get additional benefits. First, we can disambiguate the ambiguous tags. For example, we can tell whether tag '*apple*' is used as a kind of fruit or the Apple Inc. Second, we can understand the meaning of unfamiliar tags such as '*gtd*', '*lifehacks*', or '*ajax*'. (These tags may be familiar to only a few people.) However, none of the existing tag-based Websites provides any information about what each tag means. Here, we can imagine a useful service that relates each tag to the corresponding concept in some external knowledge sources, such as online dictionaries, thesauri, or ontologies. Unfortunately, it is not easy to find the suitable sources because, as aforementioned, the tag vocabulary is too huge to be well referenced by general knowledge sources.

In this paper, we suggest Wikipedia [6] as a good reference to the tag vocabulary. Wikipedia is a Web-based, free-content encyclopedia that gets the unprecedented popularity among internet users. One of the most noticeable features of Wikipedia would be the huge coverage. By March 2009, the number of articles in English is 2,829,195. Currently, it is known to be the largest knowledge repository on the Web and contains much information about the words that are not defined in a dictionary. Examining the Wikipedia makes us realize that the meaning of almost all tags can be clarified in Wikipedia. We figured out how many del.icio.us tags were covered in Wikipedia by naïve exact matching between the tag names and the titles of Wikipedia articles (TABLE 1). The minimum tag frequency means that at least that number of users used the tag. We can see that as the minimum tag frequency increases, the mapping rate also increases. This means that those tags which

978-0-7695-3823-5/09 $26.00 © 2009 IEEE
DOI 10.1109/CSE.2009.454

TABLE 1. THE TAG-TO-WIKIPEDIA MAPPING RATES BY NAÏVE EXACT MATCHING

The Minimum Tag Frequency	The # of Distinct Tags	The # of Mapping Tags	The Mapping Rate
1	57,961	12,606	21.7 %
2	7,055	4,513	64.0 %
3	4,515	3,293	72.9 %
5	3,032	2,439	80.4 %
10	1,941	1,667	85.9 %
50	745	685	91.9 %
100	478	450	94.1 %

are used by few users are not likely to be standard words (i.e. they are close to the noises), which thus turned out to be absent in Wikipedia. On the other hand, those popular tags which are used by many users, say 100 users, show the highest mapping rate of 94.1%. This means that almost all popular tags are being covered by Wikipedia. Of course, it is also true that many people doubt about the quality of information in Wikipedia since the information is not created by experts. It is written by volunteers and edited by anyone. Not all information is of high quality from the beginning. However, after a long process of discussion, it takes on a neutral point of view reached through consensus. As a result, the information of Wikipedia shows unexpectedly higher quality than we can imagine.

In this paper, we propose a novel method, called Tag Sense Disambiguation (TSD), for mapping a del.icio.us tag to the corresponding Wikipedia article and thereby clarifying the vocabulary of tags.

II. RELATED WORK

A. Social Tagging

Not much work has been done on social tagging and folksonomies. Recently, however, this area is drawing attentions from many researchers on the Web and will be of growing importance. The term 'folksonomy' was first proposed in a mailing list [7]. In [5, 8, 9], they gave good general introductions to tagging and folksonomies. In [10], Wal described del.icio.us as broad folksonomies and Flickr as narrow folksonomies. In [4], Golder and Huberman analyzed the structure of the social tagging systems as well as their dynamic aspects. Collective intelligence of Web 2.0 is also a hot issue. In [11], O'Reilly pointed out that the giants who have survived to lead the Web 2.0 era have embraced the power of the Web to harness the collective intelligence. He mentioned as examples Wikipedia and the folksonomies of del.icio.us and Flickr. An interesting and widely accepted usage that exploits this collective intelligence is the co-occurrence-based modeling of folksonomies. This is based on the belief that the frequent co-occurrences of two tags created by many users have a particular meaning, i.e. they are closely related to each other. In [12-18], the authors have proposed their own models for folksonomies, each of which is based on the co-occurrences of tags. And they proved that their co-occurrence-based modeling worked in various applications on the Web. To date, there has been no research on applying the tag co-occurrences to a mapping from a tag to the external knowledge source, thereby clarifying the tag vocabulary.

B. Word Sense Disambiguation

Word Sense Disambiguation (WSD) is the task of choosing the correct sense for a word in a context. WSD has long been a challenging task in computational linguistics. In [19], Navigli gave an excellent survey on WSD. According to him, WSD is considered an AI-complete problem, because it is very hard to know the correct sense of a word on a text in a computational manner. WSD is important in many research areas such as information retrieval, information extraction, machine translation, text classification, content analysis, word processing, lexicography, and the Semantic Web. If we succeed in WSD, we can solve many problems regarding the semantics of words. To date, there has been some research on disambiguating the sense of tags. In [18], the authors proposed a global semantic model to disambiguate tags and group synonymous tags. In [20], the authors proposed a method to disambiguate tags based on the tripartite structure of folksonomies. However, the problem of tag ambiguity has not been addressed very well. This may be because, in order to know the sense of a tag, we should know the context in which the tag occurs, but they find it hard to define the context of each tag. In this paper, we propose a method for disambiguating the sense of a tag, and name it Tag Sense Disambiguation (TSD).

III. TAG SENSE DISAMBIGUATION

A. Overview of TSD

The main goal of TSD is to automatically find a correct one-to-one mapping from a tag to a Wikipedia topic (Fig. 1). Here, by a Wikipedia topic, we mean the title of a Wikipedia article. In other words, the Wikipedia topic represents the concept of a Wikipedia article. TSD is a challenging task since we should know the exact semantics of a tag and find a Wikipedia article that best describes the semantics of the tag. After thorough examination, we observed that assigning tags to a resource is a cognitive process and that a series of tags assigned by a user can be semantically related to each other, although some of them are not. Fig. 2 shows an example of the semantic relatedness among tags. Suppose that a user is going to assign several tags to a resource regarding the JDBC. She is likely to assign '*jdbc*' first, and, next, its related tags such as '*java*', '*database*', and '*programming*'. All these tags are semantically related to each other. In other words, birds of a feather flock together. However, tag '*article*' does not look semantically related to the other tags. There are invisible semantic border lines somewhere in a list of tags according to their semantics. The idea behind TSD is that a tag's neighbor tags can be very useful keywords to clarify the meaning of the tag. In this example, '*java*', '*database*', and '*programming*' would be the useful keywords to clarify the meaning of '*jdbc*'.

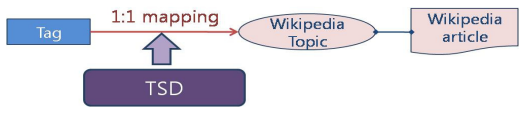

Figure 1. Overview of TSD.

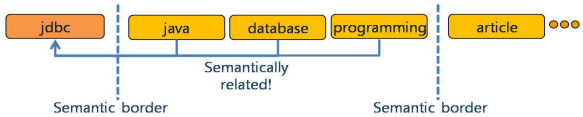

Figure 2. Semantic relatedness among tags.

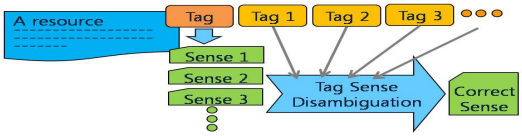

Figure 3. The main idea of TSD

To know the sense of a word, we need a context in which the word is used. Likewise, to know the sense of a tag, we need a context in which the tag is used. The main idea of TSD is that the sense of a tag can be disambiguated by the help of its neighbor tags, which act as a context (Fig. 3). Here, we define the neighbor tags as the tags that co-occur very often with the tag. The rationale behind this co-occurrence-based definition is that the frequent co-occurrences of two tags can be regarded as the high semantic relatedness between them. This approach depends on the collective intelligence hiding in folksonomies.

B. *Local Neighbor Tags*

We observed that users assign some tags to a resource at the same time more often than chance and that it is very likely for the tags to be semantically related to each other within the resource. Given a tag t of interest, the Local Neighbor tags (LN-tags) of t are the tags that co-occur very often with t within a specific resource.

[Example 1] We demonstrate an example of LN-tags that play the role of neighbor tags. Suppose we want to clarify the meaning of del.icio.us tag '*livecd*' that is assigned to resource '*http://www.sysresccd.org/*'. (Live CD is a CD containing a bootable computer operating system, many of which are based on Linux.) From TABLE 2, we can be sure that most LN-tags of '*livecd*', such as '*linux*', '*rescue*' and '*software*', are actually the essential keywords to describe the concept of '*livecd*'. ■

[Definition 1] Given a resource r and a tag t, a set T_{LN} of Local Neighbor tags of t is defined as

$$T_{LN} = \{t_i \mid CoCount(r, t, t_i) \geq 2\}$$

where $CoCount(r, t, t')$ denotes the co-occurrence count of t with t' within r. The threshold value of 2 means that this co-occurrence is not made by chance. ■

C. *Global Neighbor Tags*

The LN-tags themselves can be important keywords to clarify the meaning of a tag. They are useful for TSD since they act as a context within a specific resource to know the sense of a tag. Sometimes, however, some of them happen to co-occur often although they are not semantically related to each other. We call it *accidental co-occurrences*.

TABLE 2. THE CO-OCCURRENCE COUNTS OF '*livecd*' WITH OTHER TAGS

Tag 1	Tag 2	Co-Occurrence Count
livecd	*linux*	623
	rescue	621
	tools	393
	sysadmin	370
	backup	348
	software	247
	opensource	201
	recovery	73
	boot	58
	cd	58

TABLE 3. THE CO-OCCURRENCE COUNTS OF '*apple*' WITH OTHER TAGS IN RESOURCE '*http://fluidapp.com/*'

Tag 1	Tag 2	Co-Occurrence Count
apple	*mac*	271
	browser	259
	software	255
	osx	254
	** web*	172
	tools	161
	** webkit*	105
	apps	48
	application	46
	** web2.0*	39

TABLE 4. THE CO-OCCURRENCE COUNTS OF '*apple*' WITH OTHER TAGS IN THE WHOLE SET OF RESOURCES IN del.icio.us

Tag 1	Tag 2	Co-Occurrence Count
apple	*mac*	3492
	itunes	3235
	software	2995
	osx	2497
	audio	1773
	tools	1757
	ipod	1657
	mp3	1610
	applescript	1595
	scripts	1056

[Example 2] Suppose we want to clarify the meaning of tag '*apple*' that is assigned to resource '*http://fluidapp.com/*'. TABLE 3 and 4 list the frequently co-occurring tags with '*apple*'. The difference between the two tables is that the former covers a specific resource and the latter covers the whole set of resources in del.icio.us. From TABLE 3, we can realize that some LN-tags such as '*web*', '*webkit*', and '*web2.0*' does not look directly related to '*apple*'. This happens because the resource is about a Web application that operates on the Macintosh OSX and it, hence, is also closely related to the Web. In other words, '*apple*' and '*web*' co-occur often not because they are semantically related to each other but because they are accidentally the two main concepts that represent the resource. From TABLE 4, we can see that the Web-related tags disappeared in the whole set of del.icio.us tags. ■

The existence of these accidental co-occurrences leads us to introduce new complementary neighbor tags, or the Global Neighbor tags (GN-tags). Given a tag t of interest, the GN-tags of t are the tags that co-occur very often with t in the whole set of resources.

[Definition 2] Given a tag t, a set T_{GN} of Gocal Neighbor tags of t is defined as

$$T_{GN} = \{t_j \mid CoCount(t, t_j) \geq 2 \text{ and } t_j \text{ is among the top-k\% tags}\}$$

where $CoCount(t, t')$ denotes the co-occurrence count of t with t' in the whole set of resource and k is the threshold value. Empirically, the quality was best when k was 20. ■

D. *Neighbor Tags*

Now, we define the Neighbor tags (N-tags) that incorporate the LN-tags and GN-tags.

[Definition 3] A set T_N of Neighbor tags of tag t is defined as

$$T_N = T_{LN} \cap T_{GN}. \blacksquare$$

In other words, the Neighbor tags of a tag should satisfy the conditions of both the LN-tags and the GN-tags. Here, the LN-

tags act as a context for our TSD while the GN-tags act as a filter that eliminates the problem of accidental co-occurrences.

[Example 3] Given a tag t = *apple* and a resource r = *http://fluidapp.com/*, T_{LN} = {*mac, browser, software, osx, web, tools, webkit, apps, application, web2.0* }, T_{GN} = {*mac, itunes, software, osx, audio, tools, ipod, mp3, applescript, scripts* }. Then, T_N = {*mac, software, osx, tools* }. From the elements in T_N, we can notice that '*apple*' means the Apple Inc., not a kind of fruit. ■

E. The Relevance of Tag to Wikipedia Topic

The Neighbor tags defined in the previous subsection are expected to be the useful keywords that help clarify the meaning of a tag. With these Neighbor tags, we can estimate how relevant a Wikipedia article is to a tag. Given a tag t and a Wikipedia topic *wtp*, let T_N be the set of Neighbor tags of t. The relevance function of t and *wtp* is defined as

$$Relevance(t,wtp) = \sum_{t_i \in T_N} TF(t_i, Article(wtp)) * w(t_i)$$

where *Article(wtp)* denotes the text of Wikipedia article for *wtp*, *TF(word, text)* denotes the term frequency of *word* on *text*, and $w(t_i)$ denotes the weight of t_i. The relevance is based on the term frequency of Neighbor tags on a Wikipedia text. This means that the more frequently the Neighbor tags appear on the text, the more relevant the text is to the tag. We multiply the weight of each Neighbor tag that is assigned according to its co-occurrence count.

F. Finding a Mapping from Tag to Wikipedia Topic

Now we are ready to find the mapping from a tag to a Wikipedia article. Before we proceed, we need two assumptions for our TSD.

[Assumption 1] Wikipedia contains at least one article that corresponds to a tag. ■

[Assumption 2] The article has enough information to express the various semantics of the tag. ■

Assumption 1 is needed because the goal of TSD is to find the correct Wikipedia article that corresponds to a tag of interest. Assumption 2 is for applying the relevance metric defined in the previous subsection. Based on these assumptions, we demonstrate how TSD can be applied to find a mapping from a tag to a Wikipedia article.

According to [19], one of the traditional WSDs can be formalized as

$$S = \underset{S_i \in Senses_D(w)}{argmax}\ score(S_i)$$

where w denotes the word of interest, $Senses_D(w)$ denotes the set of senses encoded in a dictionary D for w, $score(S_i)$ denotes the predefined function, and S denotes the correct sense we want to know. The sense with the highest score is selected as the correct sense. This original WSD can be applied to our TSD. Here, the senses in WSD correspond to the Wikipedia topics in TSD, and the score function in WSD corresponds to the relevance function in TSD. Given a tag t, the mapping M produces a Wikipedia topic by the following equation

$$M(t) = \underset{topic_i \in Topics_{Wiki}(t)}{argmax}\ Relevance(t, topic_i)$$

where $Topics_{Wiki}(t)$ denotes the set of Wikipedia topics that match with t. We call these matching topics *candidate topics*. We can find the candidate topics by exact/partial matching between the tag name and the topic names. The correct Wikipedia topic that corresponds to a tag is the candidate topic with the highest relevance value. If the mapping M produces a topic for a tag, a mapping from the tag to the Wikipedia topic is built.

[Example 4] Given a tag t = *editor* and a resource r = *http://www.fckeditor.net/*, suppose we want to find the Wikipedia topic that corresponds to '*editor*'. Wikipedia has 4 candidate topics that match with t. (In fact, it has 29 candidate topics.) They are *Editor_in_chief*, *HTML_editor*, *Text_editor*, and *WYSIWYG*. The set of Neighbor tags of t is T_N = {*wysiwyg, javascript, html, opensource, ajax, web, webdesign, software, development, tools, text, browser, programming, web2.0, code, online, tool, freeware, blog, application* }. The one with the highest relevance value is, as we expect, the *Relevance (editor, HTML_ editor)*. Now we can conclude that the correct Wikipedia topic that corresponds to '*editor*' is '*HTML_editor*'. ■

Note the case in which Wikipedia has no topic that matches with the tag or all relevance values are zero. In this case, we may conclude that Wikipedia does not cover the concepts regarding the tag. However, we can think of a useful heuristics that can be applied as a last means. The heuristics is to find the *identical tags* among its Neighbor tags.

[Definition 4] Given a tag t of interest, the identical tags are defined as those tags which share a common stem within a specific resource. ■

This definition makes sense in that, within a specific resource, those tags which share a common stem can be regarded as the same, or almost the same, tags. In fact, the tag vocabulary contains a lot of identical tags. For example, tags '*blog', 'blogs',* and *'blogging'* can be treated as identical and tags '*util', 'utils', 'utility',* and *'utilities'* can also be treated as identical. The existence of identical tags makes us need the stemming techniques in IR. Stemming is the process of collapsing together the morphological variants of a word [21]. One of the most widely used stemming algorithms is the Porter Stemming Algorithm [22], which has been known to be simple and efficient. In our tag vocabulary, the Porter Stemming Algorithm is suitable to find the identical tags. The idea is that if a tag has no mapping Wikipedia topic, a new mapping is built to the topic that corresponds to one of its identical tags. This process has to be done after all the other normal tags have found their own mappings. Given a tag t, a new mapping M' is defined as

$$M'(t) = \underset{topic_i \in IdTopics_{Wiki}(t)}{argmax}\ Relevance(t, topic_i)$$

where $IdTopics_{Wiki}(t)$ denotes the set of Wikipedia topics that match with the identical tags of t. The Wikipedia topic that corresponds to t is the topic with the highest relevance value. Of course, there exist exceptions that we cannot treat those stem-sharing tags as identical tags. For example, tags '*community'* and '*communication'* share a common stem '*commun'*, but they are not identical. However, we believe that these exceptions can be ignored since 1) those cases are rare and 2) it is also unlikely that one of those two tags, say '*community'* or '*communication'*, has no corresponding Wikipedia topic.

Last, TSD benefits from two useful services of Wikipedia: redirections and disambiguation pages. Through redirections, Wikipedia sends the reader to an article, usually from an alternative title. The disambiguation pages are used to disambiguate a number of similar terms. These two services are of great help by extending the candidate topics that otherwise could only be found by naïve exact matching between the tag name and the topic names.

IV. IMPLEMENATION AND ANALYSIS

We downloaded the English Wikipedia dataset that was distributed in March 2009. The number of articles is about 8,251,357. We collected the del.icio.us popular tag dataset in November 2008. The number of URLs is 1038, the number of users is 178232, and the number of distinct tags is 57961. We made a simple UI that can be added on a Web browser. When the mouse is over a tag, it shows the name of the corresponding Wikipedia topic and provides a hyperlink to the corresponding Wikipedia article.

TABLE 5 illustrates what mappings TSD have found in resource '*http://fluidapp.com/*'. We can see that TSD produces good results since most mappings are reasonable. There are several points to mention. First, tag '*apple*' is mapped to Wikipedia topic '*Apple_inc*', not to a kind of fruit. This means that our TSD succeeds in disambiguating the sense of tags. The mapping from tag '*safari*' to Wikipedia topic '*Safari_(web_browser)*' is also a good example. Second, both tags '*application*' and '*applications*' are mapped to the same Wikipedia topic '*application_software*'. This mapping is correct because they are identical tags. In fact, this mapping is possible not by TSD but by the redirection of Wikipedia. Third, tag '*leopard*' is mapped to Wikipedia topic '*Mac_OS_X_v10.5'*. This is very interesting because the Leopard is the nick name of the Mac OS X version 10.5. This is also possible by the redirection. Last, we found an weird mapping from tag '*tool*' to Wikipedia topic '*Tool_(band)*'. Tool is another name of an American rock band. The '*tool*' should have been mapped to Wikipedia topic '*Programming_tool*'. The reason for this wrong mapping is that the Wikipedia article for '*Tool_(band)*' accidentally contains a lot of matching words such as '*web', 'programming', 'download',* and *'internet'*, thereby increasing the term frequency and relevance. We call this problem *accidental high relevance*. This shows the limitation of TSD.

V. EVALUATION

The goal of our experiments is to know how well TSD works in the social tagging environment to find correct mappings from tags to Wikipedia articles. Unfortunately, the evaluation is difficult since 1) we have neither the correct answer set about tag-to-Wikipedia mappings nor the domain experts who knows everything about the tag vocabulary and Wikipedia articles, 2) some mappings are hard to judge whether or not they are correct, and 3) there exist cases in which Wikipedia contains no article that corresponds to a tag. For these reasons, it is impossible to conduct a quantitative analysis. One possible way is the qualitative analysis by manual evaluation. For the experiments, a group of 15 Ph.D. students were chosen as subjects. They were majoring in computer science, had a large tag vocabulary, and were accustomed to using Wikipedia. In other words, they were assumed to be the domain experts.

TABLE 5. SAMPLE MAPPINGS FROM TAGS TO WIKIPEDIA TOPICS

Del.icio.us Tag	Wikipedia Topic	Del.icio.us Tag	Wikipedia Topic
* apple	Apple_Inc.	mac	Macintosh
* application	application_software	macosx	Mac_OS_X
* applications	application_software	osx	Mac_OS_X
browser	Web_browser	productivity	Productivity
cool	Cool_(aesthetic)	* safari	Safari_(web_browser)
desktop	Desktop_environment	software	Computer_software
development	Software_development	ssb	SSB
fluid	Fluid_(browser)	* tool	Tool_(band)
freeware	Freeware	web	World_Wide_Web
gmail	Gmail	webapp	Web_application
internet	Internet	webkit	WebKit
* leopard	Mac_OS_X_v10.5		

A. Precision

The goal of the first experiment is to figure out how correct the automatically built mappings from tags to Wikipedia articles are. The top-10 popular URLs and their tags were chosen from del.icio.us data set. From each of the 10 URLs, 10 mappings were randomly chosen, i.e. total 100 mappings were provided to each subject. For each mapping, subjects were given a tag and its mapping Wikipedia topic. A hyperlink from the Wikipedia topic to the real Wikipedia article was also provided to help the subjects judge whether or not the article was actually closely related to the tag. Additionally, the basic information about the tag and the URL were also provided. This information could help the subjects understand what the tag means. For each mapping, subjects were asked to judge whether the mapping looks a) Correct, b) Not correct, but related, c) Neither correct nor related, or d) I don't know.

TABLE 6 shows the results. The high proportion of "Correct" (80.2 %) and the low proportion of "Neither correct nor related" (7.8 %) are promising. In case of "Correct", almost all mapping are reasonable and actually correct. Interestingly, some mappings are even excellent. For example, tag '*education*' is mapped to Wikipedia topic '*Educational_technology*', not to '*Education*'. After we examined the context in which the tag '*education*' was used, the meaning of '*education*' turned out to be closer to '*Educational_technology*' than to general '*Education*'. This is a good example proving that the Neighbor tags actually work as a context for a tag. In case of "Neither correct nor related", most of the wrong mappings were caused by the accidental high relevance we have indicated in the previous section. For example, tag '*cms*' (Content Management System) is mapped to Wikipedia topic '*CMS-2_(programming _language)*', and tag '*clone*' is mapped to Wikipedia topic '*Video_game_clone*'. The proportion of "Not correct, but related" (10.5 %) seems no problem, but it reveals some limitations of TSD. "Not correct, but related" means that TSD should have found more correct mappings than it did. For example, tag

TABLE 6. RESULTS FOR ANSWERING TO THE QUESTIONS (%)

URL#	Correct	Not Correct, But Related	Neither Correct Nor related	I Don't Know
54	95.6	2.2	2.2	0
149	84.4	15.6	0	0
22	77.8	13.3	6.7	2.2
15	82.2	8.9	6.7	2.2
131	75.6	6.7	13.3	4.4
91	62.2	22.2	15.6	0
477	86.7	6.7	4.4	2.2
105	86.7	8.9	2.2	2.2
104	71.1	8.9	20.0	0
48	79.5	11.4	6.8	2.3
Avg.	**80.2**	**10.5**	**7.8**	**1.6**

'*resource*' should have been mapped to Wikipedia topic '*Resource_(Web)*', not to general '*Resource*'. Last, a few subjects answered with "I don't know" (1.6 %), mainly because some mappings were hard to judge whether or not it was correct.

B. Recall

We want to know whether Wikipedia contains actually no article to which TSD produced no mapping from a tag. That is, if TSD declares that Wikipedia does not contain any article that best describes a tag, we want to validate the declaration. Fortunately, the nonmapping tags produced from our experiments are so rare that we can list all of them here. The number of nonmapping tags was only 9. Above all, some tags such as '*2.0*', '*commoncraft*', '*exploratree*', and '*vetor*' are actually not being covered in Wikipedia. The '*commoncraft*' and '*exploratree*' are the names of internet sites. The '*vetor*' is thought to be the misspelling of '*vector*'. On the other hand, the other tags such as '*apps*', '*applescripts*', '*downloads*', '*resources*', and '*webapps*' turned out to be actually being covered in Wikipedia after our thorough examinations. The '*apps*' and '*webapps*' are the abbreviations for application and Web application, respectively. TSD is unable to handle these kinds of abbreviations unless Wikipedia redirects these names to their full names. The '*applescripts*', '*downloads*', and '*resources*' are the plurals for applescript, download, and resource, respectively. In case of '*applescripts*', the redirection of Wikipedia was wrong. In case of '*downloads*', and '*resources*', TSD failed to find the correct mappings. In summary, the number of nonmapping tags was very small (= 9), most of which were caused not by TSD but by the absence of appropriate information in Wikipedia. This means that the recall of TSD is very high.

VI. CONCLUSION

Many tags on the Web are unfamiliar and ambiguous to users. Unfortunately, there is no way to understand the meaning of each tag. In this paper, we presented a tag sense disambiguating method called TSD which works in the social tagging environment. We fully exploited the collective intelligence of Web 2.0 in defining the Neighbor tags by using the tag co-occurrences. We showed that TSD can be applied to the vocabulary of social tags, thereby clarifying the tag vocabulary through Wikipedia. We believe that this work will be a great help for users who try to see the folksonomy as Web metadata.

ACKNOWLEDGMENT

This research was supported by the Brain Korea 21 Project and the Ministry of Knowledge Economy, Korea, under the Information Technology Research Center support program supervised by the Institute of Information Technology Advancement (grant number IITA-2008-C1090-0801-0031).

REFERENCES

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

[2] Flickr, http://www.flickr.com.

[3] YouTube, http://www.youtube.com.

[4] Scott Golder and Bernardo A. Huberman, "Usage Patterns of Collaborative Tagging Systems," *Journal of Information Science*, vol. 32, pp. 198-208, 2006.

[5] Adam Mathes, "Folksonomies - Cooperative Classification and Communication Through Shared Metadata," *Computer Mediated Communication*, Graduate School of Library and Information Science University of Illinois Urbana-Champaign, December 2004.

[6] Wikipedia, http://wikipedia.org.

[7] Gene Smith, "Folksonomy: social classification," http://atomiq.org/archives/2004/08/folksonomy_social_classification.html, August, 2004.

[8] Emanuele Quintarelli, "Folksonomies: power to the people," http://www.iskoi.org/doc/folksonomies.htm, June 2005.

[9] Clay Shirky, "Ontology is Overrated: Categories, Links, and Tags," http://www.shirky.com/writings/ontology_overrated.html, 2005.

[10] Thomas Vander Wal, "Explaining and Showing Broad and Narrow Folksonomies," http://personalinfocloud.com/2005/02/explaining_and_.html, February 2005.

[11] Tim O'Reilly, "What Is Web 2.0," http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html, 2005.

[12] Paul Alexandru Chirita, Stefania Costache, Siegfried Handschuh, and Wolfgang Nejdl, "P-TAG: Large Scale Automatic Generation of Personalized Annotation TAGs for the Web," In: *Proc. of the 16th international conference on World Wide Web (WWW2007)*, pp. 845-854, May 2007.

[13] Harry Halpin, Valentin Robu, and Hana Shepherd, "The Complex Dynamics of Collaborative Tagging," In: *Proc. of the 16th international conference on World Wide Web (WWW2007)*, pp. 211-220, May 2007.

[14] Rui Li, Shenghua Bao, Ben Fei, Zhong Su, and Yong Yu, "Towards Effective Browsing of Large Scale Social Annotations," In: *Proc. of the 16th international conference on World Wide Web (WWW2007)*, pp. 943-952, May 2007.

[15] Xin Li, Lei Guo, and Yihong Eric Zhao, "Tag-based Social Interest Discovery," In: *Proc. of the 17th international conference on World Wide Web (WWW2008)*, pp. 675-684, April 2008.

[16] Peter Mika, "Ontologies are us: A unified model of social networks and semantics," In: *Proc. of the 4th International Semantic Web Conference (ISWC2005)*, pp. 522-536, November 2005.

[17] Börkur Sigurbjörnsson and Roelof van Zwol, "Flickr Tag Recommendation based on Collective Knowledge," In: *Proc. of the 17th international conference on World Wide Web (WWW2008)*, pp. 327-336, April 2008.

[18] Xian Wu, Lei Zhang, and Yong Yu, "Exploring Social Annotations for the Semantic Web," In: *Proc. of the 15th international conference on World Wide Web (WWW2006)*, pp. 417-426, May 2006.

[19] Roberto Navigli, "Word sense disambiguation: A survey," *ACM Computing Surveys*, vol. 41, pp. 1-69, February 2009.

[20] Ching-man Au Yeung, Nicholas Gibbins, and Nigel Shadbolt, "Tag Meaning Disambiguation through Analysis of Tripartite Structure of Folksonomies," in *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Workshops*: IEEE Computer Society, 2007.

[21] Daniel Jurafsky and James H. Martin, "Chapter 23. Question Answering and Summarization," in *Speech and Language Processing, Second Edition*: Prentice Hall, 2008.

[22] M. F. Porter, "An algorithm for suffix stripping," in *Morgan Kaufmann Multimedia Information And Systems Series*: Morgan Kaufmann Publishers Inc., 1997.