WSTRank: Ranking Tags to Facilitate Web Service Mining

Liang Chen[1], Zibin Zheng[2], Yipeng Feng[1], Jian Wu[1], and Michael R. Lyu[2]

[1] Zhejiang University, China
[2] The Chinese University of Hong Kong, China

Abstract. Web service tags, terms annotated by users to describe the functionality or other aspects of Web services, are being treated as collective user knowledge for Web service mining. However, the tags associated with a Web service generally are listed in a random order or chronological order without considering the relevance information, which limits the effectiveness of tagging data. In this paper, we propose a novel tag ranking approach to automatically rank tags according to their relevance to the target Web service. In particular, service-tag network information is utilized to compute the relevance scores of tags by employing HITS model. Furthermore, we apply tag ranking approach in Web service clustering. Comprehensive experiments based on 15,968 real Web services demonstrate the effectiveness of the proposed tag ranking approach.

1 Introduction

Web service[1] has become an important paradigm for developing Web applications. Especially the emergence of cloud infrastructure offers a powerful and economical platform to greatly facilitate the development and deployment of a large number of Web services [13]. Based on the most recent statistics[2], there are 28,593 Web services being provided by 7,728 distinct providers over the world and these numbers keep increasing in a fast rate.

WSDL (Web Service Description Language) document and extra description given by service providers are two major kinds of data to be utilized for Web services mining [8]. Despite the abundance of extra service description for most current Web services, limited semantic information can be obtained from the XML-based description document, i.e., WSDL document. The fast growing number and limited semantic information of Web services pose significant challenges to Web service mining, e.g., Web service clustering, Web service searching, etc.

[1] In this paper, we focus on non-semantic Web services. Non-semantic Web services are described by WSDL documents while semantic Web services use Web ontology languages (OWL-S) or Web Service Modeling Ontology (WSMO) as a description language. Non-semantic Web services are widely supported by both the industry and development tools.

[2] `http://webservices.seekda.com`

C. Liu et al. (Eds.): ICSOC 2012, LNCS 7636, pp. 574–581, 2012.

In recent years, tagging, the act of adding keywords (tags) to objects, has become a popular mean to annotate various Web resources, e.g., Web page bookmarks, online documents, and multimedia objects. Tags provide meaningful descriptions of objects, and allow users to organize and index their contents. Tagging data was proved to be very useful in many domains such as multimedia, information retrieval, data mining, and so on [1][12]. In Web service domain, some Web service search engines, such as SeekDa!, also allow users to annotate tags to Web services. Recently, Web service tags are being treated as collective user knowledge for Web service mining, and attract a lot of attention. Some research work have been conducted to employ tagging data for Web service clustering[6], Web service discovery[3][9], Web service composition [5], etc.

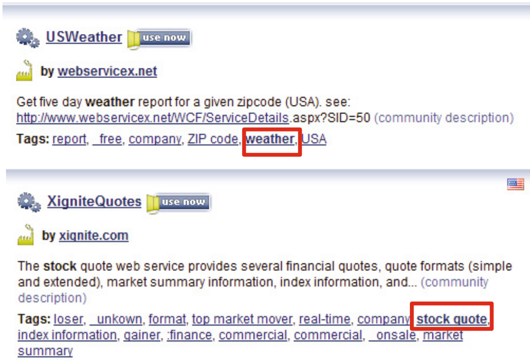

Fig. 1. Two exemplary Web services from SeekDa!

However, existing studies reveal that many tags provided by SNS (Social Network System) users are imprecise and there are only around 50% tags actually related to the target object [10]. Furthermore, the relevance levels of tags can't be distinguished from current tag list, where tags are just listed in a random order or chronological order without considering the relevance information. Figure 1 shows two exemplary Web services[3] from SeekDa! and their tags annotated by users. Take the *USWeather* Web service as an example, its most relevant tag, i.e., "weather", can not be discovered from the order of tag list directly. Similarly, the most relevant tag to *XigniteQuotes* Web service is "stock quote", while its position in the tag list is the 7^{th}. Furthermore, there are some imprecise tags annotated to Web services, such as "unknown","format", etc.

Relevance-independent tag list and imprecise tags limit the effectiveness of tags in Web service mining, or even produce negative effects. In this paper, we propose a novel tag ranking approach named *WSTRank*, to automatically rank tags according to their relevance to the target Web service. In *WSTRank*,

[3] http://webservices.seekda.com/providers/webservicex.net/USWeather
http://webservices.seekda.com/providers/xignite.com/XigniteQuotes

we employ HITS [11] model to obtain the relevance score of tag based on a service-tag network.

To demonstrate the effectiveness of tag ranking approach for Web service mining, we apply *WSTRank* into one classical application, i.e., Web service clustering, which is usually used to cluster the Web services with the same or similar functionality to handle the low recall of Web services search.

In particular, the main contribution of this paper can be summarized as follows:

1. This paper identifies the critical problem of tag ranking for Web service mining and proposes a hybrid approach named *WSTRank* to rank tags of Web services. To the best of our knowledge, *WSTRank* is the first tag ranking approach for Web services.
2. Extensive real-world experiments are conducted to study the tag ranking performance of *WSTRank*. The experimental results demonstrate the effectiveness of *WSTRank*. Further, we evaluate the impact of tag ranking to Web service clustering.
3. We publicly release our Web service tag dataset to promote future research, which includes 15,968 real-world Web services and their tags. The released dataset makes our experiment reproducible.

2 Web Service Tag Ranking

In this section, we introduce the computation of HITS based tag authority, which is treated as the relevance score of tag. Hyperlink-Induced Topic Search (HITS) (also known as hubs and authorities) is a link analysis algorithm that rates Web pages, developed by Jon Kleinberg. It is a precursor to PageRank. The idea behind Hubs and Authorities stemmed from a particular insight into the creation of Web pages when the Internet was originally forming. In this section, we propose to obtain the authority of tag in the service-tag network, which could reflect the importance of tag. In the following, we first introduce how to build the service-tag network, and then present a HITS-based algorithm for tag authority computation.

2.1 Service-Tag Network Building

Service-tag network can be modeled as a weighted directed graph G, where node s_i means a service and node t_i means a tag. For each node in G, it has two values, i.e., hub and authority. There are three kinds of directed edges in G:

1. Edge from service node to tag node. Given a service s_1 annotated with three tags t_1, t_2, and t_3, then there is a directed edge from s_1 to t_1, t_2, and t_3, respectively. In particular, the weight of this kind of edge is 1.
2. Edge from service node to service node. Given two services s_1 and s_2, if there is one or more than one common tags annotated to these two services, we create one directed edge from s_1 to s_2 and one directed edge from s_2 to s_1.

These two edges have the same weight, which is depended on the common tags, i.e., $weight = \frac{t_{s_1} \bigcap t_{s_2}}{t_{s_1} \bigcup t_{s_2}}$, where t_{s_1} and t_{s_2} mean the set of tags annotated to s_1 and s_2, respectively.

3. Edge from tag node to tag node. Given two tags t_1 and t_2, and these two tags are annotated to one or more than one services. Similarly, we create one directed edge from t_1 to t_2 and one directed edge from t_2 to t_1. The weight of edge is also depended on the common services, i.e., $weight = \frac{s_{t_1} \bigcap s_{t_2}}{s_{t_1} \bigcup s_{t_2}}$, where s_{t_1} and s_{t_2} mean the set of services contain t_1 and t_2, respectively.

In this way, we obtain the service-tag network by building a weighted directed graph.

2.2 Tag Authority Computation

HITS based algorithm is a kind of iterative algorithm. We consider two types of updates as follows:

- **Authority Update**. For each node p in G, we update the authority of node p to be:

$$Auth(p) = \sum_{i=1}^{n} Hub(p_i) \times w(p_i, p), \tag{1}$$

  where $p_i(i = 1, \dots, n)$ means the node that points to p, and $w(p_i, p)$ is the weight of edge from p_i to p. That is, the authority of node p is the sum of all the weighted hub values of nodes that point to p.
- **Hub Update**. For each node p in G, we update the hub value of p to be:

$$Hub(p) = \sum_{i=1}^{n} Auth(p_i) \times w(p, p_i), \tag{2}$$

  where $p_i(i = 1, \dots, n)$ means the node that p points to, and $w(p, p_i)$ means the weight of edge from p to p_i.

Algorithm 1 shows the detailed HITS based computation process. As the initialization, we set the authority value and hub value of each node in G as 1 (line 1-3). K in line 4 means the number of iterations. Empirically, we set $K = 50$ in the experiments. The parameter *norm* is used for normalization, and is initialized as 0 (line 5). According to the Authority Update rule, we compute the authorities of all nodes in G, and then normalize them by using parameter *norm* (line 6-16). Similarly, hub values of nodes can be computed by employing Hub Authority rule (line 18-29). After K iterations, we return the authorities of all tag nodes (line 30-32).

3 Experiment

In this section, we first give a brief description of dataset and experiment setup, and then compare the performance of different tag ranking approaches in terms of NDCG.

Algorithm 1. Tag Authority Computation Algorithm

Input: G:service-tag network; K: number of iterations
Output: Auth(t): authority of tag node

```
for all node p in G do
    Auth(p)=1,Hub(p)=1
end for
for iteration from 1 to K do do
    norm=0
    for all node p in G do
        Auth(p)=0
        for all node p_i which points to p do
            Auth(p)+=Hub(p_i) ×weight(p_i, p)
        end for
        norm+=square(Auth(p))
    end for
    norm=sqrt(norm)
    for all node p in G do
        Auth(p)=Auth(p)/norm
    end for
    norm=0
    for all node p in G do
        Hub(p)=0
        for all node p_i that p points to do
            Hub(p)+=Auth(p_i) ×weight(p, p_i)
        end for
        norm+=square(Hub(p))
    end for
    norm=sqrt(norm)
    for all node p in G do
        Hub(p)=Hub(p)/norm
    end for
end for
for all tag node t in G do
    return Auth(t)
end for
```

3.1 Dataset Description and Experiment Setup

To evaluate the performance of *WSTRank*, we employ the dataset consists of 15,968 real Web services crawled form the Web service search engine Seekda!. For each Web service, we can obtain the information of service name, WSDL document, tags, availability, and the name of service provider.

For each service, each of its tags is labeled as one of the five levels: Most Relevant (score 5), Relevant (score 4), Partially Relevant (score 3), Weakly Relevant (score 2), and Irrelevant (score 1). As the manual creation of ground truth costs a lot of work, we select 98 Web services from the dataset and distinguish the following categories: "Email", "Stock", "Tourism", "Weather", "Calculation", and "Linguistics". Specifically, There are 11 Web services in "Email"category, 18 Web services in "Stock"category, 20 Web services in "Tourism"category, 14 Web services in "Weather"category, 18 Web services in "Calculation"category, and 17 Web services in "Linguistics"category. Due to the space limitation, we don't shows the detailed information of these Web services.

It should be noted that all experiments are implemented with JDK 1.6.0-21, Eclipse 3.6.0. They are conducted on a Dell Inspire R13 machine with an *2.27 GHZ Intel Core I5 CPU* and *6GB RAM*, running *Windows7 OS*.

3.2 Performance Evaluation of Tag Ranking

To study the performance of tag ranking, we first compute the NDCG vaule of Baseline (i.e., original tag lists), and then compare the performance of the following approache:

- **WSTRank**. In this approach, linking relationship in the service-tag network is employed to rank tags. In this experiment, we choose HITS model to represent the linking relationship.

To evaluate the performance of Web service tag ranking, we employ the Normalized Discounted Cumulative Gain (NDCG) [2] metric, which is widely accepted as the metric for ranking evaluation in information retrieval. Table 1 and Table 2 show the ranking performance of above 4 approaches, respectively employing NDCG@3 and NDCG@5 as the evaluation metric. NDCG@k indicates that only the ranking accuracy of the top-k tags is investigated. Given one category of Web services, we compute the NDCG@k value of each Web service, and set the average value as the NDCG@k value of this category. For each column in the Tables, we have highlighted the best performer among all approaches. The values shown in the bottom row are the performance improvements achieved by the best methods over the Baseline.

Table 1. NDCG@3 performance of tag ranking

Method	Tourism	Weather	Calcu	Lingu	Stock	Email	Average
Baseline	0.756	0.862	0.602	0.621	0.806	0.869	0.753
WSTRank	**0.797**	**0.917**	**0.962**	**0.771**	**0.935**	**0.911**	**0.882**
	5.42%	6.38%	59.8%	24.15%	16.00%	4.83%	17.1%

Table 2. NDCG@5 performance of tag ranking

Method	Tourism	Weather	Calcu	Lingu	Stock	Email	Average
Baseline	0.714	**0.892**	0.709	0.787	**0.877**	**0.901**	0.813
WSTRank	**0.781**	0.855	**0.917**	**0.788**	0.862	0.874	**0.846**
	9.38%	-4.32%	29.34%	0.13%	-1.74%	-3.09%	4.06%

From above two Tables, it can be observed that our proposed *WSTRank* approach largely improves the accuracy of tag ranking. Compared with the Baseline, the improvement brought by *WSTRank* achieves 59.8% at the highest point, and achieves -4.32% in the worst case. In addition, we can also find that the improvement caused by *WSTRank* always decreases when the value of k increases from 3 to 5.

4 Related Work

With the popularity of SNS, tagging data, which is annotated by users and provides meaningful descriptions, is widely employed in many research domains such as mutlimedia, information retrieval, data mining, etc [1]. Recently, tagging data oriented technologies are also employed in service oriented computing. Eric *et al.* propose a folksonomy-based model for Web service discovery and automatic composition, in which tags are utilized as semantic information [5]. In our premise work, we utilize both WSDL documents and tags to cluster Web services, based on the notion that combining users' knowledge and service providers' knowledge [6]. Tagging data is also employed in Web service discovery [7]. To handle the problem of limited tags, Zeina *et al.* propose to employ machine learning technology and WordNet synsets to automatically annotate tags to Web services [4].

5 Conclusion

In this paper, we propose to rank Web service tags to facilitate Web service mining. In our proposed *WSTRank* approach, we utilize the linking relationships in service-tag network to obtain the relevance scores of tags. In particular, HITS model is employed to compute the authority of tag in service-tag network. The experimental results based on real Web services demonstrate the effectiveness of *WSTRank* approach.

In our future work, we plan to expand the scale of tag dataset by inviting volunteers and employing automated tagging approaches. Moreover, *WSTRank* will be applied in applications of Web service mining, e.g., Web service clustering, Web service search and Web service recommendation, to verify the effectiveness of tag ranking in Web service mining.

Acknowledgements. This research was partially supported by the National Technology Support Program under grant of 2011BAH15B05, the National Natural Science Foundation of China under grant of 61173176, Science and Technology Program of Zhejiang Province under grant of 2008C03007, National High-Tech Research and Development Plan of China under Grant No. 2009AA110302, National Key Science and Technology Research Program of China (2009ZX01043-003-003).

References

1. Ames, M., Naaman, M.: Why we tag: Motivations for annotation in mobile and online media. In: Proc. of the SIGCHI Conference on Human Factors in Computing Systems (CHI), pp. 971–980 (2007)
2. Arvelin, K.J., Kekalainen, J.: Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems 20(4), 422–446 (2002)
3. Averbakh, A., Krause, D., Skoutas, D.: Exploiting User Feedback to Improve Semantic Web Service Discovery. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 33–48. Springer, Heidelberg (2009)
4. Azmeh, Z., Falleri, J.-R., Huchard, M., Tibermacine, C.: Automatic Web Service Tagging Using Machine Learning and WordNet Synsets. In: Filipe, J., Cordeiro, J. (eds.) WEBIST 2010. LNBIP, vol. 75, pp. 46–59. Springer, Heidelberg (2011)
5. Bouillet, E., Feblowitz, M., Feng, H., Liu, Z., Ranganathan, A., Riabov, A.: A folksonomy-based model of web services for discovery and automatic composition. In: IEEE International Conference on Services Computing, pp. 389–396 (2008)
6. Chen, L., Hu, L., Zheng, Z., Wu, J., Yin, J., Li, Y., Deng, S.: WTCluster: Utilizing Tags for Web Services Clustering. In: Kappel, G., Maamar, Z., Motahari-Nezhad, H.R. (eds.) ICSOC 2011. LNCS, vol. 7084, pp. 204–218. Springer, Heidelberg (2011)
7. Ding, Z., Lei, D., Yan, J., Bin, Z., Lun, A.: A web service discovery method based on tag. In: International Conference on Complex, Intelligent and Software Intensive Systems, pp. 404–408 (2010)
8. George, Z., Athman, B.: Web service mining. Springer (2010)
9. Hou, J., Zhang, J., Nayak, R., Bose, A.: Semantics-Based Web Service Discovery Using Information Retrieval Techniques. In: Geva, S., Kamps, J., Schenkel, R., Trotman, A. (eds.) INEX 2010. LNCS, vol. 6932, pp. 336–346. Springer, Heidelberg (2011)
10. Kennedy, L.S., Chang, S.F., Kozintsev, I.V.: To search or to label?: predicting the performance of search-based automatic image classifiers. In: Proc. of the 8th ACM International Workshop on Multimedia Information Retrieval, pp. 249–258 (2006)
11. Li, L., Shang, Y., Zhang, W.: Improvement of hits-based algorithms on web documents. In: Proc. of the 11th International World Wide Web Conference, pp. 527–535 (2002)
12. Sigurbjörnsson, B., van Zwol, R.: Flickr tag recommendation based on collective knowledge. In: Proc. of the 17th International Conference on World Wide Web (WWW), pp. 327–336 (2008)
13. Zheng, Z., Ma, H., Lyu, M.R., King, I.: Wsrec: A collaborative filtering based web service recommender system. In: Proc. of the 7th International Conference on Web Services (ICWS), pp. 437–444 (2009)