Abstract—Rapid growth of published Web services makes their discovery more and more difficult. Many approaches are proposed to solve this discovery issue. Most of them are interested only in Web service’s description itself and neglect the user-centric aspect of the discovery process. User’s involvement may be seen as contributing information, after using or testing a service, which marks the personal opinion on service’s functionality, quality or invocation cost. This kind of contribution is perceived as a participative aspect inside the Web 2.0 environment and can enhance the Web service discovery process. We illustrate in this research work, a collaborative tagging-based environment for Web service discovery, allowing users to tag or annotate a Web service using keyword or free-text. Our system proposes consequently two types of query to search tagged Web services: keyword-based and free-text. We put in place an advanced mode in the discovery by keyword which offers different ways to combine keywords together inside a query to make it more flexible and accurate. In both types of look-up (by keyword and by free-text), synonym relation between terms is processed in order to improve the similarity computing between a query and the tagged web services. The Web services found after a discovery process are ranked according to their tags’ weight, in the case of discovery by keywords.

Index Terms—Web Service Discovery, Collaborative Tagging System, Vector Space Model, WordNet

I. INTRODUCTION

Rowning into a great number of publicly accessible Web services, a user (a neophyte Web user who needs to start with Service Oriented Computing, a designer or developer of Service Oriented Applications) can be confronted with a difficult exercise when trying to find a Web service relevant to a requirement. Generally, these users have difficulties in exploring Web services in a few steps and, to the best that we know, nothing is available yet to facilitate the construction of their requests in a free-style, flexible, intuitive but efficient way. Most of the time, users discover Web services through a search interface using text based query with some advanced features or they have to construct their queries in a complicated formalism to express their searches’ requirement. We argue that both styles of query do not facilitate the Web service discovery. The reasons behind this affirmation are: firstly, simple keyword based discovery mechanism frequently offers low matching result in the current environments because they deal with poorly described Web services (insufficient textual description of Web services within WSDL files). Secondly, complicated formalisms used to construct discovery’s queries are often description logic based, which require efforts from not only the users while constructing their queries but also from the service’s providers to enrich their published services with description logic or ontological information. The search algorithms, in this case, have recourse to ontology usage and demand standardized ontology definition.

Besides, we can consider the Web as the simplest environment for neophyte users to explore Web services as if they were executing an information lookup process. In effect, after the official shutdown of UDDI project in 2006, Internet has gained a real popularity as a global repository of Web services. We are able to find different Web sites providing access to Web services such as XMethods [23], WebServiceX.NET [25], Strikelon [24], etc. Different existing works emphasize and confirm this statement. In [26], Internet is perceived as a natural source of Web services because Web service’s descriptions are hosted in various Web servers accessible by Web crawlers. A collection of Web services crawled from the Web is provided in [22]. In a simpler way, [28] uses different general purpose search engines in the service discovery process. These works allow us to argue that there are several interesting access points to active Web services that can be found on the Internet and using this world wide environment will bring Web services a large step closer to mass public users.

Despite the fact that there are no sufficiently rich hyperlink structures between WSDL documents (Web service’s description), a Web based crawler can still be operational to collect this kind of files from the Internet [27]. Building a collection of Web services that are extracted from the Web can assure not only the public access to the services but also the regular update of services. We, henceforth, will focus our work...
on the public Web services which can be crawled from the Web.

Currently, the Web 2.0 concepts are widely adopted by several Web sites to enrich the interaction and communication between their users. We can remark that the social network implementation has emerged and allows Web users to, not only, search for Web resources but also share their ideas, opinions, bookmarks, feedbacks and finally knowledge (semantics) on the resources.

Consequently, discovering a Web resource is not limited to simple keyword based search anymore. The retrieval process can be a lot more enhanced by using tags in the Web 2.0 enabled environment. The success of such approach can be interpreted by the increasing number of users, resources and the usage’s rate of different applications on the Web such as Del.icio.us [29], Flickr [30], Connotea [31], CiteULike [32], Technorati [33], etc. Within these collaborative tagging systems, users can tag or annotate a resource (e.g., picture, URL, bibliographical reference, etc.) with keywords in order to classify it, share it with other users and facilitate the future search for that resource.

This kind of environment incites users to participate in the process of enriching any Web resource with textual descriptions (keywords or free text). A Web resource can then be discovered via the tags cloud that is associated to itself.

If we consider a Web service through its physical characteristics, it is represented in the World Wide Web environment by a URL pointing at its description (in WSDL). The latter does not “hold sufficient textual description” [1]. Associating textual description that reveals effective capability of service to the current number of published Web services or extending them with semantic annotation using formalism such as SAWSDL [19] (a semantic annotation language for WSDL), OWL-S [34] or WSDL-S [35] demand not only tedious, error-prone manual work and elaborated ontology engineering task buts also seem strongly to be an impossible task for the service’s providers. However, tagging these services will just be a participative or collaborative task of service’s users or between the service’s developers and consumers. It does, to the utmost, require very slight and negligible effort from the service’s provider and does not affect the integrity of service’s descriptions. We can also assure through this mechanism that the Web users’ habits do not change and the discovery of any Web service will happen in the same way as in other processes of search for Web resources (e.g., picture, URL, etc.).

Being convinced by various advantages of collaborative tagging systems in the Web 2.0 environment and the difficulties met by users when looking up a Web service, we are going to propose in this research work a Web service discovery environment based on collaborative tagging systems. We believe that such a system will help those who require searching for a Web service not only in easing the query construction but also keeps their search behaviors familiar to the one they are used to (searching for a tagged resource in the Web). Moreover, we will discuss later in this proposal how a Web service can be tagged and found later using tags.

The rest of this paper is organized as follows; we first describe, in section 2, briefly what collaborative tagging systems and their dynamics are. Then, we focus our study on some existing significant user-centric Web service discovery mechanisms. Later on, in section 3, we present in detail our Web service discovery environment. Next, we discuss a usage context of such system in section 4 and we present how we are going to implement and evaluate our prototype in section 5. Finally, we conclude our work and present our perspectives in the section 6.

II. RELATED WORK

Organizing digital contents using tags becomes a tendency and has gained a large popularity among contemporary Web sites. Tags are labels (word or phrase) that a user can associate to a Web resource (URL, photo, etc.) [2]. Users are not limited to using controlled vocabularies as their tags. The tags clouds are collection of non-structured and non-hierarchical tags associated to a resource (e.g., as shown in Figure 1).

Collaborative tagging system is described in [3] as the practice allowing anyone to freely attach tags to resources. Besides, it offers an interesting alternative to current efforts at semantics Web ontologies. Such tagging system is also considered useful not only in the environment where moderator or controlling authority is not required but also when there are too many resources or contents to classify. The same research work argues that, from a user perspective, navigating a tagging system is similar to conducting keyword-based searches. The study and analysis conducted in [5] illustrate the strengths of cooperative tagging systems and present some axes of future research related to tagging systems.

According to [4], collaborative tagging systems usually hold some specific characteristics. We present, in this section, the ones which are useful to further understanding of our proposal.

- “tagging right” is the system’s restriction to group tagging. A system can be self-tagging where users can only tag their own resources (e.g., Flickr); or free-for-all tagging, where anyone can tag any resource (e.g., Yahoo! Podcasts).

- “tagging support” defines the mechanisms of tags entry.

There are three types of support. Blind-tagging does not
allow users to view tags associated to the same resources by others (e.g., Del.icio.us). Viewable tagging allows users to view the tags already attached to a resource and Suggestive tagging recommends possible tags to the user.

- aggregation refers to how tags are aggregated around a resource. Bag-model aggregation allows many occurrences of a tag to be attached to the same resource. Set-model avoids multiplicity of a tag around the same resource.

We believe, after analyzing thoroughly [2], [3], [4] and [5], that collaborative tagging systems provide benefits in enhancing metadata of a resource (a Web service in our case) and consequently facilitate a socialized process of Web services’ discovery, without having to extend the initial content of their descriptions in WSDL.

As we interpret users’ feedbacks on a Web service as tags in this work, it is interesting for us to explore and analyze some significant existing works related to handling users’ preference within a Web service discovery process.

In [6], a framework based on Implicit Culture is proposed to solve the Web service discovery problem. This approach helps users to discover Web services through a recommendation system base on past users’ experiences. However, every action made by the system’s users is logged in a remote client agent that each client has to install to communicate with a server located in a remote site. Despite being an interesting technique, the system overload can still be its main drawback when the number of communicating sessions increases. The related works presented in [7] and [8] depend on machine learning approach to improve the discovery result by using user’s preference. The principal handicap of these approaches lies on the number of examples (users’ preferences) to feed the system before they return interesting results. The authors of [9] presents an approach enhancing Web services discovery by matching users’ preferences with Web services’ capacities. Both elements are represented as preference ontology concept and capacity ontology concept in the format of customized SAWSDL [19]. The similarity between a user’s preference and a Web service’s capacity is the result of both concepts matching. In this case, users have restricted flexibility to formulate their preferences and published Web services’ descriptions must be modified by adding semantic annotations expressed in SAWSDL [19]. The latter is the core hindrance to the proposed technique because it takes a lot of efforts and expertise from services’ providers to enrich their published services with outside semantic annotations. The work in [10] proposes a method taking into account the user preferences and dislikes and transforms them into contexts. The latter is then modeled using description logics. A service’s context is described by a set of parameters related to the quality of service (QoS) such as cost, response time, reliability, availability and reputation. The similarity computing between user’s context and service’s context is based on the method used to determine the similarity among entity classes from different ontologies. This proposal suffers from the same issues discussed above: the restricted flexibility for a user to express his/her context and the investment needed to model services’ contexts. Moreover, QoS parameters used to describe service’s context must be negotiated and standardized among different service providers.

In the same attempt to exploit human needs to leverage the Web service discovery process, [11] represents the users’ needs based on Maslow’s hierarchy of needs (physiological, safety, love, esteem and self-actualization). The needs’ ontologies are then defined to help in users’ needs – service matchmaking. However, Web services’ descriptions must be, before all, annotated by rich semantic descriptions, which can be error-prone, time-consuming and demands grand effort from service’s providers to realize. The same cost and effort in extending Web services’ descriptions can be seen in [12] as well.

After learning from the drawbacks of several Web service discovery techniques discussed in this section and being aware of various advantages provided by collaborative tagging systems in this Web 2.0 era, we attempt to propose a novel Web services discovery mechanism based on collaborative tagging system. We believe that such system will be beneficial for both users and services’ providers in terms of facility of use, flexibility of request and slight effort to modify/extend the descriptions of published Web services. Our proposal is discussed in detail within the next section.

III. OUR PROPOSAL

First of all, our proposal is based on the collaborative tagging environment and encouraged by [13], [14] and [15]. According to [13], despite the use of uncontrolled and subjective vocabulary in the process of tagging a Web resource, collective tagging systems can help their users organize and manage their own or shared resources. It is then perceived as an informal classification system that is able to categorize, basing only on free-form keywords, Web resources by their functionalities. This research work claims that collaborative tagging can be used to aid in the resource discovery task. The authors of [13] have developed a probabilistic model exploiting collective tagging to help discover Web documents. The interesting performance of this system has been compared to the one of Del.icio.us [29]. In [14], social/collective annotations/tags are considered as emergent useful information that can be exploited in different ways to help improving the Web search. They expressed that social annotations can optimize the Web search in two aspects: i) the annotations are usually good summaries of corresponding Web pages; and ii) the count of annotations indicates the popularity of Web pages. As to [15], social annotations or tags are used to discover the social interests of Web users. This work points out that the user-generated tags attached to a Web resource reveals that tags are consistent with the web content they are associated to, while more concise and closer to the understanding and judgments of human users about the content.

We so far believe that social annotations or collaborative tagging systems can be a key enhancement to the discovery of Web services due to the fact that we will focus our work on the
search for Web service on the World Wide Web environment.

The system we are going to propose can be basically illustrated as following (see Figure 2):

---

**Figure 2. Basic representation of proposed system**

Users of our system behave almost the same way as those who use other collaborative tagging systems. Ideally, before tagging a resource (referring to a Web service hereafter), it should be used or tested so that users can describe or judge on its functionality, capability and other related qualitative aspects (cost of invocation, response time, etc.). The system offers a virtual *Web Service Collection* that represents the set of Web services collectable and accessible on the Internet. In other word, we can claim that the Internet feeds our virtual *Web Service Collection*. We store the physical representation of a resource which is the URL pointing to the WSDL description of a Web service in this collection. Interacting with this collection provides transparency to users as if they are in the real Web environment to execute their actions.

However, due to time and technical constraints, only the tagging and discovery operation on Web service will be treated in the remainder of this work.

A. System’s tagging relative characteristics

We allow two types of tags: keyword tag (single or composed) and free-text tag. Our system’s tagging right is *free-for-all* due to the fact that a tagged resource is made accessible and shared among many users. Our tagging support is *viewable tagging* because a user may tag a resource with an already displayed tag. We consider the *bag-model tags aggregation* as we are convinced that if a resource is tagged many times by different users with the same tag, it is evident that the used tag becomes more and more significant and representative for that resource.

The system proposes two exclusive entry points for the users to express their discoveries’ queries: by keywords or by free-text. Within the whole proposed mechanism tags are treated as non case sensitive.

B. Tagging and discovery by keywords

A Web service can be tagged by different keywords provided by different users. A tag weight refers to the occurrence number of a specific tag associated to a Web service.

The larger this weight is, the more important the Web service becomes, considering the related tag. All tags entered to the system form the *tag-collection*. Each tag in the *tag-collection* is associated to a certain number of resources, forming *resource vector*.

In the case of single keyword tagging, every tag is considered independent from another. We, however, propose 3 combiners to help users formulating advanced queries: AND, OR and NOT. The basic algorithm used to discover tagged Web services is presented in as below in Table 1. We use WordNet [20] to help us in computing the synonym’s relation and set corresponding to a tag in a query.

---

**TABLE 1**

**Algorithm for Search by Single Keyword Tag**

Let

- Q: a query tag.
- TagC: a tag collection.
- t: a tag in the tag collection.
- WSvect: the vector of resources associated to a tag.

**Step 1: Exact_matching(Q)**

Foreach t in TagC do { 

If (IsSameString(Q, t)) { 

return WSvect; 

}

}

**Step 2: Synonym_matching(Q)**

If (IsEmpty(WSvect)) { 

Let

synset: a synonym set obtained from WordNet given a tag query Q.

syn: a synonym in synset.

WSvectInit: an empty resource vector 

Foreach syn in synset do {

WSvectInit=Union(WSvectInit, Exact_matching(syn));
}

return WSvectInit;

}

// Step 1 and Step 2 are realized exclusively

The result of a discovery based on single keyword is ranked according to the importance of tag weight associated to the resources.

The combiner AND is used to filter the discovery result by intersecting different result sets related to each tag together. The OR combiner is used to compute the union of result sets and the NOT combiner allows to compute the difference between result sets.

For instance, let us consider separately the result of two discoveries by keywords “Tag_i” and “Tag_j” as follows:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Web service</th>
<th>Tag weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag_i</td>
<td>Web_Service_1</td>
<td>Tag_weight_1</td>
</tr>
<tr>
<td>Tag_i</td>
<td>Web_Service_2</td>
<td>Tag_weight_2</td>
</tr>
<tr>
<td>Tag_i</td>
<td>Web_Service_3</td>
<td>Tag_weight_3</td>
</tr>
<tr>
<td>Tag_j</td>
<td>Web_Service_2</td>
<td>Tag_weight_A</td>
</tr>
</tbody>
</table>

---
The result of a query “Tag_i AND Tag_j” can be represented this way:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Web service</th>
<th>Tag weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag_i</td>
<td>Web_Service_2</td>
<td>Tag_weight_2</td>
</tr>
<tr>
<td>Tag_i</td>
<td>Web_Service_3</td>
<td>Tag_weight_3</td>
</tr>
<tr>
<td>Tag_j</td>
<td>Web_Service_2</td>
<td>Tag_weight_A</td>
</tr>
<tr>
<td>Tag_j</td>
<td>Web_Service_3</td>
<td>Tag_weight_B</td>
</tr>
</tbody>
</table>

The result of a query “Tag_i OR Tag_j” can be represented this way:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Web service</th>
<th>Tag weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag_i</td>
<td>Web_Service_1</td>
<td>Tag_weight_1</td>
</tr>
<tr>
<td>Tag_i</td>
<td>Web_Service_2</td>
<td>Tag_weight_2</td>
</tr>
<tr>
<td>Tag_i</td>
<td>Web_Service_3</td>
<td>Tag_weight_3</td>
</tr>
<tr>
<td>Tag_j</td>
<td>Web_Service_2</td>
<td>Tag_weight_A</td>
</tr>
<tr>
<td>Tag_j</td>
<td>Web_Service_3</td>
<td>Tag_weight_B</td>
</tr>
<tr>
<td>Tag_j</td>
<td>Web_Service_5</td>
<td>Tag_weight_C</td>
</tr>
</tbody>
</table>

The result of a query “NOT Tag_i” is the set of web services which does not contain the services which are present in the result set of the query “Tag_i”.

Users can tag and discover a resource using composed keyword tags. This kind of tag can be recognized as a single string (concatenation of keywords). The discovery process respects almost the same algorithm found in Table 1 except that there is not any proposal yet within this current work on synonym matching for search by composed keyword tags.

C. Tagging and discovery by free texts

To give more freedom and flexibility in expressing a Web service discovery request, we propose to users the possibility to formulate their queries as a free text. The tagging process must be performed using free text as well. In this case, a resource is associated to a number of text tags. In order to compute the similarity between a free text query and a texts cloud associated to a resource, we transform them into vector of terms. Before the transformation into vector, text tags are appended one to another in an arbitrary order to form an only text called aggregated text tag.

We are encouraged by the performance of Vector Space Model [16, 17], a search technique very common in information retrieval systems allowing us to measure the similarity between pairs of documents. Recently, this model was used to implement the work of [18] also. Within this type of discovery, we use the Porter stemming algorithm [21] to extract terms vector from a query document and the aggregated text tag associated to a resource. The similarity of a query and an aggregated text tag is the cosine value [16, 18] between the two vectors representing both texts (see Table 2). The result will be ranked by the values of cosine coefficient between the query text and all aggregated text tags associated to resources.

---

### Table 2

**Algorithm for search by free text tag**

Let $Tq$: a free text query.  
$ResVect$: a vector of resources.  
$vQ$: query terms vector  
$vT$: aggregated text tag terms vector

Foreach $WS$ in ResVect do {  
$vQ=PorterStemming(Tq)$;  
$vT=PorterStemming(getAggregatedText(WS))$;  
similarity=cosine($vQ$, $vT$);  }

---

IV. USAGE CONTEXT OF THE PROPOSED SYSTEM

In addition to the advantages of our system presented above, we can identify two contexts in which our system can be very useful.

Firstly, the system can be employed as an independent Web service discovery engine using different existing resources (Web service collection). Because of the fact that our system is based on the collaborative tagging environment, it can be, as long as there are tags coming from users, deployed on the Internet as well as within a more restricted structure such as enterprise, working group, etc.

Secondly, we can employ such approach as a complement to other usual Web service discovery techniques seen in the literature to help filter their output based on user-centric and community feedback information. Therefore, instead of processing with a large Web services collection, the system has a more tailored input which is a subset of result providing by habitual Web service discovery mechanisms. Such usage can be found in [36] which is a more global and parent project of the currently described approach, having a goal as solving certain common issues of Web service discovery.

V. IMPLEMENTATION AND EVALUATION OF THE PROPOSAL

A first prototype of our proposal is being implemented as a Java Web based application. We will be conducting an evaluation within a small team of graduate students and we are planning to use the set of Web services from [18] and QWS dataset [22]. In effect, these two collections of Web services allows us to make a step forward without having to develop a Web service crawler from scratch to collect and feed our Web service collection. Due to time constraint, only English will be firstly used as the query and tagging language in this implementation. Instead of using only the cosine coefficient to measure the similarity between a free-text query and an aggregated text tag, we also plan to benchmark our application on different other similarity coefficients available to use in Vector Space Model (e.g., Jaccard, Dice, etc).

The system will be also validated through these following points:
tag’s quantity allows us to see the impact related to the discovery quality and the popularity of a service considering the used tags.

- recall/precision ratio let us improve our algorithm’s performance and the way we acquire our Web service’s instances.
- tag’s disambiguation needs to be taken into account related to the fact that our tags are natural language based.

At the end of the implementation, we should be able to determine the minimum participation in the tagging (tag’s quantity) in order to make the best use of tags within the Web service discovery process. Recall and precision ratio will help us in choosing an appropriate “query – annotation similarity” measurement (cosine, Dice, Jaccard, etc.). Finally, tag’s disambiguation will have to be studied and put in place to improve the quality of tags and queries.

VI. CONCLUSION AND FUTURE WORK

In this paper we presented a Web service discovery environment based on collaborative tagging model using keywords or free texts. The implementation of such environment benefits users in terms of facility and flexibility to express a query (natural language) and does not demand any effort or investment from the service’s providers to semantically enrich or modify their already published services. This kind of system is believed to make Web service discovery a socialized and at-a-finger-tip task.

Such system works efficiently if there are a good number of users participating in the tagging process and if users run a qualitative test on a Web service before annotating it. This will allow users to understand more clearly on the service’s functionality and applicability. This kind of system can be used inside a community of SOAs (Service Oriented Application)’ designers or developers using, sharing and recommending mutually the Web services within the community. Besides, it is legitimate to state that this discovery mechanism based on collaborative tagging system that we are proposing can be implemented and used as a whole search system as well as a complementary approach to the usual ones that we can see in the literature. In our study case, users tag the existing Web services found on the Internet to better discover them later and to share them within the community. Thus, the system is considered as entirely a whole one though it uses the existing resources on the Web. Despite all, it can be used as complement to other usual Web service discovery technique to help refine the result based on user’s tags or feedback.

The absence of tags associated to a resource can be considered as a primary barrier to the evaluation of our approach. The remedy to this issue is to generate automatically the tags from the corresponding WSDL file and attach them to the service. Further study on this mechanism will concentrate on using Natural Language Processing and Information Retrieval techniques to extract terms from a structured document. Meanwhile, many other known problems related to collaborative tagging characteristics of our system remain unsolved in our proposal: tag bombing, tag spamming, tag reputation and quality. Ulterior study is focusing on improving the search and tagging by composed keywords, providing better synonyms processing among terms and dealing with tags ambiguity. Synonyms processing will be, in the same goal, included in the discovery by free text to ameliorate the search result. Furthermore, we will conduct our study into the use of structure and controlled vocabularies in our tagging system because we will recommend a tagging system with minimum structure and vocabularies to improve the quality of the Web service discovering by, at any means, preserving the easy and flexible query’s construction.

The current presented system opens different points of extension:

- providing an online testing interface to use or test a Web service represents an important pre-requisite to enable the good quality tagging or annotation.
- testing procedure, scenario and guide should be provided automatically by the system to facilitate the users in describing afterwards the functionality and capability of services.
- offering possibility to system’s users to add services to the Web service collection, tag it and finally make it available to share with other users in the system. This operation will help feed the Web service collection with services found, used or tested by users and recommended to others .

We are at the meantime set on exploring novel solutions and studying thoroughly on users and resources management in such system and finally we plan to propose a recommendation system for Web service discovery based on tags.

REFERENCES


