Web Services Discovery based on Semantic Tag

Sana Sellami, Hanane Becha *

Aix Marseille University, CNRS, ENSAM, Universit de Toulon, LSIS
UMR7296,13397, Marseille, France
{sana.sellami@lsis.org,hbecha@gmail.com}

Abstract. Recently tagging has been employed to improve the performance of service discovery. Two main challenges have to be addressed when tags are used in Web service discovery: tag relevancy and tag sense disambiguation. In this paper, we present our Web service tagging approach that addresses these problems and describe the results of experiments on a collection of real Web services.

Keywords: Web Service discovery, tag, semantic

1 Introduction

Nowadays, many software applications are developed by invoking different loosely-coupled services, residing in the network and accessible via standardized protocols. The quality of these applications is highly dependent on the quality of the selected Web services from which they are built; hence, the crucial importance of service discovery mechanisms. Generally, service discovery is performed based on matching the input, output, preconditions and effects of Web services descriptions available in the Universal Description Discovery and Integration (UDDI). Recently, tagging was employed to improve the performance of service discovery. However, the rich semantic information included in the user-contributed tags has many issues that have to be addressed; thus, service discovery techniques can take full advantage of the tagging mechanisms. We address in this paper two main challenges when tags are used in Web service discovery : tag relevancy and tag sense disambiguation. First, to address the tag relevancy issue, we recommend the addition of three parameters to each tag: score, popularity, and occurrence. A score is assigned to each tag to denote the relevance of that tag from the user's perspective. Popularity denotes the relevance of a given tag according to the user's expertise. More weight is given for the most experienced users' tags. Occurrence is the number of times that a given tag was added to the same service. Second, to address the tag sense disambiguation, we use the WordNet dictionary to take into account the synonyms of the tags in the service search.

* Please note that the LNCS Editorial assumes that all authors have used the western naming convention, with given names preceding surnames. This determines the structure of the names in the running heads and the author index.

The remainder of the paper is organized as follows. Section 2 presents our approach. Section 3 is devoted to the experimental results. Section 4 surveys some related works and section 5 concludes the paper.

2 Web Services Tagging Approach

The main goal of our approach is to prompt tagging that enables unlimited number of ways to classify Web services. Web services that have several different tags enhance the reusability of services in different contexts and business domains. Ideally, users are encouraged to tag a maximum number of Web services using as many relevant tags as possible. Our approach consists of three main processes: Web services tagging, semantic search of tagged Web services and ranking processes.

2.1 Web Services Tagging

Web service Tagging is the process of describing a resource by assigning some relevant key-words (tags) to it. Users can add a set of tags to each Web service and have the ability to request and select a service by specifying a set of tags.

Users can improve the tag-based service discovery mechanism by assigning new tags to each Web service that they have invoked. The user is empowered to perform service discovery by defining a search request using tags and their associated scores only. A Tagging-based request allows users to describe in a simple and specific manner precisely which services correspond best to their needs by specifying tags. Tagging-based request and tagged services are the two key parameters (i.e., services and request tags) which empower users to select the most relevant services for their request based on tag relevance (i.e., between the tags assigned to services and the tags specified in the user's requests and services). However, one of the tag use limits is the tag relevance. To avoid imprecise tags, users can assign tags only to the services that they have actually invoked. Even though, one can argue that there are no wrong tags, still some tags could be more meaningful or relevant than others with respect to a given service. In order to address the tag relevance problem, users are requested to add a score value to each tag that they add (see Definition 1).

Definition 1 (Tag definition): A tag denoted as Tag(S) is a couple of name and a score: Tag (S) = ($name_{Tag}$, $score_{Tag}$)

where $name_{Tag}$ and $score_{Tag}$ are respectively the name and the score of the tag $score_{Tag} \in [0, 100]$. We define the scores values as described in Table 1.

A service could have an unlimited number of tags. Whenever, the same tag has been added more than once, the average of scores is calculated.

2.2 Semantic Search of Web Service

The semantic search process locates the most relevant services according to the user's request by performing a similarity search function. This search function

Tag label	Tag significance	Score range
Strong	Most relevant Tag	[90-100]
Secondary	Relevant Tag	[60-90]
Weak	The tag is partially relevant	[30-60]
Fail	The tag may be irrelevant	[0-30]

Table 1. Score values of the tags

takes a set of tags and their score relevance as inputs to calculate the similarity between tags and their scores (see below Definition 2). Each similarity has a weight assigned to it. The sum of the assigned weight of both similarities is equal to one. During the tag search, sense disambiguation techniques based on the WordNet [1] lexical thesaurus are also applied. In WordNet each term (tag) is a synset and it has a set of synonyms. Once the search is performed, a list of the most relevant services is returned to the user.

Definition 2 (Semantic Tag Similarity): The tag similarity denoted as Sim_{Tag}(R, S), where R is a user request and S a service, is defined as follow:
$Sim_{Tag}(R,S) = Sim_{score}(TagR, TagS) * W_{score} + Sim_{sem}(TagR, TagS) * W_{sem} \in [0,1]$ Where

- Sim_{score} (TagR, TagS) is a tag similarity between the user's score request and services:
  $Sim_{score}(TagR, TagS) = \frac{|min(Score(|TagR \cap TagSi|))|}{\sum Score(TagR)} \in [0,1]$
- Sim_{sem} (TagR, TagS) is semantic similarity between tags. This similarity considers synonym relationships based on WordNet Thesaurus.

$$Sim_{sem}(TagR, TagS) = \frac{|Tag(R) \cap Tag(S)|}{|Tag(R)|} \in [0,1]$$

- W_{score} and W_{sem} are weight set of the tag score and the semantic similarity, respectively and W_{score} +W_{sem}=1 .

2.3 Tagged Web Services Ranking

Semantic similar services are sorted based on users' votes to find the most relevant services. The ranking is based on both the tag similarity and the user's vote (Ve) which considers the user's expertise via a simple quiz to give more weight for tags assigned by most experienced users. The user's vote (Ve) is defined as follows:

$$Ve = \begin{cases} 2 & \text{if user responds correctly to the quiz questions} \\ 1 & \text{if the user makes one mistake} \\ 0 & \text{if mistake>1} \end{cases}$$

[1] https://wordnet.princeton.edu

Based on user's vote and the occurrence of tag, we can define the most popular tags as follow:

Definition 3 (Tag popularity): Popularity evaluates the relevance of the tag according to the user's expertise. It is the ratio of the number of times that a service was tagged relative (Ve(Si)) to the sum of all the number of the services Votes (Ve(S)).

$$TagPopularity(Si) = \frac{|Ve(Si)|}{\sum Ve(S)}$$

3 Preliminary Evaluation

3.1 Experimental Setup

We performed an evaluation on the precision of the semantic search and Web services ranking in our proposed framework for tagging Web Services called WSTP [2](Web Services Tagging Platform). We considered a collection of 151 Web services from different categories including Stock, Tourism, Weather, Telecommunication, Economy and Finance. This dataset [3] [1] is the one used in the Titan [4] search engine. First, we tag all these Web services with the same tags provided by the Titan search engine. We evaluate the precision of the service retrieval (e.g., the number relevant services among all the retrieved ones) and recall (number of relevant services that are retrieved) measures.

3.2 Experimental Results

Tagged Web Services Search Evaluation: We performed an initial evaluation on the precision of WSTP in retrieving relevant Web services. We focused on two categories: Tourism which includes several tagged services and Stock which contains a few tagged services, viz.23% of services (see Table 2). We performed two different query searches: a simple tag (stock:100) and a combination of most frequent tags (e.g., stock:100 company:100).

Categories	Number of services	Number of tagged Web services	Ratio
Tourism	39	31	79%
Stock	39	9	23%

Table 2. Tagged Web services characteristics

As illustrated in Fig. 1, precision values are the same regardless of the query tag searches. WSTP returned only tagged Web services which contain the queried

[2] http://www.lsis.org/sellamis/Projects.html
[3] http://www.zjujason.com/data.html
[4] http://ccnt.zju.edu.cn:8080/

tag. We notice the increasing of recall values when we use a combination of tags. The more tags associated to a service, the greater the probability to retrieve relevant services, this validates our assumption.

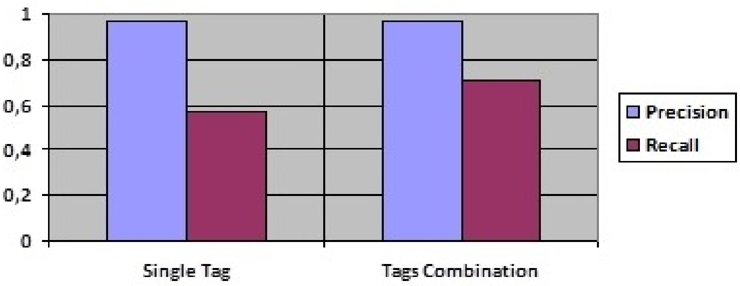

Fig. 1. Precision and Recall of WSTP according to the tag queries

We performed the precision and recall evaluation of WSTP according to two categories (Fig. 2). We can observe that the recall value is higher for the search on the Tourism category than for the Stock category. The reason for this was that the Tourism category has 79% of Web services tagged whereas only 23% were tagged in the Stock category.

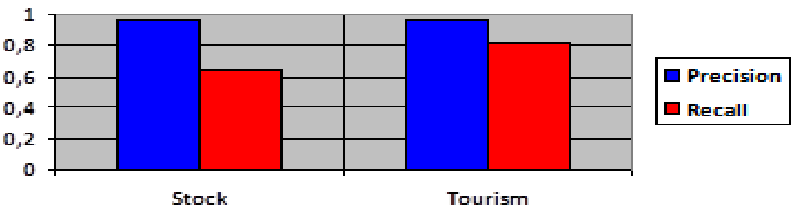

Fig. 2. Precision and Recall of WSTP according to the domains

Search Engine Ranking: We compared the WSTP results with those of the Titan Search engine to evaluate ranking procedure. Our hypothesis is that the first returned services are the most relevant ones according to the user request. Then, we executed the same request in both the WSTP and Titan search engines and we compared the obtained results by using the Normalized Discounted Cumulative Gain (NDCG) as defined below: Given the ideal search ranking (used as baseline) and a predicted search ranking, the NDCG value of Top-k search services can be calculated by:

NDCG@k= $\frac{DCG@k}{IDCG@k'}$

Where DCG@k and IDCG@k are the discounted cumulative gain (DCG) values of the top-k services of the predicted ranking and ideal ranking, respectively. The discounted CG (DCG) accumulated is defined as:

DCG@k=rel1+ $\sum_{i=2}^{p} \frac{reli}{log_2(i)}$
Where rel_i is the graded relevance of the result at position i.

NDCG@K	Tool	Stock	Tourism
K=3	WSTP	0.70	0.87
	Titan	0.74	0.87
K=5	WSTP	0.68	0.85
	Titan	0.72	0.85

Table 3. NDCG@K performance of tagged Web services ranking

Experimental results (see Table 3) showed that our system produced the same search results for the Tourism category that were obtained by Titan. However, for the Stock category WSTP does not perform as well. This occurs because only 23% of the Web services are tagged in the Stock category. Moreover, it is interesting to underline that, if we change the search tag tourism with touristy (which is the synonym of tourism); WSTP returns all the results corresponding to the tourism, while Titan does not return any results for this query. Compared to Titan, WSTP offers a semantic search service based on the WordNet thesaurus and thus, it provides a better visualization and browsing of services based on links. Moreover, WSTP displays better results than Titan when we combine different tags of services. However, WSTP could be improved to better process queries by considering n-gram, prefix and suffix techniques, etc.

4 Related work

Several Web services tagging approaches have been proposed in the literature. We discuss those that rely on tag identification and tag relevance. Mining approaches clustering algorithms have been applied to assign tags to Web services [3], [4],[5], [6]. For example, the authors in [3] applied clustering techniques to extract tags from WSDL documents and consider both structural and lexical information of text segments. Tags are then generated automatically. However, their approach highly depends on the WSDL description of Web services and does not consider other description languages. The approach in [5] uses a clustering technique based on Carrot search clustering and K-means to group similar services to generate tags. The authors also proposed the use of naive bayes algorithm to rank Web services and recommend WSDL services based on tags. The second category of approaches is those that deal with tag relevance. The approach proposed in [2] handles the problem of imprecise, irrelevant and malicious tags and it considers the tag relevance for Web service mining. This approach is based on a mechanism called WS-TRM to measure the relevance of user tag. It considers both semantic tag relevance and the relationship in Service Tag Network (STNet). The tag relevance is computed by evaluating the semantic relevance between each tag and the WSDL document of the corresponding service. Semantic relevance considers both syntactic and structural relations between tags and WSDL elements. In [7] the authors propose a system for semantic collaborative tagging APIs to be deployed in a ProgrammableWeb [5]. The approach addresses the polysemy and homonyms problems and takes into account a social characterization of Web APIs that considers the past experiences of Web designers. However, in ProgrammableWeb, tags are only assigned to the source code of APIs and the APIs themselves are not tagged. Moreover, most of the published APIs do not contain code source support and tags are assigned according to the second category of the service. Service search is then based on category traversal search which is expensive when there are thousands of services. To handle the problem of limited tags, Azmeh et al. [8] propose to employ machine learning technology and WordNet synsets to automatically assign tags to Web services. In this paper, the authors propose an approach that automatically generates a set of relevant tags from a WSDL service description. Similar to our approach, authors in [9] propose a collaborative tagging system for Web service discovery. Tags are labels that a user can associate to a specific Web service and for each tag, a tag weight is assigned. A tag weight is the count of number of occurrences of a specific tag associated to a Web service. Unlike existing approaches, our approach is technology agnostic and it is not tied to the Web Service description language WSDL. Moreover, our approach has been implemented, evaluated on a real collection of Web services, and demonstrated to perform better results.

5 Conclusion and future works

In this paper, we described a Web services discovery approach based on a semantic tag that addresses tag relevancy and tag sense disambiguation. The approach has been implemented and tested against a collection of real Web services. In the future, we plan to enhance the semantic services search by enabling querying of LOD (Linked Open Data) like DBpedia[6] and interacting with the Programmable Web to retrieve information on Web APIs from the repository. With this enhancement we will be able to provide a platform for linked services based not only on tag similarities but also on mashups and category links.

References

1. Chen, L., Wang, Y., Yu, Q., Zheng, Z., Wu, J.: WT-LDA: User Tagging Augmented LDA for Web Service Clustering. 11st International Conference on Service Oriented Computing (ICSOC), Berlin, Germany, Dec 2 - 5, (2013) 162-176

[5] http://www.programmableweb.com/
[6] http://wiki.dbpedia.org/

2. Chen, L., Wu, J., Zheng, Z., R. Lyu, M., Wu, Z.: Modeling and exploiting tag relevance for Web service mining. Knowl. Inf. Syst.39, (2014) 153-173
3. Fang, L., Wang, L., Li, M., Zhao, J., Zou, Y., Shao, L.: Towards Automatic Tagging for Web Services. 2012 IEEE 19th International Conference on Web Services, (2012) 528-535
4. M. Kapitsaki,G.: Annotating Web Service Sections with Combined Classification, in Proceedings of the 21th IEEE International Conference on Web Services (ICWS), (2014) 622-629
5. Lin, M., W. Cheung, D.: Automatic Tagging Web Services Using Machine Learning Techniques. International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) (2014) 258-265
6. Pan, W., Chen, S., Feng, Z.: Automatic Clustering of Social Tag using Community Detection. Applied Mathematics & Information Sciences; Vol. 7 Issue 2, (2013) 675
7. Bianchini, D., De Antonellis, V., Melchiori, M. : Semantic Collaborative Tagging for Web APIs Sharing and Reuse. 12th International Conference on Web Engineering ICWE, Lecture Notes in Computer Science, Vol.7387, Springer (2012) 76-90
8. Azmeh, Z., Falleri, J-R., Huchard, M., Tibermacine, C.: Automatic Web Service Tagging Using Machine Learning and WordNet Synsets. Web Information Systems and Technologies - 6th International Conference WEBIST (2010) 46-59
9. Chukmol, U., Benharkat A., and Amghar Y.: An Approach for Web Service Discovery based on Collaborative Structured Tagging. Proceedings of the 12th International Conference on Enterprise Information Systems ICEIS (2010) 47-56