Tag Suggestion Method based on Association Pattern and Bigram Approach

Hyunwoo Kim
Seoul National University
Seoul, Korea
hwkim@idb.snu.ac.kr

Kangpyo Lee
Seoul National University
Seoul, Korea
kplee@idb.snu.ac.kr

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

Hyoung-Joo Kim
Seoul National University
Seoul, Korea
hjk@snu.ac.kr

***Abstract*—Recently, the number of articles, blog posts, photos and videos on the web is dramatically increasing because of the increase of internet usage. In this situation, the web search is the most important thing in the web. When we search, we can use text information from articles or blog posts. In the case of photos and videos, we can only use a title. If there are tags − significant keywords of that multimedia, we can use tag information to search. Tag is a keyword of text, blog post, or multimedia. Users have already recognized about the value and importance of tags but only a few users are using tags. They might be annoying to add tags or they don't know what to add for good search result. This is why tag suggestion system is needed. Our method analyzes crawled tag data and suggests appropriate tags to user using association pattern and bigram approach. By experiments, we conclude that our tag suggestion method suggests appropriate tags.**

Keywords-Tag Suggestion; Association Pattern; Bigram; Web 2.0; Folksonomy

I. INTRODUCTION

These days, there are many UGC (User Generated Contents) on the web. Internet users create thousands of articles, blog posts, photos, and videos. In this situation, the process of getting information is more important than any other thing. We have to search if we want to find some information on the web. Search engine can use text information from articles or blog posts but it cannot use any text information from photos or videos. The only information can be used in the multimedia data is the title of the multimedia. The title is not very useful because it is typically very short or not descriptive [1]. If we use tags in the multimedia search, it will be helpful.

Tag is a meaningful keyword that describes corresponding content. If there is a diagram about JDBC, we can add tags, such as *programming*, *Java*, *DBMS*, and *JDBC*. These tags will help the result of web search. We can use tag information as well as the title of that content for searching. Tag is also used for personal reason. User can use tags for organizing and indexing, such as *to read*, *funny*, and *my stuff*.

People already know about the merit of using tags and they have motivation to make their multimedia more accessible to the public [2]. However, tagging − action of adding tags to content − is a little bit annoying job. Users even don't know which tag they have to add for better accessibility. This is the reason why tag suggestion system is needed. When a user adds tags to content, some appropriate tags can be suggested for better accessibility and for better search result.

In this paper, we provide tag suggestion method based on association pattern and bigram. We discuss related work in Section 2. In Section 3, we introduce our suggestion method in detail. We evaluate our method in Section 4 and conclude this paper in Section 5.

II. RELATED WORK

People already know advantage of tag suggestion approach. It helps tagging process and improves tag quality. There are some tag suggestion approaches. If full text information is available, tag suggestion is considerably easier than no text information [3]. This method finds similar blog posts using some retrieval models and suggests based on blog posts.

There exist some tag suggestion methods without full text information [4]. They suggest tags based on the resource title and a lexicon. They don't use full text information but part of text information. Most of other methods are focusing on broad audience folksonomies but their method is focusing on individual users.

There also exist some tag suggestion methods without any text information [5]. They analyze how users tag photos and what kind of tags they provide in Flickr, and they propose tag recommendation strategies. Their method is based on photo contents which have no text at all. There is an evaluation concept to compare different suggestion methods [6]. For each tagging use case, we should not adjust same strategy for them. A certain method works well in some cases but it may not work well

in other cases. They introduced relevant comparison measures.

Tag clustering method is not directly for tag suggestion method but for visualization of folksonomy [7]. However, it can be extended for tag suggestion. We do clustering tags before suggestion, and recommend appropriate tags in same cluster.

To the best of our knowledge, current researches on tag suggestion retrieve the most frequent tags and aggregate list, find similar posts and recommend tags of those posts, or cluster tags and recommend tags in the same cluster. Most methods are based on co-occurrence between the tags.

III. TAG SUGGESTION METHOD

We introduce association pattern and bigram approach to suggest appropriate tags. Association patterns about relations of tags are extracted. We use bigram for co-occurrence. It means that all tags in the same content are not considered as co-occurred tags. Only adjacent tags in the same content are co-occurrence tags. This method doesn't use any textual information from the contents. It uses tag itself. As long as the contents have tags, this method could be applied to not only blogs and articles but also any multimedia contents.

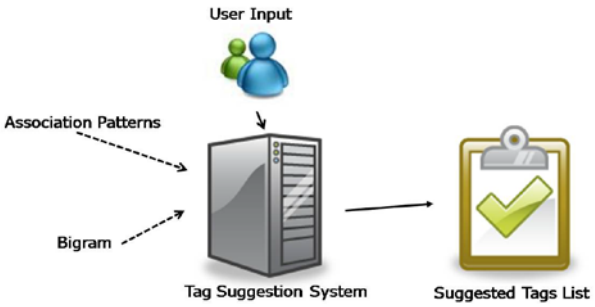

Figure 1: Overall System

A. Association Pattern

As we mentioned previously, current researches on tag suggestion are based on co-occurrence tags. These approaches could get wrong results because they only care one tag at a time. For example, if a user inputs *Apple* and *Farm* as tags, people knows this *Apple* means a fruit not a company. However, in the concept of co-occurrence, current tag suggestion systems don't know exactly whether *Apple* means a fruit or a company. They make candidate tag lists for *Apple* and *Farm* separately, and make an aggregated ranked list. It is possible that these methods may provide *Mac*, *iPod*, and *Fruit* as candidate tags.

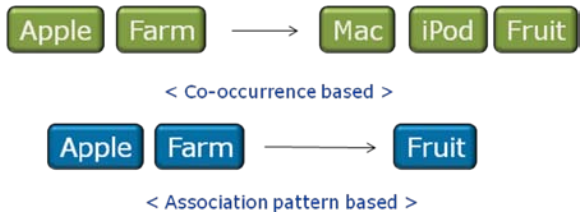

Figure 2: Difference between co-occurrence and association pattern

Association pattern is a kind of data mining technology and mostly used in marketing [9]. In the sales marketing analysis, association pattern {*beer, water* implies *diaper*} means that some customers who already bought a bottle of beer and water tend to buy a diaper. In tagging, association pattern {*delicious, bookmarking* implies *Web 2.0*} means that some users who already tagged *delicious* and *bookmaking* in their content tend to add *Web 2.0* to same content. Association pattern concerns not one tag at a time but whole context at a time. So this method can avoid word sense disambiguation problem. In the previous *Apple* example, association pattern considering *Apple* and *Farm* together so that it doesn't suggest *Mac* or *iPod*.

B. The Process of Though

People tend to develop their thoughts using association. First, thinking about something and then, thinking about another thing related with previous though, and so on. For example, we sequentially think *programming*, *Java*, *JDBC*, and *DBMS*. It is not easy to think *programming* and *JDBC* directly. In this case, the adjacent tags are more related with each other. *Programming* and *Java* are more related than *programming* and *DBMS*. We assume that people would develop their thoughts in a same way when they add tags to content. This concept is similar to bigram approach in natural language processing. This is the reason why we use bigram as a tag suggestion method.

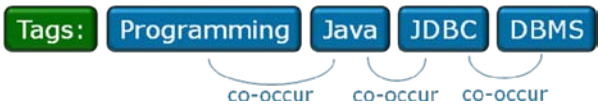

Figure 3: Bigram for tags

In this paper, we take adjacent tags only. In the previous example, *programming* and *Java* would be considered as co-occurred tags but *programming* and *DBMS* would not. It prevents too much tags are appeared in the result and it also suggests more significant tags.

C. Tag Suggestion Method

Our tag suggestion method is based on collective knowledge of public web users. We use tag data which is pre-

viously built by the public. When a user enters some tags for tagging her content, our tag suggestion method will suggest appropriate tags based on user-entered tags and our tag data.

We get tag data from CiteULike [1] and crawl from del.icio.us[2]. We cannot use these data directly. Preprocessing is needed for using association pattern and bigram. Based on our crawled data, we make association patterns and their confidence values. The confidence of the association pattern is the conditional probability of that pattern. If there exists an association pattern ***p*** {*Flickr* and *Yahoo* imply *Photo*}, the confidence of the pattern ***p*** is

$$Confidence(p) = \frac{P(Flickr, Yahoo, Photo)}{P(Flickr, Yahoo)} \quad (1)$$

There are A_{score}, B_{score}, and T_{score} for our method. A stands for association pattern, B stands for bigram, and T stands for tag.

Each association pattern has confidence value. $A_{score}(\boldsymbol{p})$, the score of the association pattern ***p,*** is the confidence of the pattern ***p***. $B_{score}(\boldsymbol{t}_u, \boldsymbol{t}_c)$, the score between user entered tags $\boldsymbol{t}_u$ and a candidate tag $\boldsymbol{t}_c$, is calculated using bigram.

$$B_{score}(t_u, t_c) = \frac{1}{N_u} \sum_{u \in t_u} \frac{P(u, t_c)}{P(t_c)} \quad (2)$$

N_u is the number of tags in $\boldsymbol{t}_u$ which is entered by the user. B_{score} is actually the average probability of bigram conditional probabilities between user-entered tags $\boldsymbol{t}_u$ and candidate tag $\boldsymbol{t}_c$.

Figure 4: Calculating B_{score} for Association Pattern

For example, when the association pattern is {*delicious, bookmarking* imply *Web 2.0*}, $\boldsymbol{t}_u$ is *delicious* and *bookmarking*, $\boldsymbol{t}_c$ is *Web 2.0*. Then, B_{score} of this association pattern is the average probability of bigram conditional probabilities (*delicious, Web2.0*) and (*bookmarking, Web2.0*).

$$\frac{1}{2}\left(\frac{P(delicious, Web2.0)}{P(Web2.0)} + \frac{P(bookmarking, Web2.0)}{P(Web2.0)}\right) \quad (3)$$

$T_{score}(\boldsymbol{t}_c)$, the score of the candidate tag $\boldsymbol{t}_c$, is calculated by the weighted product of A_{score} and B_{score}.

$$T_{score}(t_c) = \alpha \cdot A_{score}(p_b) \times \beta \cdot B_{score}(t_u, t_c) \quad (4)$$

The purpose of our tag suggestion method is making a ranked list of candidate tags and retrieving top *k* tags for suggestion based on T_{score}.

Algorithm: *Suggestion*($\boldsymbol{t}_u$)
Input: tags $\boldsymbol{t}_u$ which user entered Output: a list of candidate tags for suggestion 1. Construct P in the form of P = {***p*** | ***p*** satisfies $\boldsymbol{t}_u$} 2. C ← Φ 3. For each ***p*** in P, do loop 3.1 Add candidate tags $\boldsymbol{t}_c$ in the right side of ***p*** to C 4. For each $\boldsymbol{t}_c$ in C 4.1 Find the best association pattern $\boldsymbol{p}_b$ which maximizes $T_{score}(\boldsymbol{t}_c)$ 4.2 Set the tag score of $\boldsymbol{t}_c$ as $T_{score}(t_c) = \alpha \cdot A_{score}(p_b) \times \beta \cdot B_{score}(t_u, t_c)$ 5. L ← Φ 6. Retrieve top *k* tags based on T_{score} 6.1 Add retrieved tags to L **return** List of candidate tags L

Table 1: Tag Scoring Algorithm

The process of our method is the followings. For given tags $\boldsymbol{t}_u$ which user already entered, our method finds all association patterns ***p*** which satisfy the conditions of given tags $\boldsymbol{t}_u$. Association pattern satisfies the conditions means that the left side of that association pattern is subset of user entered tags $\boldsymbol{t}_u$. For instance, if a user enters *Flickr*, *Yahoo*, and *photo* as tags, the following patterns can be retrieved.

Flickr, Yahoo → *Web 2.0*
photo → *photographer*
Flickr, Yahoo, photo → *tag, Web 2.0*
Yahoo → *news*

Then candidate tags will be the right side of the association patterns, such as *Web 2.0*, *photographer*, *tag*, and *news*. For each candidate tag $\boldsymbol{t}_c$, it computes B_{score} based on $\boldsymbol{t}_u$ and $\boldsymbol{t}_c$. Based on previous works, our system finds the best association pattern $\boldsymbol{p}_b$ which maximizes $T_{score}(\boldsymbol{t}_c)$.

$$p_b = \arg\max_p (\alpha \cdot A_{score}(p) \times \beta \cdot B_{score}(t_u, t_c)) \quad (5)$$

[1] http://www.citeulike.org
[2] http://www.delicious.com

α and β are weight values. These values are empirically determined. T_{score} of the candidate tag $\boldsymbol{t}_c$ is calculated by (4). $\boldsymbol{p}_b$ in (4) is selected by (5). Our approach calculates the T_{score}'s of all candidate tags, and it makes ranked list based on T_{score} of the tag. Finally, retrieving top *k* scored tags and suggesting them to user. Table 1 is pseudo code for our algorithm.

IV. EVALUATION

In the following experiments, we evaluate our method based on tag data from CiteULike and del.icio.us. These two sites are much alike from collaborative tagging point of view. Their tagging systems are collaborative tagging. Collaborative tagging means that more than two people can add tags to the same content. But they also have different characteristics. CiteULike is the web site for archiving and sharing of academic papers. People post papers and tagging them. del.icio.us is the web site for bookmarking. Anyone can add her own bookmarks on the web and tagging them. These URLs can be shared with any other person on the Web. We use tag data of these two sites. del.icio.us has many tagged contents. Almost all contents of del.icio.us have more than a tag but they are not well-defined tags. A few contents of CiteULike have no tag at all but others have very well-organized tags because users of CiteULike are researchers. This difference between them has an effect on the suggestion result.

A. Evaluation Setup

To apply our method, we need to make association patterns and bigram counts from corpus. We select 50,000 contents from CiteULike and del.icio.us, respectively. Each contents, such as a photo, a video, or a blog post, has a series of tags. A series of tags is considered as a transaction. For example, when the content is about del.icio.us, a series of tags could be *delicious*, *bookmarking*, and *Web2.0*. From these data, we extract association patterns and bigram counts for suggesting tags.

B. Evaluation Method

We tested various numbers of samples. Of course, these samples are not included in the above corpus. Selected contents for evaluation setup and samples are distinct.

We checked the probability of correct suggestion for samples. It is difficult to determine whether suggested tags are correct answers or not. It used to be evaluated by human decision in other papers. This evaluation method is easy and seems good but it depends on testers. When a tag is suggested to testers, some tester could regard this tag as correct answer, others could regard this tag as not correct answer, or the others could not know whether this tag is correct answer or not. So, we take another strategy for evaluation.

As we already mentioned, a sample transaction consists of a series of tags. When the tags are *delicious*, *bookmarking*, and *Web2.0*, our method splits the tags into two groups, user-inputs and answers. If *delicious* is a user-input, then *bookmarking* and *Web2.0* are answers. In this situation, our method regards *delicious* as a user-entered tag $\boldsymbol{t}_u$ and makes a list of suggested tags based on $\boldsymbol{t}_u$. For a suggested tag $\boldsymbol{t}_s$, if there exist same tag in the answers, $\boldsymbol{t}_s$ is considered as correct. If there is no same tag in the answers, $\boldsymbol{t}_s$ is not correct.

correct: the suggested tag $\boldsymbol{t}_s$ is in the answers
not correct: the suggested tag $\boldsymbol{t}_s$ is not in the answers

This strategy seems a little bit strict but other strategies require external method. For example, human should make a decision or similarity measure should be a substitute. Similarity between suggested tag $\boldsymbol{t}_s$ and answers can be a criterion of correctness. We think our evaluation method is simpler and more efficient.

The following is suggested tag example in our evaluation step. Sample transaction consists of *rescue*, *livecd*, *tools*, *software*, *opensource*, *backup*, and *sysadmin*. When they are separated *opensource* as a user-entered tag $\boldsymbol{t}_u$ and others as answers, the first suggested tag is *software*. This is correct answer because *software* is in answers.

C. Evaluation Result

Based on evaluation setup and method, we evaluate our tag suggestion method. We check the probability of correctly suggested tags in various conditions. We evaluate our method in three categories.

C@1: the probability of the top scored tag in suggested tag list is correct
C@3: the probability of at least one of top 3 tags in suggested tag list is correct
P@3: precision of top 3 tags in suggested tag list.

C@n means that the probability of at least one of top *n* tags in the suggested tag list is correct. *P@n* is the ratio of correct tags in the top *n* tags in the suggested tag list.

Test set	P@3	C@1	C@3	Diff.
D100	0.5848	0.5811	0.6538	12.51%
D200	0.5365	0.5199	0.6044	16.25%
D300	0.5221	0.5124	0.5872	14.60%
C100	0.5989	0.6364	0.6818	7.13%
C200	0.7198	0.7246	0.7381	2.00%
C300	0.7956	0.8035	0.8273	2.96%

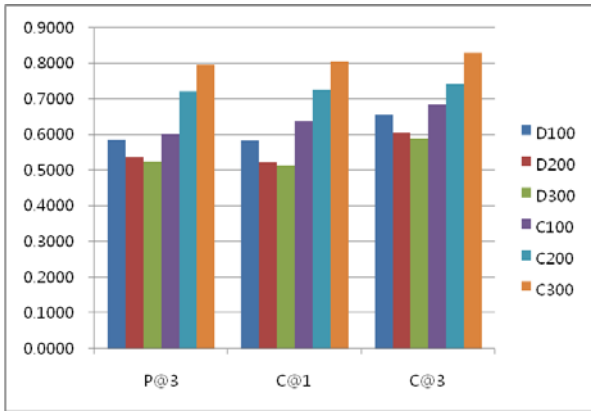

Table 2: Evaluation Result

Table 2 shows the result of our evaluation. Test set which starts from *D* is del.icio.us data and test set which starts from *C* is CiteULike data. *100* of *D100* means the number of test cases. *D100* is a subset of *D200* and *D200* is a subset of *D300*. *Diff.* is the difference between C@1 and C@3.

D. Result Analysis

In the correctness (*C@n*) and precision (*P@n*), the average result from CiteULike is higher than from del.icio.us.

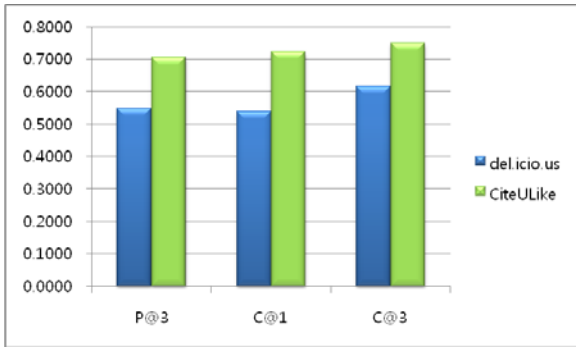

Table 3: Difference between CiteULike and del.icio.us data

This different comes from the characteristics of them. Users of CiteULike are researchers and they use formal word and controlled vocabulary. Their tags are limited in their research area and these tags are of small number. Users of del.icio.us, however, are the public. Anyone can add any word to any content. They use not only the formal word but also informal language. They even make their own word and use tags as personal purpose, such as *myStuff*, *toRead*, or *fun*. Their tags could be in the dictionary or not. Because our method is based on the corpus data, its quality effects on the result. CiteULike data is more accurate and formally refined, so we could get a better result.

The result of precision is between 0.5211 and 0.7956. It implies that our method suggests about 6 correct tags out of 10 suggested tags. If we take similarity measure or human decision, this precision value will go up much higher.

The difference between C@1 and C@3 is not that big.

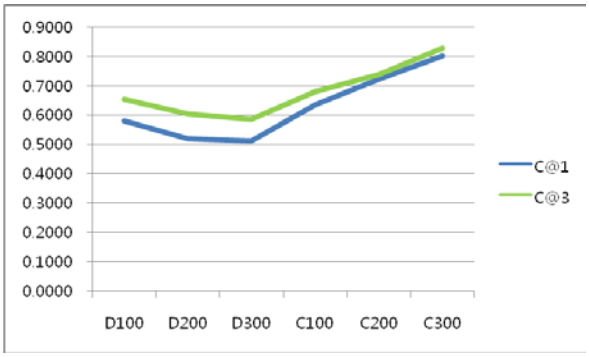

Table 4: Difference between C@1 and C@3

The average difference is about 4% in case of CiteULike. This result implies that the top scored tag is a correct answer in the 96 suggestions out of the 100 suggestions. Only 4 suggestions out of 100 suggestions don't. It means our tag suggestion method works well in scoring tags.

V. CONCLUSTION

To get the information which we want, we have to search because too many contents on the web. In the web search, tags could be simple and efficient indexes, especially for multimedia contents. However, current tag data is not sufficiently enough and most web site don't provide tag suggestion system. Even tagging bothers users. This is the reason why the tag suggestion system is needed.

In this paper, we propose tag suggestion method using association pattern and bigram approach. We confirm our suggestion method works well by experiments. Our method can be applied to both textual contents and non-textual contents. It is important because multimedia data has no textual information except a title. Proposed algorithm has good characteristics for multimedia data.

There are also limitations of our method. We figure out that quantity of corpus data is important and quality of corpus data is much more important factor by evaluation result. Our method is sensitive to the quality of data. It cannot be directly applied to dynamic data because our approach needs preprocessing.

For the future work, we will try not only bigram but also trigram, 4-gram, n-gram and compare them. We will try another method as a substitute of association pattern. Association pattern is good for managing all user entered

tags at once but it’s not easy to find association patterns among corpus data and takes some time.

ACKNOWLEDGE

This research was supported by the Brain Korea 21 Project, 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), and a grant (07High Tech A01) from High tech Urban Development Program funded by Ministry of Land, Transportation and Mari-time Affairs of Korean government.

REFERENCES

[1] N. Garg and I. Weber, "Personalized tag suggestion for flickr," in *WWW*, 2008.

[2] M. Ames and M. Naaman, "Why we tag: motivations for annotation in mobile and online media," in *CHI*, 2007, pp. 971-980.

[3] G. Mishne, "AutoTag: a collaborative approach to automated tag assignment for weblog posts," in *WWW*, 2006, pp. 953-954.

[4] M. Lipczak, "Tag Recommendation for Folksonomies Oriented towards Individual Users," *ECML PKDD Discovery Challenge*, p. 84, 2008.

[5] B. Sigurbjornsson and R. Zwol, "Flickr Tag Recommendation based on Collective Knowledge," in *WWW*, 2008.

[6] S. Oldenburg, L. Zielinski, M. Garbe, and C. Cap, "Comparative Analysis of Tag Suggestion Algorithms," in *SNA-KDD*, 2008, pp. 24-27.

[7] Y. Hassan-Montero and V. Herrero-Solana, "Improving Tag-Clouds as Visual Information Retrieval Interfaces," 2006.

[8] P. N. Tan, M. Stenbach, and V. Kumar, *Introduction to Data Mining*: Addison Wesley, 2006.

[9] J. Hipp, U. Guntzer, and G. Nakhaeizadeh, "Algorithms for association rule mining - general survey and comparison," *ACM SIGKDD Explorations Newsletter*, vol. 2, pp. 58-64, 2000.

[10] D. Jurafsky and J. H. Martin, *Speech and Language Processing*: Pearson Education International, 2008.