Explore the Structure of Social Tags by Subsumption Relations

Xiance Si, Zhiyuan Liu, Maosong Sun
Department of Computer Science and Technology
State Key Lab on Intelligent Technology and Systems
National Lab for Information Science and Technology
Tsinghua University
{sixiance,lzy.thu}@gmail.com, sms@tsinghua.edu.cn

Abstract

Thanks to its simplicity, social tagging system has accumulated huge amount of user contributed tags. However, user contributed tags lack explicit hierarchical structure, while many tag-based applications would benefit if such a structure presents. In this work, we explore the structure of tags with a directed and easy-to-evaluate relation, named as the subsumption relation. We propose three methods to discover the subsumption relation between tags. Specifically, the tagged document's content is used to find the relations, which leads to better result. Besides relation discovery, we also propose a greedy algorithm to eliminate the redundant relations by constructing a Layered Directed Acyclic Graph (Layered-DAG) of tags. We perform quantitative evaluations on two real world data sets. The results show that our methods outperform hierarchical clustering-based approach. Empirical study of the constructed Layered-DAG and error analysis are also provided.

1 Introduction

In this work, we aim at exploring the structure of social tags. Social tagging is widely used in Web-based services, in which a user could use any word to annotate an object. Thanks to its simplicity, services with social tagging features have attracted a lot of users and have accumulated huge amount of annotations. However, comparing to taxonomies, social tagging has an inherent shortcoming, that there is no explicit hierarchical relations between tags. Figure 1 (a) shows an example of the commonly used flat tag cloud, in which only the popularity of a tag is concerned. Kome et al. (2005) argued that implicit hierarchical relations exist in social tags. Previous literature shows that organizing tags in hierarchical structures will help tag-based Information Retrieval applications (Begelman et al., 2006; Brooks and Montanez, 2006).

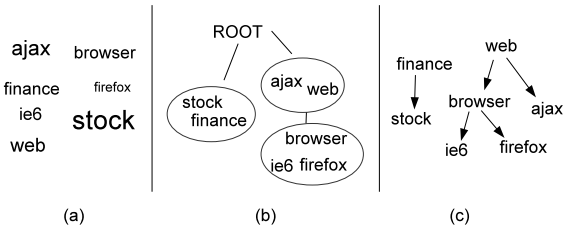


Figure 1: Examples of (a) flat tag cloud, (b) hierarchical clusters, and (c) subsumption relations.

Hierarchical clustering could reveal the similarity relations of tags. Figure 1 (b) shows an example of a typical hierarchical clustering of tags. While clusters can capture similarity between tags, problems still remain: First, clusters mix different relations, such as synonyms and hypernyms. Second, clusters also ignore the direction of relations, for example, the direction in `browser` $\rightarrow$ `firefox`. Third, it is hard to evaluate the correctness of clustering. Specifically, it is hard to tell if two tags are similar or not. In practice, directed and easy-to-evaluate relations between tags are preferred, such as Figure 1 (c).

In this work, we explore the structure of social tags by discovering a directed and easy-to-evaluate relation between tags, named *subsumption relation*. A tag t_a subsumes t_b, if and only if wherever t_b is used, we can also replace it

with t_a. Unlike *similar-to*, subsumption relation is asymmetric, and its correctness is easier to assess. Then, we propose three ways to discover the subsumption relations, through tag-tag, tag-word and tag-reason co-occurrences respectively. In the third way, A tag's *reason* is defined as the word in the content that explains the using of the tag. We employ the Tag Allocation Model (TAM) proposed by Si et al. (2010) to find the reason for each tag. Besides subsumption relation discovery, we also propose a greedy algorithm to remove the redundant relations. The removal is done by constructing a Layered Directed Acyclic Graph (Layered-DAG) of tags with the subsumption relations.

We carried out the experiments on two real world data sets. The results of quantitative evaluation showed that tag-reason based approach outperformed other two methods and a commonly used hierarchical clustering-based method. We also do empirical study on the output of Layered-DAG construction.

The contribution of this paper can be summarized as follows:

1. We explore the structure of social tags by a clearly defined subsumption relation. We propose methods to discover the subsumption relation automatically, leveraging both the co-occurred tags and the content of annotated document.
2. We propose an algorithm to eliminate the redundant relations by constructing a Layered-DAG of tags.
3. We perform both empirical and quantitative evaluation of proposed methods on two real world data sets.

The rest of the paper is organized as follows: Section 2 surveys the related work; Section 3 defines the subsumption relation we used, and proposes methods for relation discovery; Section 4 proposes a greedy algorithm for Layered-DAG construction; Section 5 explains the experimental settings and shows the evaluation results. Section 6 concludes the paper.

2 Related Work

To explore the hierarchical relations between tags, an intuitive way is to cluster the tags into hierarchical clusters. Wu et al. (2006b) used a factorized model, namely Latent Semantic Analysis, to group tags into non-hierarchical topics for better recommendation. Brooks et al. (2006) argued that performing Hierarchical Agglomerative Clustering (HAC) on tags can improve the collaborative tagging system. Later, HAC on tags was also used for improving personalized recommendation (Shepitsen et al., 2008). Heymann et al. (2006) clustered tags into a tree by a similarity-based greedy tree-growing method. They evaluated the obtained trees empirically, and reported that the method is simple yet powerful for organizing tags with hierarchies. Based on Heymann et al.'s work, Schwarzkopf et al. (2007) proposed an approach for modeling users with the hierarchy of tags. Begelman et al. (2006) used top-down hierarchical clustering, instead of bottom-up HAC, to organize tags, and argued that tag hierarchies improve user experiences in their system. Most of the hierarchical clustering algorithms rely on the symmetric similarity between tags, while the discovered relations are hard to evaluate quantitatively, since one cannot distinguish similar from not-similar with a clear boundary.

People have also worked on bridging social tagging systems and ontologies. An ontology defines relations between entities. Peter Mika (2005) proposed an extended scheme of social tagging that includes actors, concepts and objects, and used tag co-occurrences to construct an ontology from social tags. Wu et al. (2006a) used hierarchical clustering to build ontology from tags that also use similar-to relations. Later, ontology schemes that fits social tagging system were proposed, such as (Van Damme et al., 2007) and (Echarte et al., 2007), which mainly focused on the relation between tags, objects and users, rather than between tags themselves. Alexandre Passant (2007) mapped tags to domain ontologies manually to improve information retrieval in social media. To construct tag ontology automatically, Angeletou et al. (2007) used ontologies built by domain experts to find relations between tags, but observed a very low coverage. Specia et al. (2007) proposed an integrated framework for organizing tags by existing ontologies, but no experiment was performed. Kim et al. (2008) summarized the state-

of-the-art methods to model tags with semantic annotations.

Before social tagging was invented, Sanderson et al. (1999) proposed to use *subsumption* relation to organize words in text hierarchically. Schmitz et al. (2006) followed the idea to use subsumption relation for organizing Flickr [1] tag, where tag-tag co-occurrences are used for discover the relations. We follow the idea of subsumption relation in this paper, and explore alternative ways for relation discovery.

3 Subsumption Relations in Tags

In this section, we define the subsumption relation used in our study, and propose three methods to discover the subsumption relations.

3.1 Definitions

First, we introduce the symbols used through out the paper: A tag is denoted as $t \in T$, where T is the set of all tags. To distinguish from words, we use `fixed-width` to represent the example tags. An annotated document is denoted as $d \in D$, where D is the set of all documents. The words in d are denoted as a set $\{w_{d_i}\}$, where $i \in [1, |d|]$, and $|d|$ is the number of words in d.

Inspired by (Sanderson and Croft, 1999), we define the subsumption relation between t_a and t_b as follows: t_a **subsumes** t_b**, means that wherever the tag** t_b **is used,** t_a **can also be used without ambiguity**. The subsumption relation between t_a and t_b is denoted as $t_a \rightarrow_s t_b$.

Subsumption relation is directional, that is, $t_a \rightarrow_s t_b$ does not imply $t_b \rightarrow_s t_a$. For example, `literature` $\rightarrow_s$ `chineseliterature`, since for any document annotated with `chineseliterature`, we can also annotate it with `literature`. However, if we swapped the two tags, the statement would not hold.

Subsumption relation is more strict than similarity. For example, during the time of Haiti earthquake, the tag `earthquake` is close to `haiti` in similarity, but none of them implies the use of the other one: document annotated with `earthquake` may refer to the earthquake in China, while document annotated with `haiti` may mean the traveling experience in Haiti.

Note that the subsumption has transitivity property, that $t_a \rightarrow_s t_b$ and $t_b \rightarrow_s t_c$ means $t_a \rightarrow_s t_c$, which corresponds to our intuition. For instance, `naturaldisaster` $\rightarrow_s$ `earthquake` and `disaster` $\rightarrow_s$ `naturaldisaster` means `disaster` $\rightarrow_s$ `earthquake`.

3.2 Discover Subsumption Relation

We discover the subsumption relations by estimating the probability $p(t_a|t_b)$. The motivation is, if $t_a \rightarrow_s t_b$ and t_b is used, it would be more likely to see t_a. So, by sorting all (t_a, t_b) pairs by $p(t_a|t_b)$ in descending order, top-ranked pairs are more likely to have subsumption relations.

In this work, we present three methods to estimate the probability $p(t_a|t_b)$, using tag-tag, tag-word and tag-reason co-occurrences respectively. By using tag-word and tag-reason co-occurrences, we leverage the content of the annotated document for subsumption relation discovery.

3.2.1 Tag-Tag Co-occurrences Approach

The most intuitive way to estimate $p(t_a|t_b)$ is via tag-tag co-occurrences. Specifically, we use the following formula:

$$p(t_a|t_b) = \frac{N_d(t_a, t_b)}{N_d(t_b)}, \quad (1)$$

where $N_d(t_a, t_b)$ is the number of documents that are annotated by both t_a and t_b, and $N_d(t_b)$ is the number of documents annotated by t_b. We denote the tag-tag co-occurrences approach as TAG-TAG.

The use of TAG-TAG can be found in previous literature for organizing tags for photos(Schmitz, 2006). One of TAG-TAG's benefits is that it does not rely on the content of the annotated document, thus it can be applied to tags for non-text objects, such as images and music. However, when coming to text documents, this benefit is also a shortcoming, that TAG-TAG makes no use of the content when it is available.

Using TAG-TAG for subsumption relation discovery relies on an implication, that if a user has annotated d with t_b, he would also annotate all tags that subsumes t_b. The implication may not always hold in real world situations. For example,

[1] http://www.flickr.com. An image sharing site that allows users to annotate images with tags

a novel reader would use tags such as `scifi` and `mystery` to organize his collections, but he is not likely to annotate each of his collection as `novel` or `book`, since they are too obvious for him. We name the problem as the *omitted-tag problem*.

3.2.2 Tag-Word Co-occurrences Approach

When the content of the annotated document is available, using it for estimating $p(t_a|t_b)$ is a natural thought. The content is expected to be complete and information-rich whether or not the user has omitted any tags. We use the following formula to estimate $p(t_a|t_b)$ by tag-word co-occurrences:

$$\begin{aligned} p(t_a|t_b) &= \sum_{w \in W} p(t_a|w)p(w|t_b) \\ &= \sum_{w \in W} \frac{N_d(t_a, w)}{N_d(w)} \frac{N_d(t_b, w)}{N_d(t_b)}, \end{aligned} \quad (2)$$

where $N_d(t_a, w)$ is the number of documents that contains both tag t_a and word w, and $N_d(w)$ is the number of documents that contains the word w. We denote this approach as TAG-WORD.

Instead of computing tag-tag co-occurrences directly, TAG-WORD uses words in the document as a bridge to estimate $p(t_a|t_b)$. By introducing words, the estimation is less affected by the omitted-tag problem, Take the novel reader example again: Although he does not use the tag `novel` too often, the words in book descriptions would suggest the using of `novel`, according to all other documents annotated by `novel`.

While using the content may weaken the omitted-tag problem, it also brings the noise in text to the estimation. Not every word in the content is related to one of the tags. To the opposite, most words are functional words or that about other aspects of the document. $p(t_a|t_b)$ estimated by using all words may largely depends on these irrelevant words.

3.2.3 Tag-Reason Co-occurrences Approach

To focus on the words that are highly relevant to the interested tags, we propose the third method that uses tag-reason co-occurrences. The *reason* is defined as the word(s) that can explain the using of a tag in the document. For example, the tag `scifi` for a book could be explained by the words "robot", "Asimov" in the book description. If the reason of each tag could be identified, the noise in content-based $p(t_a|t_b)$ could be reduced.

Si et al. (2010) proposed a probabilistic model for content-based social tags, named Tag Allocation Model (TAM). TAM introduces a latent variable r for each tag in the data set, known as the reason variable. The value of r can be a word in the corresponding document, or a global noise variable μ. Allowing the reason of tags to be a global noise makes TAM deal with content-irrelevant tags and mistakenly annotated tags effectively. The likelihood that a document d is annotated by tag t is given as:

$$\begin{aligned} p(t|d) &= \sum_{w \in d} p(t|r = w)p(r = w|d)p(s = 0) \\ &+ p(t|\mu)p(r = \mu)p(s = 1), \end{aligned} \quad (3)$$

where r is the reason of the tag t, $r \in \{w_{di}|i \in [0, |d|]\} \cup \{\mu\}$, μ is the global noise variable. s is the source of reason t, $s = 0$ means the source is the content of the document, while $s = 1$ means the source is the global noise variable μ. TAM can be trained use Gibbs sampling method. For the details of TAM, please refer to (Si and Sun, 2010).

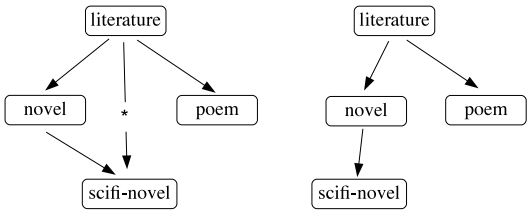
With a trained TAM, we can infer $p(t|r)$, the probability of seeing a tag t when using r as the reason, and $p(r|t)$, the probability of choosing r as the reason for tag t. With these probabilities, we can estimate $p(t_a|t_b)$ by

$$p(t_a|t_b) = \sum_{r \in W} p(t_a|r)p(r|t_b). \quad (4)$$

Note that we use only word reasons ($r \in W$), ignoring the noise reason μ completely. We denote this approach as TAG-REASON.

With the help of TAM, TAG-REASON covers the problems of the TAG-WORD method in two aspects: First, instead of using all words, TAG-REASON emphasizes on the really relevant words, which are the reasons identified by TAM. Second, by ignoring the noise variable μ, TAG-REASON is less affected by the content-irrelevant noise tags, such as `thingstodo` or `myown`.

After $p(t_a|t_b)$ is estimated for each $(t_a, t_b) \in T \times T$, we use the top-$n$ pairs with largest $p(t_a|t_b)$



(a) Directed Acyclic Graph (b) Layered Directed Acyclic Graph

Figure 2: DAG and Layered-DAG

as the final set of discovered subsumption relations.

4 Remove Redundancy with Layered-DAG Construction

The discovered subsumption relations connect all tags into a directed graph $G = \{V, E\}$, where V is the set of nodes, with each node is a tag; E is the set of edges, an edge e_{t_a,t_b} from t_a to t_b means $t_a \rightarrow_s t_b$. Furthermore, we define the weight of each edge w_e as the probability $p(t_a|t_b)$.

Recalling that subsumption relation has transitivity property, to avoid the cyclic references in G, we would like to turn G into a Directed Acyclic Graph (DAG). Further, DAG may also contains redundant information. Figure 2 (a) shows a part of a DAG. Note the edge marked as "*", which is perfectly correct, but does not provide extra information, since `literature` $\rightarrow_s$ `novel` and `novel` $\rightarrow_s$ `scifi-novel` have already implied that `literature` $\rightarrow_s$ `novel`. We would like to remove these redundant relations, turning a DAG into the form of Figure 2 (b).

We define Layered-DAG formally as follows: For a DAG G, when given any pair of nodes, if every path that can connect them has equal length, G is a Layered-DAG. Layered-DAG prohibits edges that link cross layers, such like edge "*" in Figure 2 (a). Constructing a Layered-DAG from the discovered relations can eliminate the redundant information.

Given a set of subsumption relations, multiple Layered-DAGs may be constructed. In particular, we want to find the Layered-DAG that maximizes the sum of all edges' weights. Weight maximization implies two concerns: First, when we need to remove a relation to resolve the conflicts or redundancy, the one with lower weight is prefered.

Layered-DAG Construction Algorithm

Input: A set of weighted relations, $R = \{t_a \rightarrow_s t_b | t_a \in T, t_b \in T\}$, $w_{t_a \rightarrow_s t_b} > 0$
Output: A Layered-DAG of tags $G^* = \{V^*, E^*\}$

1: $V^* = \{\}$
2: **while** $R \neq \varnothing$
3: **if** $V^* = \varnothing$
4: choose $t_a \rightarrow_s t_b \in R$ with highest weight.
5: $E^* \Leftarrow t_a \rightarrow_s t_b$
6: $V^* \Leftarrow t_a, V^* \leftarrow t_b$.
7: remove $t_a \rightarrow_s t_b$ from R.
8: **else**
9: $C \Leftarrow \{t_a \rightarrow_s t_b | t_a \rightarrow_s t_b \in R, \{t_a, t_b\} \cap V^* \neq \varnothing\}$
10: **for** $t_a \rightarrow_s t_b \in C$ in descending weight order
11: **if** adding $t_a \rightarrow_s t_b$ to G^* keeps G^* a Layered-DAG.
12: $E^* \Leftarrow t_a \rightarrow_s t_b$
13: $V^* \Leftarrow t_a, V^* \leftarrow t_b$.
14: **break**
15: **endif**
16: remove $t_a \rightarrow_s t_b$ from R.
17: **endfor**
18: **endif**
19: **endwhile**
20: **output** G^*

Figure 3: A greedy algorithm for constructing Layered-DAG of tags

Second, when more than one valid Layered-DAGs are available, we want to use the one that contains as many edges as possible.

Finding and proving an optimal algorithm for maximum Layered-DAG construction are beyond the scope of this paper. Here we present a greedy algorithm that works well in practice, as described in Figure 3.

The proposed algorithm starts with a minimal Layered-DAG G^* that contains only the highest weighted relation in R (Steps 1-8). Then, it moves an edge in G to G^* once a time, ensuring that adding the new edge still keeps G^* a valid Layered-DAG (Step 11), and the new edge has the highest weights among all valid candidates (Steps 9-10).

5 Experiments

In this section, we show the experimental results of proposed methods. Specifically, we focus on the following points:

- The quality of discovered subsumption relations by different methods.
- The characteristics of wrong subsumption relations discovered.
- The effect of Layered-DAG construction on the quality of relations.
- Empirical study of the resulted Layered-DAG.

Name	N	$\bar{N}_{tag}$	$\bar{N}_{content}$
BLOG	100,192	2.78	332.87
BOOK	110,371	8.51	204.76

Table 1: Statistics of the data sets. N is the number of documents. $\bar{N}_{tag}$ is the mean number of tags per document. $\bar{N}_{content}$ is the mean number of words per document.

5.1 Data Sets

We use two real world social tagging data sets. The first data set, named BLOG, is a collection of blog posts annotated by blog authors, which is crawled from the web. The second data set, named BOOK, is from a book collecting and sharing site[2], which contains description of Chinese books and user contributed tags. Table 1 lists the basic statistics of the data sets.

The two data sets have different characteristics. Documents in BLOG are longer, not well written, and the number of tags per document is small. To the opposite, documents in BOOK are shorter but well written, and there are more tags for each document.

5.2 Discovered Subsumption Relations

5.2.1 Experimental Settings

For BLOG, we use the tags that have been used more than 10 times; For BOOK, we use the tags that have been used more than 50 times. We perform 100 iterations of Gibbs sampling when training the TAM model, with first 50 iterations as the burn-in iterations. All the estimation methods require proper smoothing. Here we use additive smoothing for all methods, which adds a very small number (0.001 in our case) to all raw counts. Sophisticated smoothing method could be employed, but is out of the scope of this paper.

5.2.2 Evaluation

We use *precision* and *coverage* to evaluate the discovered relations at any given cut-off threshold n. First, we sort the discovered relations by their weights in descending order. Then, we take the top-n relations, discarding the others. For the remaining relations, precision is computed as N_c/n, N_c is the number of correct relations in the top-n list; coverage is computed as $N_t/|T|$, where N_t is the number of unique tags appeared in the top-n list, and $|T|$ is the total number of tags.

To get N_c, the number of correct relations, we need a standard judgement of the correctness of relations, which involves human labeling. To minimize the bias in human assessment, we use **pooling**, which is a widely accepted method in Information Retrieval research (Voorhees and Harman, 2005). Pooling works as follows: First, relations obtained by different methods are mixed together, creating a pool of relations. Second, the pool is shuffled, so that the labeler cannot identify the source of a single relation. Third, annotators are requested to label the relations in the pool as correct or incorrect, based on the definition of subsumption relation. After all relations in the pool are labeled, we use them as the standard judgement to evaluate each method's output.

Precision measures the proportion of correct relations, while coverage measures the proportion of tags that are connected by the relations. The cut-off threshold n affects both precision and coverage: the larger the n, the lower the precision, and the higher the coverage.

5.2.3 Baseline methods

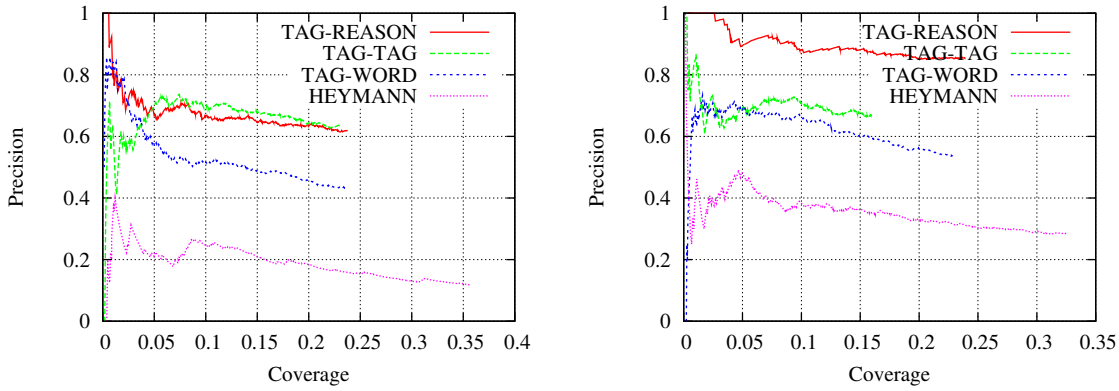
Besides TAG-TAG, TAG-WORD and TAG-REASON, we also include the method described in (Heymann and Garcia-Molina, 2006) as a baseline, denoted as HEYMANN. HEYMANN method was designed to find similar-to relation rather than subsumption relation. The similar-to relation is symmetric, while subsumption relation is more strict and asymmetric. In our experiments, we use the same evaluation process to evaluate TAG-TAG, TAG-WORD, TAG-REASON and HEYMANN, in which only subsumption relations will be marked as correct.

5.2.4 Results

For each method, we set the cut-off threshold n from 1 to 500, so as to plot the psrecision-coverage curves. The result is shown in Figure 4. The larger the area under the curve, the better the method's performance.

We have three observations from Figure 4. First, TAG-REASON has the best performance

[2] http://www.douban.com



(a) BLOG (b) BOOK

Figure 4: The precision and coverage of TAG-TAG, TAG-WORD, TAG-REASON and HEYMANN methods. The larger the area under the curve, the better the result. The cut-off threshold $n \in [1, 500]$.

BLOG			BOOK		
Insufficient	Reversed	Irrelevant	Insufficient	Reversed	Irrelevant
childedu $\rightarrow_s$ father	stock $\rightarrow_s$ security	travel$\rightarrow_s$building	textbook $\rightarrow_s$ exam	English $\rightarrow_s$ foreignlang	japan $\rightarrow_s$lightnovel
childedu $\rightarrow_s$ grandma	stock $\rightarrow_s$ financial	emotion $\rightarrow_s$time	history $\rightarrow_s$ military	biography$\rightarrow_s$people	building$\rightarrow_s$textbook
emotion$\rightarrow_s$warm	delicious$\rightarrow_s$taste	emotion$\rightarrow_s$original	piano$\rightarrow_s$scores	jpbuilding$\rightarrow_s$jpculture	sales$\rightarrow_s$O
childedu$\rightarrow_s$child	delicious$\rightarrow_s$food	culture$\rightarrow_s$spring	history$\rightarrow_s$culture	novel$\rightarrow_s$pureliterature	japan $\rightarrow_s$ shower
education$\rightarrow_s$child	earthquake$\rightarrow_s$disaster	poem$\rightarrow_s$night	novel$\rightarrow_s$love	ancientgreek$\rightarrow_s$greek	photo$\rightarrow_s$umbrella
Total 52%	Total 14%	Total 34%	Total 37%	Total 48%	Total 15%

Table 2: Examples of mistakes and the percentage of each mistake type.

on both data sets: On the BOOK data set, TAG-REASON outperforms others by a marked margin; On the BLOG data set, TAG-REASON has higher precision when coverage is smaller (which means within top-ranked relations), and has comparable precision to TAG-TAG when coverage increases. Second, similarity-based clustering method (namely HEYMANN) performed worse than others, suggesting it may not be adequate for discovering subsumption relation. Third, while also using content information, TAG-WORD performs poorer than both TAG-REASON and TAG-TAG, which suggests that noise in the content would prevent TAG-WORD from getting the correct estimation of $p(t_a|t_b)$.

To summarize, by leveraging *relevant* content, TAG-REASON could discover better subsumption relations than just using tag-tag co-occurrences and similarity-based hierarchical clustering.

5.2.5 Mistakes in Discovered Relations

We also studied the type of mistakes in subsumption relation discovery. To our observation, a mistakenly discovered relation $t_a \rightarrow_s t_b$ falls into one of the following categories:

1. **insufficient** t_a relates with t_b, but using t_b does not implies the using of t_a in all cases.
2. **reversed** $t_b \rightarrow_s t_a$ is correct, while $t_a \rightarrow_s t_b$ is not.
3. **irrelevant** There is no obvious connection between t_a and t_b.

We collected all incorrect relations discovered by the TAG-REASON method. Then, the type of mistake for each relation is labeled manually. The result is shown in Table 2, along with selected examples of each type.

Table 2 shows different error patterns for BLOG and BOOK. In BLOG, most of the mistakes are of the type *insufficient*. Taking "`education` $\rightarrow_s$ `child`" for example, annotating a document as `child` does not imply that it is about child education, it may about food or clothes for a child. In BOOK, most of the mistakes are *reversed* mistakes, which is a result of the omitted-tag problem discussed in Section 3.2.1.

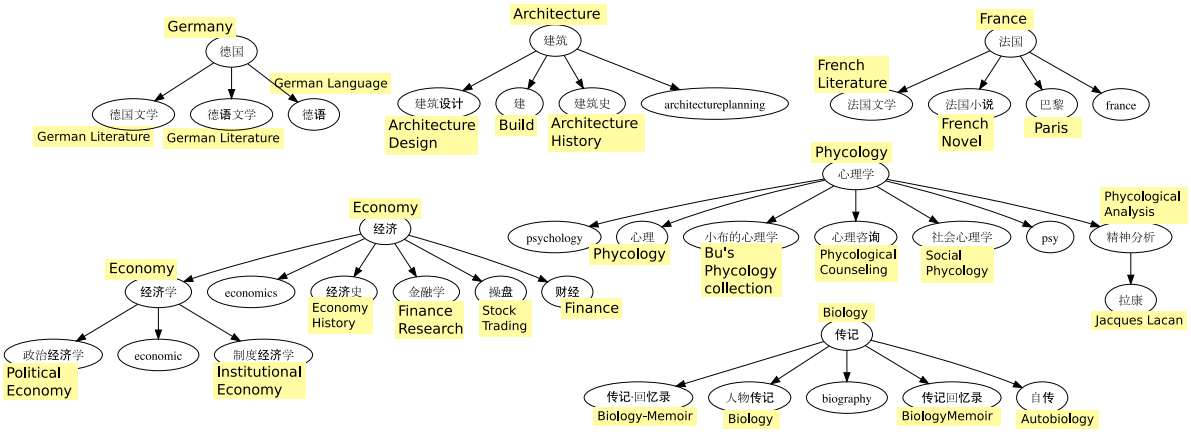


Figure 5: Part of the constructed Layered-DAG from the BOOK data set.

	BLOG		BOOK	
Method	Precision	Coverage	Precision	Coverage
TAG-TAG	-4.7%	$+7.9\%$	-7.4%	$+12.5\%$
TAG-WORD	0%	0%	-9.0%	$+2.2\%$
TAG-REASON	-3.6%	$+5.4\%$	-0.9%	$+5.4\%$

Table 3: The effects on precision and coverage by Layered-DAG construction

5.3 Layered-DAG Construction

Using the algorithm introduced in Section 4, we constructed Layered-DAGs from the discovered relations. Constructing Layered-DAG will remove certain relations, which will decrease the precision and increase the coverage. Table 3 shows the changes of precision and coverage brought by Layered-DAG construction. In most of the cases, the increasing of coverage is more than the decreasing of precision.

As a representative example, we show part of a constructed Layered-DAG from the BOOK data set in Figure 5, since the whole graph is too big to fit in the paper. All tags in Chinese are translated to English.

6 Conclusion and Future Work

In this paper, we explored the structure of social tags by discovering subsumption relations. First, we defined the subsumption relation $t_a \rightarrow_s t_b$ as t_a can be used to replace t_b without ambiguity. Then, we cast the subsumption relation identification problem to the estimation of $p(t_a|t_b)$. We proposed three methods, namely TAG-TAG, TAG-WORD and TAG-REASON, while the last two leverage the content of document to help estimation. We also proposed an greedy algorithm for constructing a Layered-DAG from the discovered relations, which helps minimizing redundancy.

We performed experiments on two real world data sets, and evaluated the discovered subsumption relations quantitatively by pooling. The results showed that the proposed methods outperform similarity-based hierarchical clusteing in finding subsumption relations. The TAG-REASON method, which uses only the relevant content to the tags, has the best performance. Empirical study showed that Layered-DAG construction works effectively as expected.

The results suggest two directions for future work: First, more ways for $p(t_a|t_b)$ estimation could be explored, for example, combining TAG-TAG and TAG-REASON; Second, external knowledge, such as the Wikipedia and the WordNet, could be exploited as background knowledge to improve the accuracy.

ACKNOWLEDGEMENTS

This work is supported by the National Science Foundation of China under Grant No. 60873174 and the National 863 High-Tech Program of China under Grant No. 2007AA01Z148. We also thank Douban Inc.(www.douban.com) for providing the DOUBAN data set, and Shoukun Wang, Guozhu Wen et al. of Douban Inc. for insightful discussion.

References

Angeletou, S., M. Sabou, L. Specia, and E. Motta. 2007. Bridging the gap between folksonomies and the semantic web: An experience report. In *Workshop: Bridging the Gap between Semantic Web and Web*, volume 2. Citeseer.

Begelman, Grigory, Keller, and F. Smadja. 2006. Automated tag clustering: Improving search and exploration in the tag space. In *Collaborative Web Tagging Workshop, 15 th International World Wide Web Conference*.

Brooks, Christopher H. and Nancy Montanez. 2006. Improved annotation of the blogosphere via autotagging and hierarchical clustering. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 625–632, New York, NY, USA. ACM.

Echarte, F., J. J. Astrain, A. Córdoba, and J. Villadangos. 2007. Ontology of folksonomy: A New modeling method. *Proceedings of Semantic Authoring, Annotation and Knowledge Markup (SAAKM)*.

Heymann, Paul and Hector Garcia-Molina. 2006. Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical Report 2006-10, Stanford University, April.

Kim, Hak L., Simon Scerri, John G. Breslin, Stefan Decker, and Hong G. Kim. 2008. The state of the art in tag ontologies: a semantic model for tagging and folksonomies. In *DCMI '08: Proceedings of the 2008 International Conference on Dublin Core and Metadata Applications*, pages 128–137. Dublin Core Metadata Initiative.

Kome, Sam H. 2005. Hierarchical subject relationships in folksonomies. Master's thesis, University of North Carolina at Chapel Hill, November.

Mika, P. 2005. Ontologies are us: A unified model of social networks and semantics. *The Semantic Web–ISWC 2005*, pages 522–536.

Passant, Alexandre. 2007. Using ontologies to strengthen folksonomies and enrich information retrieval in weblogs. In *Proceedings of International Conference on Weblogs and Social Media*.

Sanderson, M. and B. Croft. 1999. Deriving concept hierarchies from text. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–213. ACM.

Schmitz, P. 2006. Inducing ontology from flickr tags. In *Collaborative Web Tagging Workshop at WWW2006, Edinburgh, Scotland*, pages 210–214. Citeseer.

Schwarzkopf, E., D. Heckmann, and D. Dengler. 2007. In *Workshop on Data Mining for User Modeling, ICUM'07*, page 63. Citeseer.

Shepitsen, Andriy, Jonathan Gemmell, Bamshad Mobasher, and Robin Burke. 2008. Personalized recommendation in collaborative tagging systems using hierarchical clustering. In *Proceedings of ACM RecSys'08*.

Si, Xiance and Maosong Sun. 2010. Tag allocation model: Modeling noisy social annotations by reason finding. In *Proceedings of 2010 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*.

Specia, Lucia and Enrico Motta. 2007. Integrating folksonomies with the semantic web. pages 624–639.

Van Damme, C., M. Hepp, and K. Siorpaes. 2007. Folksontology: An integrated approach for turning folksonomies into ontologies. *Bridging the Gap between Semantic Web and Web*, 2:57–70.

Voorhees, E.M. and D.K. Harman. 2005. *TREC: Experiment and evaluation in information retrieval*. MIT Press.

Wu, Harris, Mohammad Zubair, and Kurt Maly. 2006a. Harvesting social knowledge from folksonomies. In *HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia*, pages 111–114, New York, NY, USA. ACM.

Wu, Xian, Lei Zhang, and Yong Yu. 2006b. Exploring social annotations for the semantic web. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 417–426, New York, NY, USA. ACM.