Bringing the Associative Ability to Social Tag Recommendation

Miao Fan,★◇ **Yingnan Xiao**◇ and **Qiang Zhou**★
★ Department of Computer Science and Technology, Tsinghua University
◇School of Software Engineering, Beijing University of Posts and Telecommunications
{fanmiao.cslt.thu,lxyxynt}@gmail.com,
zq-lxd@mail.tsinghua.edu.cn

Abstract

Social tagging systems, which allow users to freely annotate online resources with tags, become popular in the Web 2.0 era. In order to ease the annotation process, research on social tag recommendation has drawn much attention in recent years. Modeling the social tagging behavior could better reflect the nature of this issue and improve the result of recommendation. In this paper, we proposed a novel approach for bringing the associative ability to model the social tagging behavior and then to enhance the performance of automatic tag recommendation. To simulate human tagging process, our approach ranks the candidate tags on a weighted digraph built by the semantic relationships among meaningful words in the summary and the corresponding tags for a given resource. The semantic relationships are learnt via a word alignment model in statistical machine translation on large datasets. Experiments on real world datasets demonstrate that our method is effective, robust and language-independent compared with the state-of-the-art methods.

1 Introduction

Social tagging systems, like Flickr[1], Last.fm[2], Delicious[3] and Douban[4], have recently become major infrastructures on the Web, as they allow users to freely annotate online resources with personal tags and share them with others. Because of the no vocabulary restrictions, there are different kinds of tags, such as tags like keywords, category names or even named entities. However, we can still find the inner relationship between the tags and the resource that they describe. Figure 1 shows a snapshot of a social tagging example, where the famous artist, Michael Jackson was annotated with multiple social tags by users in Last.fm[2]. Actually, Figure 1 can be divided into three parts, which are *the title*, *the summary* and *the tags* respectively.

Figure 1: A music artist entry from website Last.fm[2]

We can easily find out that social tags concisely indicate the main content of the given online resource and some of them even reflect user interests. For this reason, social tagging has been widely studied and applied in recommender systems (Eck et al., 2007; Musto et al., 2009; Zhou et al., 2010), advertising (Mirizzi et al., 2010), etc.

For the sake of easing the process of user annotation and providing a better effect of human-computer interaction, researchers expected to build

[1] http://www.flickr.com
[2] http://www.lastfm.com
[3] http://delicious.com
[4] http://www.douban.com

Proceedings of the TextGraphs-7 Workshop at ACL, pages 44–54,
Jeju, Republic of Korea, 12 July 2012.

automatic social tagging recommender systems, which could automatically suggest proper tags for a user when he/she wants to annotate an online resource. By observing huge amount of online resources, researchers found out that most of them contain summaries, which could play an important role in briefly introducing the corresponding resources, such as the artist entry about Michael Jackson in Figure 1. Thus some of them proposed to automatically suggest tags based on resource summaries, which are collectively known as the *content-based approach* (F. Ricci et al., 2011).

The basic idea of *content-based approach* in recommender systems is to select important words from summaries as tags. However, this is far from adequate as not all tags are statistically significant in the summaries. Some of them even do not appear in the corresponding summaries. For example, in Figure 1, the popular tag *dance* does not appear in the summary, but why most of users choose it as a proper tag to describe Michael Jackson. This "out-of-summary" phenomenon reflects a fact that users usually exploit their own knowledge and associative ability to annotate online resources. When a summary comes, they associate the important words in the summary with other semantic-related tags based on their knowledge. To improve the automatic tag recommendation, a social computing issue (Wang et al., 2007), modeling the *social tagging behavior* is the straightforward way. Namely, how to analyze the human tagging process and propose a suitable approach that can help the computer to simulate the process are what we will explore in this paper.

The novel idea of our approach is to rank the candidate tags on a weighted digraph built by the semantic relationships among meaningful words in the summary and the corresponding tags for a given resource. The semantic relationships are learnt via a word alignment model in statistical machine translation. Our approach could bring the associative ability to social tag recommendation and naturally simulate the whole process of human social tagging behavior and then to enhance the performance of automatic tag recommendation. So, we name this approach for *Associative Tag Recommendation* (ATR).

The remainder of the paper is organized as follows. Section 2 analyzes the process of human tagging behavior. Section 3 describes our novel approach to simulate the process of human tagging behavior for social tag recommendation. Section 4 compares our approach with the state-of-the-art and baseline methods and analyzes the parameter influences. Section 5 surveys some related work in social tag recommendation. Section 6 concludes with our major contributions and proposes some open problems for future work.

2 Human Tagging Behavior Analysis

Here, we will analyze the human tagging process to discover the secret why some of the tags are widely annotated while are not statistically significant or even do not appear in the summaries.

In most cases, the information in summaries is too deficient for users to tag resources or to reflect personalities. Users thus exploit their own knowledge, which may be partly learnt from other resource entries containing both summaries and tags in Table 1. Then when they want to tag an online resource, they will freely associate meaningful words in the summary with other semantic related words learnt from former reading experiences. However, the result of this association behavior will be explosive. Users should judge and weigh these candidate tags in brain, usually via forming a semantic related word network and finally decide the tags that they choose to annotate the given resource.

For example, after browsing plentiful of summary-tag pairs, we could naturally acquire the semantic relationships between the words, such as "singer", "pop", in the summary and the tag, "dance". If we tag the artist entry in Figure 1, the tag "dance" is more likely associated by the words like "pop", "artist", "Rock & Roll" et al. While reading the summary of artist Michael Jackson in Figure 1, we may construct an abstract tag-network in Figure 2 with the important words (king, pop, artist et al.) in the summary, the associated tags (dance, 80s, pop et al) and their semantic relationships.

Summary: David Lindgren (born April 28, 1982 in Skelleftea, Sweden) is a Swedish ***singer*** and musical artist…
Tags: swedish, pop, ***dance***, musical, david lindgren
Summary: Wanessa Godói Camargo (born on

December 28, 1982), known simply as Wanessa, is a Brazilian ***pop singer***…	
Tags: pop, ***dance***, female vocalists, electronic, electropop …	

Table 1: Examples of artist entries from Last.fm[2]

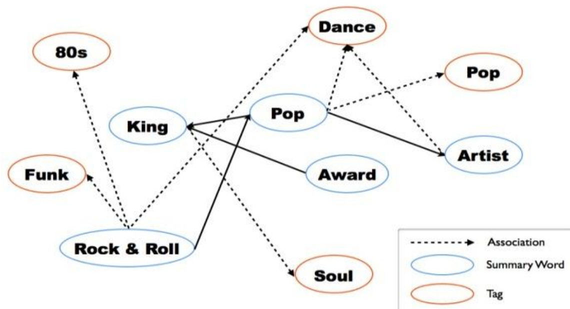

Figure 2: A part of the abstract associative tag-network in human brains.

3 Associative Tag Recommendation

We describe our ATR approach as a three-stage procedure by simulating the human annotation process analyzed in Section 2. Figure 3 shows the overall structure of our approach.

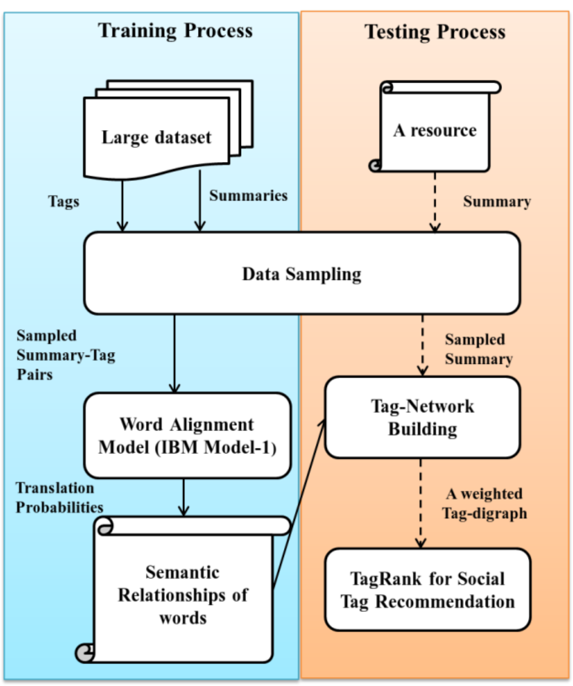

Figure 3: The overview of ATR approach.

Stage 1: Summary-tag pairs sampling. Given a large collection of tagged resources, we need to pre-process the dataset. Generally, the pre-processing contains tokenizing the summaries, extracting the meaningful words and balancing the length ratio between the summaries and tags.

Stage 2: Associative ability acquiring. We regard a summary-tag pair as a parallel text. They are really suitable to acquire the semantic relation knowledge by using word alignment model (In this paper, we adopt IBM Model-1) from the large amount of summary-tag pairs prepared by Stage 1. After gaining the translation probabilities between the meaningful words in summaries and tags, our social tagging recommender system initially has the capability of association, namely from one word to many semantic related tags.

Stage 3: TagRank algorithm for recommendation. Stage 2 just helps our recommender system acquire the ability of associating one word with many semantic related tags. However, when the system faces a given resource with a long summary, the association results may be massive. Thus, we propose a TagRank algorithm to order the candidate tags on the weighted Tag-digraph, which is built by the meaningful words in the summary and their semantic related words.

Before introducing the approach in details, we define some general notations, while the other specific ones will be introduced in the corresponding stage. In our approach, a resource is denoted as $r \in R$, where R is the set of all resources. Each resource contains a summary and a set of tags. The summary s_r of resource is simply regarded as a bag of meaningful words $w_r = \{(w_i, cw_i)\}_{i=1}^{N_r}$, where cw_i is the count of meaningful word w_i and N_r is the number of the unique meaningful words in r. The tag set (annotations) a_r of resource r is represented as $a_r = \{(t_i, ct_i)\}_{i=1}^{M_r}$, where ct_i is the count of tag t_i and M_r is the number of the unique tags for r.

3.1 Summary-Tag Pairs Sampling

We consider that the *nouns* and *tags* that appear in the corresponding summary are *meaningful* for our tagging recommendation approach.

It is not difficult for language, such as English, French et al. As for Chinese, Thai and Japanese, we still need to do word segmentation (D. D. Palmer., 2010). Here, to improve the segmentation results of these language texts, we collect all the unique tags in resource r as the user dictionary to solve the out-of-vocabulary issue. This idea is inspired by M. Sun (2011) and we will discuss its effort on the performance improvement of our system in Section 4.3.

After the meaningful words have been extracted from the summaries, we regard the summary and the set of tags as two bags of the sampled words without position information for a given resource. The IBM Model-1(Brown et al., 1993) was adopted for training to gain the translation probabilities between the meaningful words in summary and the tags. Och and Ney (2003) proposed that the performance of word alignment models would suffer great loss if the length of sentence pairs in the parallel training data set is unbalanced. Moreover, some popular online resources may be annotated by hundreds of people with thousands of tags while the corresponding summaries may limit to hundreds of words. So, it is necessary to propose a sampling method for balanced length of summary-tag pairs.

One intuitional way is to assign each meaningful word in summaries and tags with a term-frequency (TF) weight, namely cw_i and ct_i . For each extracted meaningful word w_i in a given summary s_r , $TF_{w_i}^{\mathrm{s}_r} = \frac{cw_i}{\sum_{i=1}^{N_r} cw_i}$ and the same tag set (annotations) a_r , $TF_{t_i}^{\mathrm{a}_r} = \frac{ct_i}{\sum_{i=1}^{M_r} ct_i}$. Here, we bring a parameter δ in this stage, which denotes the length ratio between the sampled summary and tag set, namely, $\delta = N_r / M_r$

3.2 Associative Ability Acquiring

IBM Model-1 could help our social tagging recommender system to learn the lexical translation probability between the meaningful words in summaries and tags based on the dataset provided by stage 1. We adjust the model to our approach, which can be concisely described as,

$$\Pr(\mathrm{W}_r \mid \mathrm{T}_r) = \sum_{A} \Pr(W_r, A | T_r) \qquad (1)$$

For each resource r, the relationship between the sampled summary $\mathrm{W}_r = \{w_i\}_{i=1}^{N_r}$ and the sampled tags $T_r = \{t_i\}_{i=1}^{M_r}$ is connected via a hidden variable $A = \{a_i\}_{i=1}^{N_r}$. For example, $a_j = i$ indicates word w_j in W at position j is aligned to tag t_i in T at position i.

For more detail description on mathematics, the joint likelihood of W_r and an alignment A given T_r is

$$\Pr(W_r, A \mid \mathrm{T}_r) = \frac{\varepsilon}{(M_r+1)^{N_r}} \prod_{j=1}^{N_r} p\left(w_j \mid t_{a_j}\right) \qquad (2)$$

in which $\varepsilon \equiv \Pr(N_r \mid \mathrm{T}_r)$ and $p\left(w_j \mid t_{a_j}\right)$ is called the translation probability of w_j given t_{a_j}. The alignment is determined by specifying the values of a_j for j from 1 to N_r, each of which can take any value from 0 to M_r. Therefore,

$$\Pr(W_r \mid \mathrm{T}_r) = \frac{\varepsilon}{(M_r+1)^{N_r}} \sum_{a_1=0}^{M_r} \cdots \sum_{a_{N_r}=0}^{M_r} \prod_{j=1}^{N_r} p\left(w_j \mid t_{a_j}\right) \qquad (3)$$

The goal is to adjust the translation probabilities so as to maximize $\Pr(W_r \mid \mathrm{T}_r)$ subject to the constraints that for each t,

$$\sum_{w} p(w|t) = 1 \qquad (4)$$

IBM Model-1 can be trained using Expectation-Maximization (EM) algorithm (Dempster et al., 1977) in an unsupervised fashion. At last, we obtain the translation probabilities between summaries and tags, i.e., $p(w|t)$ and $p(t|w)$ for our recommender system acquiring associative ability.

From Eq. (4), we know that IBM Model-1 will produce one-to-many alignments from one language to another language, and the trained model is thus asymmetric. Sometimes, there are a few translation pairs appear in both two direction, i.e., summary → tag (p_{s2t}) and tag → summary (p_{t2s}). For this reason, Liu et al. (2011) proposed a harmonic means to combine the two models.

$$p(t|w) = \left(\frac{\lambda}{p_{t2s}(t|w)} + \frac{1-\lambda}{p_{s2t}(t|w)}\right)^{-1} \quad (5)$$

3.3 TagRank Algorithm for Recommendation

By the time we have generated the “harmonic” translation probability list between meaningful words in summaries and tags, our recommender system could acquire the capability of association like human beings. For instance, it could “trigger” a large amount of semantic related tags from a given word: *Novel* (Figure 4). However, if we collected all the “triggered” tags associated by each meaningful word in a given summary, the scale would be explosive. Thus we need to explore an efficient way that can not only rank these candidate tags but also simulate the human tagging behavior as much as possible.

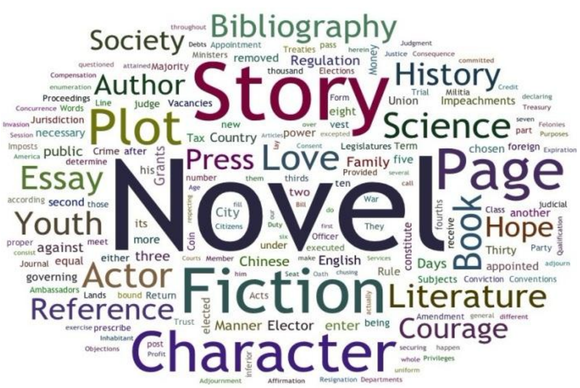

Figure 4: The association results from the word “Novel” via our social tagging recommender system.

Inspired by the PageRank algorithm (S. Brin and L. Page., 1998), we find out that the idea could be brought into our approach with a certain degree improvement as the human tagging ranking process is on a weighted Tag-digraph G. We regard the association relationship as one word recommending the corresponding candidate tags and the degree of preference could be quantified by the translation probabilities.

For a given summary, we firstly sample it via the method described in stage 1 to obtain all the meaningful words, which are added to the graph as a set of seed vertices denoted as V_s. Then according to stage 2, we could obtain a set of semantic related vertices associated by these seeds denoted as V_a. We union the V_a and V_s to get the set of all candidate tags V. For a directed edge e_{ij} from v_i to v_j, the weight $w(e_{ij})$ equals the translation probability from v_i to v_j, namely $p(v_j|v_i)$. So the weighted Tag-digraph could be formulized as,

$$\begin{cases} G = (V, E) \\ V = V_s \cup V_a \\ E = \{e_{ij}\} \\ e_{ij} = \{(v_i, v_j), v_i, v_j \in V\} \\ w(e_{ij}) = p(v_j|v_i) \end{cases} \quad (6)$$

The original TextRank algorithm (Mihalcea et al., 2004) just considered the words recommending the nearest ones, and assumed that the recommending strengths were same. As all the words had the equal chance to recommend, it was the fact that all the edges in the graph gained no direction information. So this method brought little improvement on ranking results. In the Eq. (7) they used, $In(v_i)$ represents the set of all the vertices that direct to v_i and $Out(v_j)$ denotes the set of all the vertices that direct from v_j. The factor d is usually set to 0.85.

$$\mathrm{Score}(v_i) = (1-d) + d * \sum_{v_j \in In(v_i)} \frac{1}{|Out(v_j)|} \mathrm{Score}(v_j) \quad (7)$$

We improve the TextRank model and propose a TagRank algorithm (Eq. 8) that is suitable to our approach. For each v_j, $\frac{w(e_{ji})}{\sum_{v_k \in Out(v_j)} w(e_{jk})}$ represents the proportion of trigger ability from v_j to v_i. This proportion multiplying the own score of v_j reflect the the degree of recommend contribution to v_i. After we sum up all the vertices willing to “recommend” v_i, namely $v_j \in In(v_i)$, We can calculate the score of v_i in one step.

Some conceptual words could trigger hundreds of tags, so that our recommender system will suffer a rather high computation complexity. Thus, we add a parameter θ which stands for the maximum out-degree of the graph G. That means for each vertex in the graph G, it can at most trigger top-θ candidate tags with the θ highest translation probabilities.

$$\mathrm{Score}(v_i) = (1-d) + d * \sum_{v_j \in In(v_i)} \frac{\mathrm{w}(\mathrm{e}_{ji})}{\sum_{v_k \in Out(v_j)} \mathrm{w}(\mathrm{e}_{jk})} \mathrm{Score}(v_j) \tag{8}$$

Starting from *vertex initial values* assigned to the seed nodes (V_s) in the graph, the computation iterates until convergence below a given threshold is achieved. After running the algorithm, a score is assigned to each vertex. Finally, our system can recommend best M tags with high score for the resource.

4 Experiments

4.1 Datasets and Evaluation Metrics

Datasets: We prepare two real world datasets with diverse properties to test the performance of our system in different language environment. Table 2 lists the statistical information of the English and Chinese datasets.

Dataset	P	V_s	V_t	N_s	N_t
BOOK	29464	68996	40401	31.5	7.8
ARTIST	14000	35972	4775	19.0	5.0

Table 2: Statistical information of two datasets. P , V_s , V_t , N_s, and N_t represent the number of parallel texts, the vocabulary of summaries, the vocabulary of tags, the average number of unique words in each summary and the average number of unique tags in each resource respectively.

The first dataset, BOOK, was crawled from a popular Chinese book review online community Douban[4], which contains the summaries of books and the tags annotated by users. The second dataset, ARTIST, was freely obtained via the Last.fm[2] API. It contains the descriptions of musical artists and the tags annotated by users. By comparing the characteristics of these two datasets, we find out that they differ in language, data size and the length ratio (Figure 5). The reason of preparing two datasets with diverse characteristics is that we would like to demonstrate that our approach is effective, robust and language-independent compared with others.

Evaluation Metrics: We use precision, recall and F-measure to evaluate the performance of our ATR approach. Given a resource set R, we regard the set of original tags as T_O, the automatic recommended tag set as T_R. The correctly recommended set of tags can be denoted as $T_R \cap T_O$. Thus, precision, recall and F-measure are defined as[5]

$$\mathrm{p} = \frac{|T_R \cap T_O|}{|T_R|}, \mathrm{r} = \frac{|T_R \cap T_O|}{|T_O|}, F = \frac{2\,p\,r}{(p+r)} \tag{9}$$

The final precision and recall of each method is computed by performing 7-fold cross validation on both two datasets.

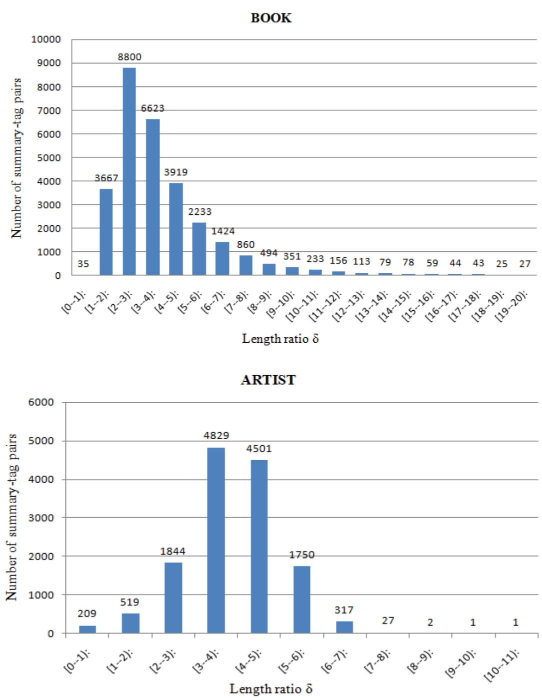

Figure 5: The length ratio distributions of BOOK and ARTIST datasets.

4.2 Methods Comparison

Baseline Methods: In this section, we compare the performance of our associative tagging recommendation (ATR) with three other relative methods, the state-of-the-art WTM (Liu et al., 2011), TextRank (Mihalcea et al., 2004) and the traditional TFIDF (C. D. Manning et al., 2008; R. Baeza-Yates et al., 2011).

[5] The reason why we do not calculate the precision, recall and F-measure alone is that we cannot guarantee that recommending at least one correct tag for each resource.

The reasons we choose those methods to compare were as follows.

- WTM can reflect the state-of-the-art performance on *content-based* social tag recommendation.
- TextRank can be regarded as a baseline method on graph-based social tag recommendation.
- TFIDF, as a traditional method, represents the baseline performance and can validate the "out-of-summary" phenomenon.

For the TFIDF value of each word in a given summary, it can be calculated by multiplying term frequency $TF_{w_i}^{s_r} = 1 + \log\frac{cw_i}{\sum_{i=1}^{N_r} cw_i}$ (log normalization) by inverted document frequency $IDF_{w_i}^{s_r} = \log(1 + \frac{|R|}{\left|\sum_{r \in R} I_{cw_i > 0}\right|})$ (inverse frequency smooth), where $\left|\sum_{r \in R} I_{cw_i > 0}\right|$ indicates the number of resources whose summaries contain word w_i.

TextRank method regarded the word and its forward and backward nearest words as its recommendation. Thus, each word in a given summary is recommended by its neighborhood with no weight. Simply, we use Eq. (7) to calculate the final value of each word in a given summary.

Liu et al. (2011) proposed a state of the art method which summed up the product the weight of a word and its translation probabilities to each semantic related tag as the final value of each tag in a given resource (Eq. 10).

$$\Pr(\mathrm{t}|\mathrm{w}_r) = \sum_{w \in \mathrm{w}_r} \Pr(t|w)\Pr(w|\mathrm{w}_r) \qquad (10)$$

Experiment Results: Figure 6 illustrates the precision-recall curves of ATR, WTM, TextRank and TFIDF on two datasets. Each point of a precision-recall curve stands for different number of recommended tags from M = 1 (upper left) to M = 10 (bottom right). From the Figure 6, we can observe that:

- ATR out-performs WTM, TextRank and TFIDF on both datasets. This indicates that ATR is a language-independent approach for social tag recommendation.
- ATR shows consistently better performance when recommending different number of tags, which implies that our approach is efficient and robust (Figure 7).

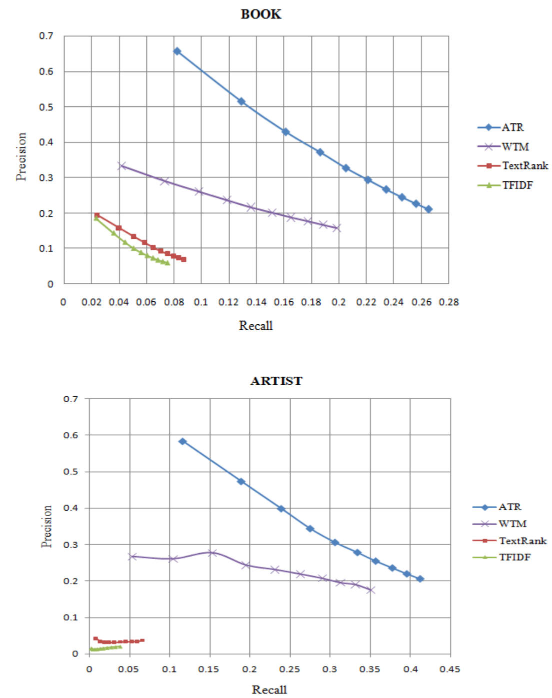

Figure 6: Performance comparison among ATR, WTM, TextRank and TFIDF on BOOK and ARTIST datasets when $\lambda = 0.5$, $\theta = 5$ and vertex initial values are assigned to one.

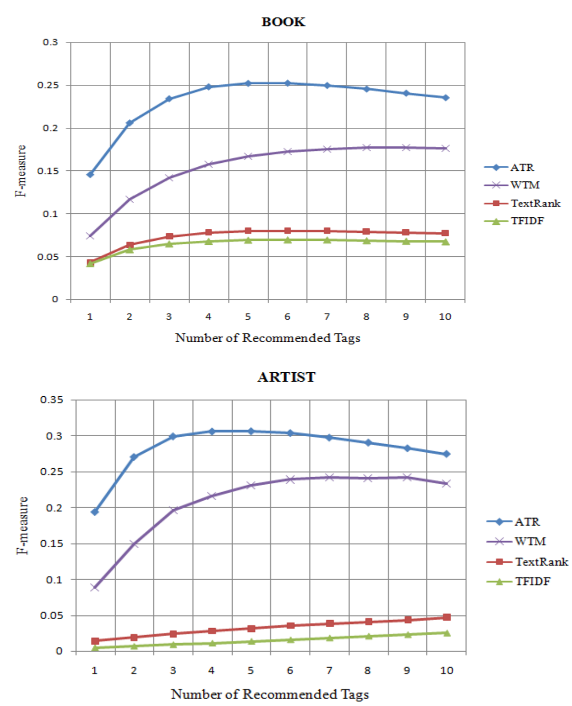

Figure 7: F-measure of ATR, WTM, TextRank and TFIDF versus the number of recommended tags (M) on the BOOK and ARTIST datasets when $\lambda = 0.5, \theta = 5$ and vertex initial values are assigned to one.

4.3 Sampling Methods Discussion

Section 3.1 proposed an idea on summary-tag pairs sampling, which collected all the unique tags as the user dictionary to enhance performance of the summary segmentation, especially for the Chinese, Thai, and Japanese et al. Though M. Sun (2011) put forward a more general paradigm, few studies have verified his proposal. Here, we will discuss the efficiency of our sampling method. Figure 8 shows the comparison of performance between the unsampled ATR and (sampled) ATR.

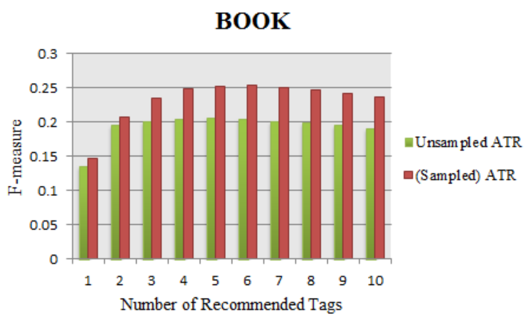

Figure 8: Performance comparison between unsampled ATR and (sampled) ATR on BOOK datasets when $\lambda = 0.5, \theta = 5$ and vertex initial values are assigned to one

Experiments on the Chinese dataset BOOK demonstrates that our (sampled) ATR approach achieves average 19.2% improvement on performance compared with the unsampled ATR.

4.4 Parameter Analysis

In Section 3, we brought several parameters into our approach, namely the harmonic factor λwhich controls the proportion between model p_{t2s} and p_{s2t}, the maximum out-degree θ which specifies the computation complexity of the weighted tag-digraph and the vertex initial values which may affect the final score of some vertices if the weighted tag-digraph is not connected.

We take the BOOK dataset as an example and explore their influences to ATR by using controlling variables method, which means we adjust the focused parameter with the other ones stable to observe the results.

Harmonic factor: In Figure 9, we investigate the influence of harmonic factor via the curves of F-measure of ATR versus the number of recommended tags on the BOOK dataset. Experiments showed that the performance is slightly better when $\lambda = 0.0$. As λ controls the proportion between model p_{t2s} and p_{s2t}, $\lambda = 0.0$ means model p_{t2s} contributes more on performance.

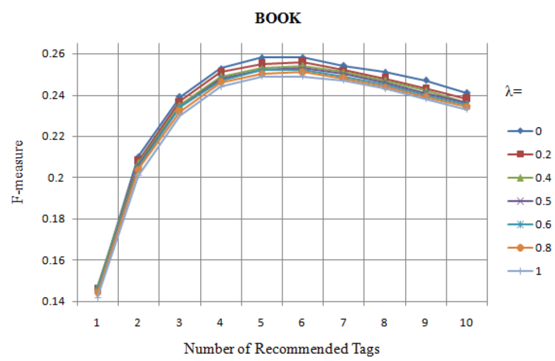

Figure 9: F-measure of ATR versus the number of recommended tags on the BOOK dataset when harmonic factor λ ranges from 0.0 to 1.0, when $\theta = 5$ and vertex initial values are assigned to one.

Maximum out-degree: Actually, during the experiments, we have found out that some meaningful words could trigger hundreds of candidate tags. If we bring all these tags to our Tag-Network, the computation complexity will be dramatically increased, especially in large datasets. To decrease the computation complexity with little impact on performance, we need to explore the suitable maximum out-degree. Figure 10 illustrates how the complexities of tag-digraph will influent the performance. We discover that ATR gains slight improvement when θ is added from 5 to 9 except the "leap" from 1 to 5. It means that $\theta = 5$ will be a suitable maximum out-degree, which balances the performance and the computation complexity.

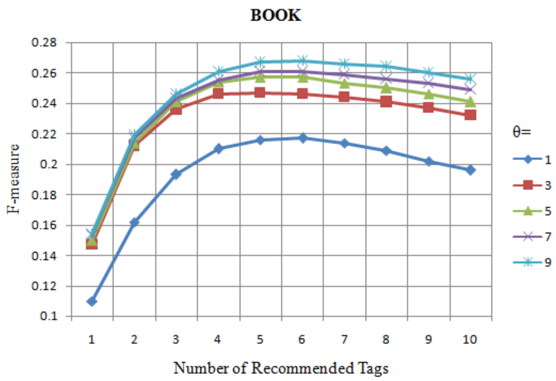

Figure 10: F-measure of ATR versus the number of recommended tags on the BOOK dataset, when $1 \leq \theta \leq 9, \lambda = 0.2$ and vertex initial values are assigned to one.

Vertex initial values: The seeds (meaningful words in the summaries) may not be semantic related, especially when the maximum out-degree is low. As a result, the graph G may be disconnected, so that the final score of each vertex after iteration may relate to the vertex initial values. In Figure 11, we compare three different vertex initial values, namely value-one, value of TF (local consideration) and value of TFIDF (global consideration) to check the influence. However, the results show that there is almost no difference in F-measure when the maximum out-degree θ ranges from 1 to 9.

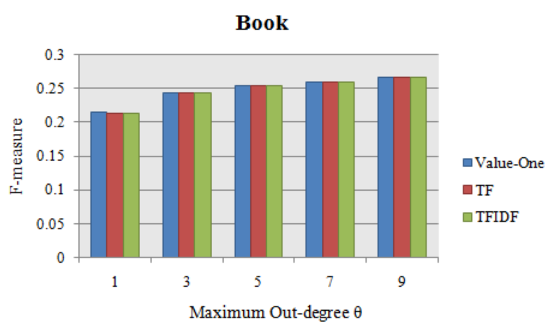

Figure 11: F-measure of ATR versus maximum out-degree on BOOK dataset when the vertex initial values equal to Value-One, TF, TFIDF separately with $\lambda = 0.2$ and number of recommended tags M = 5.

5 Related Work

There are two main stream methods to build a social tag recommender system. They are collaboration-based method (Herlocker et al., 2004) and the content-based approach (Cantador et al., 2010).

FolkRank (Jaschke et at., 2008) and Matrix Factorization (Rendle et al., 2009) are representative collaboration-based methods for social tag recommendation. Suggestions of these techniques are based on the tagging history of the given resource and user, without considering the resource summaries. Thus most of these methods suffer from the cold-start problem, which means they cannot perform effective suggestions for resources that no one has annotated.

To remedy the defect of cold-start problem, researchers proposed content-based methods exploiting the descriptive information on resources, such as summaries. Some of them considered social tag recommendation as a classification problem by regarding each tag as a category label. Various classifiers such as kNN (Fujimura et al., 2007), SVM (Cao et al., 2009) have been discussed. But two issues exposed from these methods.

- Classification-based methods are highly constrained in the quality of annotation, which are usually noisy.
- The training and classification cost are often in proportion to the number of classification labels, so that these methods may not be efficient for real-world social tagging system, where thousands of unique tags may belong to a resource.

With the widespread of latent topic models, researchers began to pay close attention on modeling tags using Latent Dirichlet Allocation (LDA) (Blei et al., 2003). Recent studies (Krestel et al., 2009; Si and Sun, 2009) assume that both tags and words in summary are generated from the same set of latent topics. However, most latent topic models have to pre-specify the number of topic before training. Even though we can use cross validation to determine the optimal number of topics (Blei et al., 2010), the solution is obviously computationally complicated.

The state of the art research on social tagging recommendation (Z. Liu, X. Chen and M. Sun, 2011) regarded social tagging recommendation problem as a task of selecting appropriate tags from a controlled tag vocabulary for the given resource and bridged the vocabulary gap between the summaries and tags using word alignment models in statistical machine translation. But they simply adopted the weighted sum of the score of candidate tags, named word trigger method (WTM), which cannot reflect the whole process of human annotation.

6 Conclusion and Future Work

In this paper, we propose a new approach for social tagging recommendation via analyzing and modeling human associative annotation behaviors. Experiments demonstrate that our approach is effective, robust and language-independent compared with the state of the art and baseline methods.

The major contributions of our work are as follows.

- The essential process of human tagging process is discovered as the guideline to help us build simulating models.
- A suitable model is proposed to assist our social tagging recommender system to learn the semantic relationship between the meaningful words in summaries and corresponding tags.
- Based on the semantic relationship between the meaningful words in the summaries and corresponding tags, a weighted Tag-digraph is constructed. Then a TagRank algorithm is proposed to re-organize and rank the tags.

Our new approach is also suitable in the tasks of keyword extraction, query expansion et al, where the human associative behavior exists. Thus, we list several open problems that we will explore in the future:

- Our approach can be expanded from lexical level to sentence level to bring the associative ability into semantic-related sentences extraction.
- We will explore the effects on other research areas, such as keyword extraction, query expansion, where human associative behavior exists as well.

Acknowledgements

The work was supported by the research projects of National Science Foundation of China (Grant No. 60873173) and National 863 High-Tech research projects (Grant No. 2007AA01Z173). The authors would like to thank Yi Luo for his insightful suggestions.

References

R. Baeza-Yates and B. Ribeiro-Neto. 2011. *Modern information retrieval: the concepts and technology behind search, 2nd edition*. ACM Press.

D. M. Blei, A. Y. Ng, and M. I. Jordan. 2003. Latent dirichlet allocation. JMLR, 3:993-1022.

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

P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263-311.

I. Cantador, A. Bellog ń, D. Vallet. 2010. Content-based recommendation in social tagging systems. *In Proceedings of ACM RecSys*, pages 237-240.

H. Cao, M. Xie, L. Xue, C. Liu, F. Teng, and Y. Huang. 2009. Social tag predication base on supervised ranking model. *In Proceeding of ECML/PKDD 2009 Discovery Challenge Workshop*, pages 35-48.

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

D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green. 2007. Automatic generation of social tags for music recommendation. *In Proceedings of NIPS*, pages 385-392.

S. Fujimura, KO Fujimura, and H. Okuda. 2007. Blogosonomy: Autotagging any text using bloggers' knowledge. *In Proceedings of WI*, pages 205-212.

J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1):5-53.

R. Jaschke, L. Marinho, A. hotho, L. Schmidt-Thieme, and G. Stumme. 2008. Tag recommendations in social bookmarking systems. *AI Communications*, 21(4):231-247.

R. Krestel, P. Fankharser, and W. Nejdl. 2009. Latent dirichlet allocation for tag recommendation. *In Proceedings of ACM RecSys*, pages 61-68.

Z. Liu, X. Chen, M. Sun. 2011. A simple word trigger method for social tag suggestion. *In Proceedings of EMNLP*, pages 1577-1588.

C. D. Manning. P. Raghavan, and H. Schtze. 2008. *Introduction to information retrieval*. Cambridge University Press, NY, USA.

R. Mihalcea and P. Tarau. 2004. TextRank: Bringing order into texts. *In Proceedings of EMNLP*, pages 404-411. Poster.

R. Mirizzi, A. Ragone, T. Di Noia, and E. Di Sciascio. 2010. Semantic tags generation and retrieval for online advertising. *In Proceedings of CIKM*, pages 1089-1098.

C. Musto, F. Narducci, M. de Gemmis, P. Lops, and G. Semeraro. 2009. STaR: a social tag recommender system. *In Proceeding of ECML/PKDD 2009 Discovery Challenge Workshop*, pages 215-227.

F. J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1): 19-51.

D. D. Palmer. 2010. Text preprocessing. *Handbook of natural language processing 2nd edition*, chapter 2. CRC Press.

S. Rendle, L. Balby Marinho, A. Nanopoulos, and L. Schmidt-Thieme. 2009. Learning optimal ranking with ensor factorization for tag recommendation. *In Proceedings of KDD*, pages 727-726.

F. Ricci, L. Rokach, B. Shapira and P. B. Kantor. 2011. *Recommender Systems Handbook*. Springer Press.

X. Si and M. Sun. 2009. Tag-LDA for scalable real-time tag recommendation. *Journal of Computational Information Systems*, 6(1): 23-31.

M. Sun. 2011. Natural language processing based on naturally annotated web resources. *Journal of Chinese Information Processing*, 25(6): 26-32

T. C. Zhou, H. Ma, M. R. Lyu, and I. King. 2010. UserRec: A user recommendation approach in social tagging systems. *In Proceedings of AAAI*, pages 1486-1491.