Content-based Semantic Tag Ranking for Recommendation

Miao Fan, Qiang Zhou and Thomas Fang Zheng
Center for Speech and Language Technologies, Division of Technical Innovation and Development, Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University
Beijing 10084, China
fanmiao.cslt.thu@gmail.com, zq-lxd@mail.tsinghua.edu.cn, fzheng@tsinghua.edu.cn

***Abstract*—Content-based social tagging recommendation, which considers the relationship between the tags and the descriptions contained in resources, is proposed to remedy the cold-start problem of collaborative filtering. There is such a common phenomenon that certain tag does not appear in the corresponding description, however, they do semantically relate with each other. State-of-the-art methods seldom consider this phenomenon and thus still need to be improved. In this paper, we propose a novel content-based social tag ranking scheme, aiming to recommend the semantic tags that the descriptions may not contain. The scheme firstly acquires the quantized semantic relationships between words with empirical methods, then constructs the weighted tag-digraph based on the descriptions and acquired quantized semantics, and finally performs a modified graph-based ranking algorithm to refine the score of each candidate tag for recommendation. Experimental results on both English and Chinese datasets show that the proposed scheme performs better than several state-of-the-art content-based methods.**

Keywords-social tagging; recommender system; ranking

I. INTRODUCTION

Recent years have witnessed an increasing growth of Web 2.0 applications. Social Tagging Systems (STS for short), known as a particular family of Web 2.0 applications [21], assign a major role to the ordinary user, who is not only allowed to publish and edit resources, but also and more importantly, to create and share lightweight metadata in the form of freely chosen keywords called *tags* [22].

Some notable examples of STS are sites like *Flickr*, *Del.icio.us*, *Last.fm* and *Douban*. Though STS encourages users to participate in tagging resources, it revives an old problem of information overload. Recommender Systems (RS for short) thus are brought in as an indispensable component to increase the level of relevant information over the "noise" that continuously grows as more and more resources become available in STS. When a user plans to add tags to a resource that he/she is interested in, social tagging recommender system suggests a list of tags that the user most possibly chooses.

In order to build effective social tagging recommender system, two main stream methods are proposed, which are collaborative filter (CF) and content-based. CF methods [24] generally rely on the tagging history of the given resource and user, and recommend the tags from the resource and user with the highest similarity. However, these methods expose the cold-start problem [7] that new resource without annotation cannot be recommended tags via CF methods effectively as no profiles can be considered to measure the similarity with other resources. Content-based methods [5, 11, 13, 23] thus are proposed to remedy the issue and attract more and more literatures in recent years. For a given resource, the methods focus on extracting tags from its content feature, which ignore the influence of its own popularity. Nevertheless, some of the tags do not appear in while semantically relate with the corresponding content description. State-of-the-art content-based methods seldom consider this phenomenon, so that the performance of these methods can be improved via expanding to the level of semantics.

In this paper, we propose a novel content-based tag ranking scheme, aiming to recommend semantic tags that descriptions may not contain. The scheme follows the typical "learning-prediction" paradigm and is fine grained into three steps.

Learning Step:

1. Acquiring the quantized semantic relationships between the terms [1] in resource descriptions and corresponding tags on large datasets with empirical methods.

Prediction Steps:

2. For a given resource, constructing the tag-digraph based on the acquired quantized semantics and its key terms in the description.
3. Performing a modified graph-based ranking algorithm to refine the score of each candidate tag (vertex) and selecting the ones with highest scores for recommendation.

II. GENERAL NOTATIONS

In the *Learning Step*, an online resource set is denoted as R. Every resource $\mathrm{r} \in \mathrm{R}$ can be represent as a binary tensor $\mathrm{r} = (\mathrm{d}_r, \mathrm{a}_r)$. d_r represents a bag of terms extracted from the description of r including nouns and tags. It is formally defined as $\mathrm{d}_r = \{(w_i, cw_i)\}_{i=1}^{N_r}$, where cw_i denotes the count of term w_i and N_r represents the number of unique

[1] To distinguish the nouns and tags in the description, we generally define these words as *terms* in this paper.

978-0-7695-4880-7/12 $26.00 © 2012 IEEE
DOI 10.1109/WI-IAT.2012.32

terms in d_r. Meanwhile, a_r represents a bag of tags and is defined as $a_r = \{(t_j, ct_j)\}_{j=1}^{M_r}$, where ct_j [2] denotes the count of tag t_j and M_r represents the number of unique tags in a_r. The sampled d_r and a_r are denoted as $W_r = \{w_i\}_{i=1}^{L_r}$ and $T_r = \{t_j\}_{j=1}^{L_r}$ respectively.

In the *Prediction Steps*, every resource without annotation is defined as r'. The terms appearing in the corresponding description d_r' are regarded as the seed vertex set $Vs_r' = \{w_i'\}_{i=1}^{N_r'}$ to construct the weighted tag-digraph G'. The semantic tags are represented as a vertex set $Vt_r' = \{t_j'\}_{j=1}^{M_r'}$. e_{ij} represents a directed edge from v_i to v_j and the weight of e_{ij} is defined as $w(e_{ij})$. We regard the vertices that direct to the vertex v_i as a set denoted as $In(v_i)$ and the vertices that the vertex v_i directs to as a set denoted as $Out(v_i)$.

III. SEMANTIC TAG RANKING SCHEME

Figure 1 shows the overview of our proposed STR scheme. It contains three main components, which are named as *empirical semantic acquisition* (Section 3.1), *content-based tag network construction* (Section 3.2) and *candidate tag ranking algorithm* separately (Section 3.3).

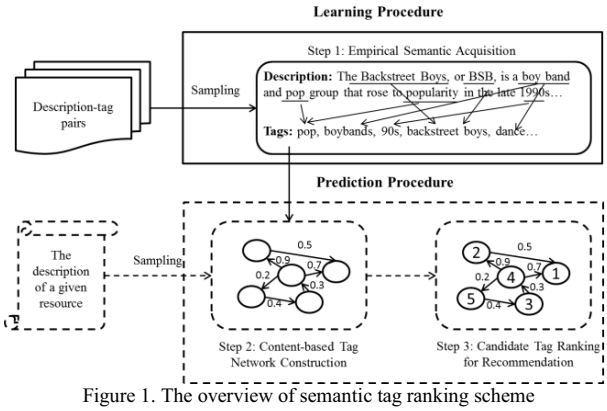

Figure 1. The overview of semantic tag ranking scheme

A. Empirical Semantic Acquisition

Section 1 exposes the phenomenon that the terms in the description semantically relate with the corresponding tags for a given resource. We call it *Semantic Tag Annotation Phenomenon* (STA). For more details, in Figure 2, we can find a variety of semantic relationships, such as BSB is short for Backstreet Boys, pop may both mean popularity and pop group etc.

An intuitive way is to use knowledge-based dictionaries like WordNet [1] to extend the descriptions semantically. However, these dictionaries have three main shortcomings which do not fit for semantic tag recommendation.

- Most of the dictionaries only expert on semantic expansion of common words, while cannot deal with many new-emerge words in STS.
- Knowledge-based dictionaries are rather rigid and unable to keep up with updating of new words and semantics.
- The degrees of semantic relationships between words are fixed but in fact they change with the increasing of knowledge.

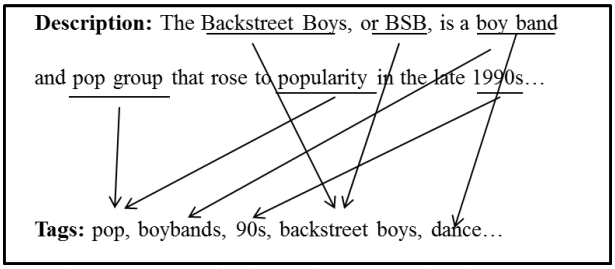

Figure 2. An example of semantic tag annotation phenomenon

To solve the issue above, we adopt IBM Model-1 [4], a word alignment model in statistical machine translation, as an approach to empirically acquire the quantized semantics. It was mostly used to bridge the vocabulary gap between two types of objects. We further find out that the most gaps can be bridged semantically. IBM Model-1 is thus adjusted to monolingual environment [15] and brought to STR scheme. The description-tag pairs are regarded as *parallel texts* for empirical semantic acquisition. As Och and Ney [19] proposed that performance of word alignment models would suffer great loss if the length of sentence pairs in the parallel training data set is unbalanced, we need to sample d_r (W_r) and a_r (T_r) to equal length with most important terms coming first, according to cw_i and ct_j respectively. The IBM Model-1 in STR scheme is concisely described as,

$$\Pr(T_r \mid W_r) = \sum_{A} \Pr(T_r, A \mid W_r) \quad (1)$$

For each resource r, the relationship between the sampled description $W_r = \{w_i\}_{i=1}^{L_r}$ and the sampled tags $T_r = \{t_j\}_{j=1}^{L_r}$ is connected via a hidden variable $A = \{a_i\}_{i=1}^{L_r}$. For example, $a_j = i$ indicates the tag t_j in T_r at position j is aligned to the term w_i in W_r at position i. The goal is to adjust the translation probabilities so as to maximize $\Pr(T_r \mid W_r)$ subject to the constraints that for each term,

$$\sum_{t} p(t|w) = 1 \quad (2)$$

In STR scheme, $p(t|w)$ is regarded as the semantic tag trigger ability from term w to semantically related tag t.

B. Content-based Tag Network Construction

This part is going to show how to construct a weighted Tag-digraph G' based on the description of a given resource r' without annotation, for the sake of providing a basis to perform our new proposed tag ranking algorithm in the next part.

[2] In our datasets, Chinese dataset Douban Book contains the times of annotation for each tag. Even though English dataset Last.fm Artist does not record this information, we can still obtain a descending order list of tags for each resource instead.

Firstly, the terms in the description of r' are regarded as seed vertex set $\mathrm{Vs}_r{}' = \{w_i{}'\}_{i=1}^{N_r{}'}$ and all the translation probabilities provide more semantic tags $t_j{}'$ triggered from $w_i{}'$. Then we collect all the semantic tags[3] to form a new vertex set $\mathrm{Vt}_r{}'$ and regard $\mathrm{Vs}_r{}'$ and $\mathrm{Vt}_r{}'$ as candidate tag vertex $\mathrm{V}_r{}'$ of r'.

$$\mathrm{V}_r{}' = \mathrm{Vs}_r{}' \cup \mathrm{Vt}_r{}' \quad (3)$$

Each translation probability $p(t|w)$ is expressed as a triple, $< \mathrm{w}, \mathrm{t}, p(t|w) >$. e_{ij} can be thus described as a directed edge from v_i to v_j, in which v_i comes from W ($\mathrm{w} \in W$) and v_j comes from T ($\mathrm{t} \in T$).

$$\mathrm{e}_{ij} = \{(v_i, v_j), v_i \in W, v_j \in T\} \quad (4)$$

And naturally,

$$\mathrm{w}(\mathrm{e}_{ij}) = p(v_i|v_j) \quad (5)$$

C. Candidate Tag Ranking Algorithm

After the TagNet is constructed for a given resource, each vertex is regarded as a candidate tag equally at first by assigned the same arbitrary initial score.

In this section, we propose a novel tag ranking algorithm to refine the score of each candidate tag inspired by the TextRank [17], which is a well-known ranking algorithm to extract keyphrase. It simply considered the *co-occurrence* relation as edge, controlled by the distance between word occurrences: two vertices are connected if their corresponding lexical unites co-occur with in a window of maximum N words. Even though the literature came up with a model (Eq. 6) applied to weighted graph, the weight of each edge ($\mathrm{WS}(v_i)$) was random in the interval 0-10, which seems unreasonable today. Moreover, the first term of Eq. 6 was set to be the same value $1 - \mathrm{d}$ for all vertices within the graph, which indicated there were equal probabilities of random jump to all vertices. However, this term can be non-uniformed as the terms in the description are barely same relevance to the main content of a given resource.

$$\mathrm{WS}(v_i) = (1-d) + d * \sum_{v_j \in In(v_i)} \frac{\mathrm{w}_{ji}}{\sum_{v_k \in Out(v_j)} \mathrm{w}_{jk}} \mathrm{WS}(v_j) \quad (6)$$

The basic idea of proposed tag ranking algorithm (Eq. 7) is that a vertex (candidate tag) is more important if there are other important vertices (candidate tags) pointing to it. This can be regarded as the voting behavior among human beings. Our algorithm uses the translation probability $p(v_i|v_j)$ to quantize the degree of voting from candidate tag v_j to v_i and differing from keyphrase extraction [14] more semantic tags which may not appear in the description can also be triggered. What's more, the offset (first term in Eq. 7) can be the normalized term frequency[4] of the corresponding candidate tag (vertex) and the final tag ranking scores will prefer these tags. λ is the damping factor range from 0 to 1 indicating the extent to which v_i depend on other semantic related vertices' ($v_j \in In(v_i)$) "*supportings*".

$$\mathrm{Score}(v_i) = (1-\lambda) * offset + \lambda * \sum_{v_j \in In(v_i)} \frac{\mathrm{w}(\mathrm{e}_{ji})}{\sum_{v_k \in Out(v_j)} \mathrm{w}(\mathrm{e}_{jk})} \mathrm{Score}(v_j) \quad (7)$$

Moreover, the proposed tag ranking algorithm will still work even if there is no semantic tag that can be triggered by the terms in the description, namely $\theta = 0$ or $\lambda = 0$. Under this condition, the algorithm can at least ensure that each term in the description could be scored by their normalized term frequency and ranked for recommendation.

Candidate tag ranking is an iterative algorithm. Starting from the same arbitrary score (usually 1) assigned to each vertex in the graph, the computation iterates until convergence below a given threshold (The threshold is set to 10^{-9} in this paper) is achieved. After running the algorithm, a final score is assigned to each vertex. According to these final scores, top M tags with high score are recommended to the target resource.

IV. EXPERIMENTS

A. Datasets

We prepare two real world datasets with diverse properties to evaluate our scheme. Table 1 shows the detailed statistical information of the two datasets.

TABLE I. STATISTICAL INFORMATION OF TWO DATASETS

Dataset	R	$\mathrm{V_d}$	$\mathrm{V_t}$	$\mathrm{N_d}$	$\mathrm{N_t}$
BOOK	25000	62680	32437	31.5	9.2
ARTIST	12000	35466	4156	19.0	5

R, $\mathrm{V_d}$, $\mathrm{V_t}$, $\mathrm{N_d}$, and $\mathrm{N_t}$ represent the number of resources, the vocabulary of descriptions, the vocabulary of tags, the average number of unique words in each description and the average number of unique tags in each resource respectively.

BOOK dataset is crawled via Douban API from a well-known Chinese book review website Douban, which posts the descriptions of books and collects the tags collaboratively annotated by users. It contains 25,000 unique book entries with descriptions and corresponding social tags. The second dataset, denoted as ARTIST, is obtained via Last.fm API from Last.fm, where music and artist entries are published and users can freely tag his/her interested resources. ARTIST contains 12000 unique artists with summaries (descriptions) and corresponding tags in English. The reason that we prepare two datasets with

[3] Given the performance of STR scheme in real-world STS, we need to constrain the number of tags triggered from terms in the corresponding description with topic-θ translation probabilities. Formally, θ thus denotes the maximum out-degree of the constructed TagNet.

[4] The NTF (normalized term frequency) of the corresponding candidate tag (vertex) is calculated by the formula $\mathrm{NTF}(v_i) = \frac{TF(v_i)}{\sum TF(v_j)}$

different language environment and statistical information is to prove our scheme can out-perform state-of-the-art methods in the following experiments.

B. Evaluation Metrics

We use precision, recall and F-measure to evaluate the performance of our STR scheme. Given a resource set R, we regard the set of tags that annotated by users T_U as the *gold standard*, the automatic recommended tag set as T_R. The correctly recommended set of tags can be denoted as $T_R \cap T_U$. Thus, precision, recall and F-measure are defined as

$$p = \frac{|T_R \cap T_U|}{|T_R|}, r = \frac{|T_R \cap T_U|}{|T_U|}, F = \frac{2\,p\,r}{(p + r)} \quad (8)$$

The final precision and recall of each method is computed by performing 5-fold cross validation on both two datasets. F-measure is the comprehensive evaluation.

C. Methods Comparison

1) Baseline Methods: We choose three other relative content-based social tagging recommendation methods to compare with our proposed STR scheme under the same datasets in Section 4.1 and evaluate the performances via the metrics in Section 4.2. Besides STR scheme, the three other methods are the state-of-the-art WTM [13], TextRank [17] and TFIDF [2,16].

The reasons that we choose these methods are listed as follows.

- WTM performs the best as the state-of-the-art content-based social tagging recommendation method. Our STR scheme should demonstrate its effectiveness and efficiency via comparing with WTM under the same experimental environment.
- TextRank is the well-known graph-based ranking algorithm to extract keyphrase, which also inspires STR. It is an undoubtable fact that TextRank is the ancestor being listed.
- TFIDF is the most successful approach that is widely applied to IR, keypharse extraction etc. to extract important information from texts. It is regarded as the baseline method to check the performance on social tagging recommendation.

2) Experiment Results: For WTM, TextRank and TFIDF, we set their best performance parameters according to their respective articles. And for our STR scheme, we random set the damping factor (λ) to 0.85 and maximum out-degree (θ) to 10, which did not fully stand for the best performance of STR.

Figure 3 shows the precision-recall curves of STR, WTM, TextRank and TFIDF on BOOK and ARTIST datasets. Each point from upper left to bottom right represents the number of recommended tag(s) ranging from 10 to 1 respectively. Moreover, Figure 4 illustrates the F-measures of those four tested methods when suggesting different number of tags. From those two figures, we can observe that STR scheme apparently out-performs WTM, TextRank and TFIDF on different language environments even though different number of tags is suggested.

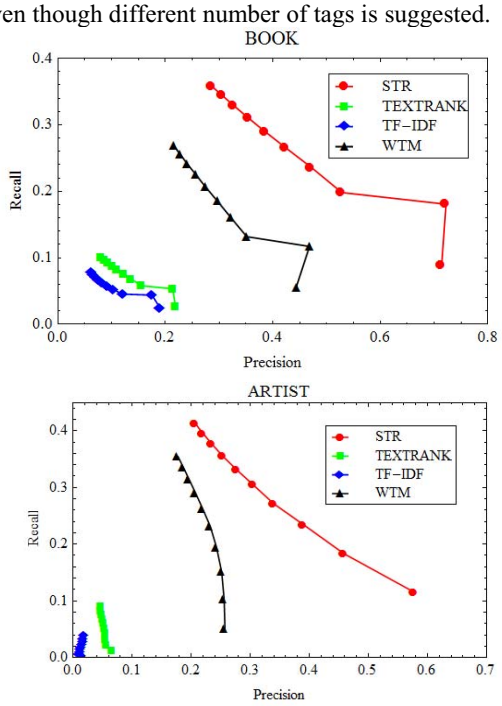

Figure 3. Precision-recall curves of STR, WTM, TextRank and TFIDF on BOOK and ARTIST datasets

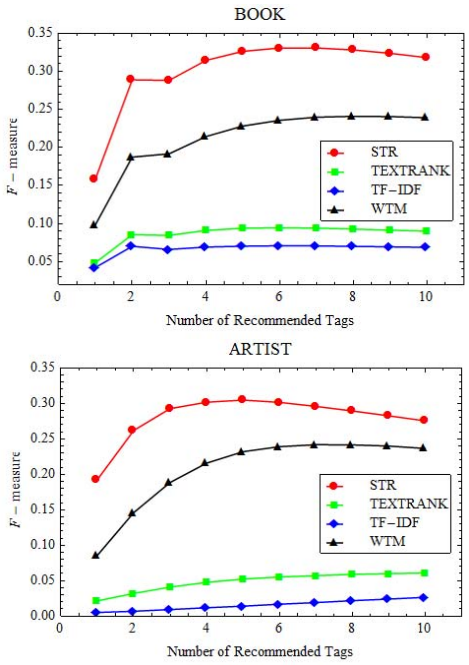

Figure 4. Number of recommended tags – F-measure curves of STR, WTM, TextRank and TFIDF on BOOK and ARTIST datasets

V. Related Work

Content-based social tagging recommendation methods are the usual approaches for addressing the cold start issue that new and unpopular resources with no or few annotations

will suffer. The basic idea of those methods is to generate content-relative tags from the description of resources. In this section, we will survey on several literatures on content-based tag recommendations in STS and divide them into four aspects, namely, classification-based, IR (Information Retrieval)-based [5], LDA (Latent Dirichlet Allocation)-based [11, 23] and SMT (Statistical Machine Translation)-based [13].

Classification-based methods regard each tag as a category label. Various classifiers [6, 8, 9, 12, 18, 20] such as KNN, SVM, Naïve Bayes and Neural Networks have been explored. As most of the resources are annotated by multiple tags, it is natural to adopt multi-label classification methods [10]. IR-based methods [5] made use of user and item profiles, presented and evaluated various adapted information retrieval models (Vector Space and Okapi BM25). In these methods, TFIDF weighing scheme is usually applied to the document profiles. With the widespread of latent topic models, researchers began to focus on modeling tags using Latent Dirichlet Allocation (LDA) [3]. Krestel et al. [11] and Si et al. [23] assumed that both tags and terms in description are generated from the same set of latent topic. Liu et al. [13] proposed a state-of-the-art social tagging recommendation, which regarded the issue as a task of selecting appropriate tags from a controlled tag vocabulary for the given resource and bridged the vocabulary gap between the descriptions and tags using word alignment models in SMT.

VI. Conclusion and Future Work

In this paper, we propose a new semantic tag ranking scheme for content-based social tagging recommendation. Experiments demonstrate that our approach is effective, robust and language-independent compared with the state of the art and baseline methods.

On the other hand, as we propose a new direction on semantic tag ranking for recommendation, there are several open problems that need to be explored in the future:

- We assume that the performance of STR will increase if the tag acquisition is more domain-specific. Because the polysemy in STS is a ubiquitous phenomenon, noise may be brought in if the learning knowledge is too general.
- The STR may add user profiles, which extends the binary relationship between resources and tags to a third-order tensor (users, description of resources, tags) to provide personal tag recommendation.

Acknowledgement

The research was support by the National Natural Science Foundation of China (No. 60873173). Thank the comments and advices of the anonymous reviewers.

References

[1] George A. Miller. 1995. WordNet: a lexical database for English. *Communications of ACM*. 38, 11, 39-41.

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

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

[4] 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.

[5] I. Cantador, A. Bellogín, D. Vallet. 2010. Content-based recommendation in social tagging systems. *In Proceedings of ACM RecSys*, 237-240.

[6] 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*, 35-48.

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

[8] S. Fujimura, KO Fujimura, and H. Okuda. 2008. Blogsonomy: Autotagging any text using bloggers' knowledge. *In Proceedings of WI*. 205-212.

[9] P. Heymann, D. Ramage, and H. Garcia-Molina. 2008. Social tag prediction. *In Proceedings of SIGIR*. 531-538.

[10] I. Katakis, G. Tsoumakas, and I. Vlahavas. 2008. Multilabel text classification for automated tag suggestion. *ECML PKDD Discovery Challenge 2008*.

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

[12] S.O.K. Lee and A.H.W. Chun. 2007. Automatic tag recommendation for the web 2.0 blogosphere using collaborative tagging and hybrid ann semantic structures. *In Proceedings of WSEAS*, 88-93.

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

[14] Z. Liu, W. Huang, Y. Zheng and M. Sun. 2010. Automatic keyphrase extraction via topic decomposition. *In Proceedings of EMNLP*, 366-376.

[15] Z. Liu, H. Wang, H. Wu and S. Li. 2009. Collocation extraction using monolingual word alignment method. *In Proceedings of EMNLP*. 487-495.

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

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

[18] G. Mishne. 2006. Autotag: a collaborative approach to automated tag assignment for weblog posts. *In Proceedings of World Wide Web*. 953-954.

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

[20] T. Ohkura, Y.Kiyota, and H. Nakagawa. 2006. Browsing system for weblog articles based on automated folksonomy. *In Proceedings of World Wide Web.*

[21] T. O'Reilly. 2005. What is Web 2.0: Design Patterns and Business Models for the next generation of software. http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html.

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

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

[24] Karen H. L. Tso-Sutter, Leandro Balby Marinho, and Lars Schmidt-Thieme. 2008. Tag-aware recommender systems by fusion of collaborative filtering algorithms. *In Proceedings of the 2008 ACM symposium on applied computing*. 1995-1999.