Social Web Services Discovery: A Community-Based Approach

Abdelmalek Metrouh
LAMIS Laboratory, University of Tebessa – Algeria
University of Badji Mokhtar, Annaba - Algeria
abdelmalek.metrouh@yahoo.fr

Farid Mokhati
LAMIS Laboratory, University of Tebessa – Algeria
University of Oum El Bouaghi - Algeria
mokhati@yahoo.fr

ABSTRACT

Over the past few years, an exceptional interest has been taken in the area of Service-Oriented Computing. Particularly, a remarkable effort has been made in the context of Web service discovery, a very important and active research domain. In fact, the number of Web services has grown rapidly and the task of their discovery resting on standards, UDDI and ebXML becomes more and more difficult. The most proposed approaches for Web services discovery focused on the description of Web services themselves and neglect their interaction with each other. In this paper, we use the concepts of social networking, the principles of recommender systems and Web services communities, in the context of Web 2.0, introduced in our previous work to significantly reduce this task. The obtained results seem promising.

Categories and Subject Descriptors

H.3.5 **[Information storage and retrieval]**: Online Information Services – *Web-based services*.

General Terms

Algorithms, Experimentation.

Keywords

Web Services, Social Web Services, Social Networks, Discovery, Recommendation, Community.

1. INTRODUCTION

Service Oriented Computing (SOC) has emerged in recent years as a powerful paradigm for assembling complex Web applications from distributed simpler application components known as services [15]. Web services and social networks are emerging in which applications can search and compose services according to users' needs in a seamless and an automatic fashion [10]. A wide range of similar functionalities are expected to be offered by a vast number of Web services. Web services are expected to outsource some of their functionalities to other Web services [10]. Some services may be new to the service market. To ensure the discovery and retrieval of suitable Web services, it is essential to provide mechanisms for that.

Currently, dynamic Web services discovery is managed with registries allowing their definition, storage and discovery. One of the two main standards is based on the UDDI[1] (Universal Description Discovery and Integration) protocol, which is a registry listing exclusively Web services based on WSDL[2] (Web Services Description Language) for their descriptions. There exists another standard with more extended aiming for the e-business named ebXML (Electronic Business using eXtensible Markup Language) that covers all the paradigms proposed by UDDI and WSDL. Based on UDDI or ebXML, the Web services discovery algorithms use a research technique with key words or tables of correspondence of couples (keys, values), which is very restrictive, too undoubtedly. Moreover, this technique prevents the possibility of combining Web Services. There is a further difficulty: it would require a software agent that can monitor the registry is actually updated.

As part of this paper we propose a significant improvement to our algorithm for Web services discovery proposed in our previous work [12]. The obtained results are promising and allow us to highlight the contribution of social networks in the task of social Web services discovery. The use of the community notion in Web service discovery is not new. In recent years, some community-based approaches ([6], [7], [12], [19]) have been proposed in the literature in order to deal with Web service discovery in the context of social networks. Those works have considerably forwarded the domain by proposing novel strategies for Web services discovery.

The remainder of this paper is organized as follows. Section 2 introduced the concept of Web service. Section 3 gives an overview on the concept of Web service community. State of the art on Web services discovery is given in section 4. Section 5 describes our approach for social Web services discovery based on communities. Our implementation is discussed in section 6. Conclusion and future works are given in section 7.

2. WEB SERVICE

A Web service[3] is defined as a self-contained, self-describing software application that can be advertised, located, and used across the Web using a set of standards [14]. Web services are with public interfaces described in XML[4]. According to World Wide Web Consortium (W3C) [18], Web services identified by a Uniform Resource Identifier (URI), their interfaces are defined in Web Service Description Language (WSDL), published in the

[1] http://www.uddi.org/

[2] http://www.w3.org/TR/wsdl

[3] http://www.w3.org/TR/ws-arch/

[4] http://www.w3.org/XML/

Universal Description, Discovery and Integration (UDDI) directory, Web services can be discovered and invoked by other software systems. These systems interact with Web services using XML-based messages conveyed by Simple Object Access Protocol (SOAP[5]). Web services like any other middleware technologies; aim to provide mechanisms to bridge heterogeneous platforms, allowing data to flow across various programs [17].

3. WEB SERVICE COMMUNITY

Several definitions have been proposed in the literature. In [7], Maamar et al. considered a community as a means to provide a common description of a desired functionality without referring explicitly to a specific Web service. In [19], the authors discussed a framework to manage Web services using the concept of community and the social networking. A community gathers similarly-functional Web services together and social networks establish relationships between Web services. Relationships like supervision, competition, and substitution are intra-community, while others like collaboration and recommendation are inter-community. In [6], the authors discussed the binding of Web services communities to social qualities like selfishness, trustworthiness, and spitefulness. They defined a community as a virtual space that groups Web services together subject to offering similar functionalities. In [13], the authors proposed that communities can dynamically be created and Web services can at run time be associated to these communities, in the context of social networks. In [12], our previous work, we defined the notion of Web services community as relationship between Web services without any particular weight.

In this paper, we describe informally, the notion of Web services community as relationship between Web services characterized by weights (house construction, hotel reservation). Besides, a Web service may belong to different communities. Our approach for the construction of Web services communities is detailed in section 5.

4. SERVICE DISCOVERY OVERVIEW

4.1 Web Service Discovery

Web service discovery is the process of finding Web services, which satisfy specific requirements. Discovery is achieved by searching the service with a matching description and service profile on the UDDI. Due to the increase of the Web services availability with similar functionalities on the Web, this drives the need for more complicated Web service discovery mechanisms [4].

The Web services discovery algorithms using UDDI or ebXML and based on the syntactic correspondence are very restrictive. Two semantic approaches have been proposed in the literature, the semantic annotation [1, 3, 11] and the Web services ontology. Two models of services ontologies are proposed OWL-S (Ontology Web Language for Services) and WSMO (Web Service Modeling Ontology).

4.2 Social Web service Discovery

4.2.1 Social Networks Overview

Social networks currently play an important role in the life of millions of active Internet users; over the last few years, interest in social networks Websites such as Facebook[6], MySpace[7], Friendster[8], Twitter[9] and LinkedIn[10] has increased considerably, and the popularity of such sites has increased significantly [2]. Social networks have recently been receiving increased research attention; Social networks are online communities: people, teams, or any social group [16] interacting on the web, and connected via social relationships, such as co-working, friendship or knowledge exchange in varied contexts such as entertainment, politics, religion, dating or business.

4.2.2 Social Web services

The social networks of Web services or social Web services differ from conventional social networks. Social networks are based on the absolute cooperation and mutual assistance between their members (i.e., no competition). By cons, Web services in social networks are especially competitive [8]. Web services are the only constituents of the social network. Social networks are used to represent the interactions between Web services in the aim of their discovery or selection. Social networks of Web services are some of the recent approaches introduced in the field of Web services discovery ([8], [5], [9], [12]).

4.2.3 Discovery

In [17], the authors used recommendation-based associations that could enrich the process of composition with additional Web services. They also used these associations to allow direct selection (and possibly automatic) of a Web service, which will replace a failed Web service. In [9], the authors introduced a semantic dimension for calculating the similarity between Web services in the same spirit as the work proposed in [8]. In [5], the authors introduced the concept of Web services competition in addition of the concepts discussed in [8] to build social networks for Web services discovery. In [12], the authors proposed in addition to the collaboration-based associations to build Web services communities, recommendation-based associations to define a Web services discovery process.

5. PROPOSED APPROACH FOR SOCIAL WEB SERVICES DISCOVERY

We present in this section our approach for the social Web services discovery process. This process rests on the creation of a social network of Web services for one or more registers UDDI. The social network associated with Web services is defined by two types of associations between these Web services: collaboration-based associations that help to define the notion of Web services community and thereby reduce the search space of these latter, and recommendation-based associations that allow us to target only the Web services that have been retained in their previous participation in response to user queries. It is neither substitution, nor the recommendation of additional Web services proposed by Maamar et al. in [8].

---

[5] http://www.w3.org/TR/soap/

[6] http://www.facebook.com

[7] http://www.myspace.com

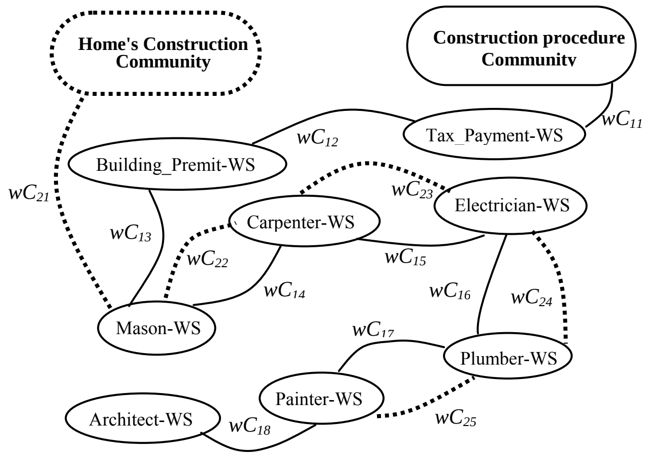
[8] http://www.friendster.com

[9] http://twitter.com

[10] http://www.linkedin.com

5.1 Building social networks of Web services

Web services are the only constituents of the social network and designate the nodes.



wC_{ij} : the Web service (WS_j) *C*ollaboration *w*eight defined by the equation in (1)

$wC_{11} >= wC_{12} >= wC_{13} >= wC_{14} >= wC_{15} >= wC_{16} >= wC_{17} >= wC_{18}$

$wC_{21} >= wC_{22} >= wC_{23} >= wC_{24} >= wC_{25}$

Figure 1. Example of Web services communities definition in a social networks of Web services.

5.1.1 Collaboration-based association (C)

Collaboration based associations (*C*) define the notion of Web services community. Each community is defined by a tree of Web services; "*Construction procedure community*" in Figure 1, for example. These associations are evaluated by the following equation:

$$wC_i(WS_i) = \frac{|WS_i\, selection|}{|WS_i\, participation|} \qquad (1)$$

$|WS_i\ selection|$ represents the number of times that WS_i has really participated in scenarios of composition, so selected by the client. $|WS_i\ participation|$ is the number of times that WS_i has been appointed to participate in these scenarios of composition. The tree associated with each community is sorted in descending order of the weights associated to the associations (Figure 1).

5.1.2 Recommendation-based Association (R)

In the same Web services community, a Web service could propose that new Web services resulting from recommendation-based associations should be part of a composition. Web services resulting from recommendation-based associations may be required to satisfy a user query. Example: the Web service *Mason-SW* will recommend that *Carpenter-SW* could be part of the composition scenario with a weight wR_{12}. *Carpenter-SW* could also recommend *Mason-SW* with a weight wR_{21}. The weight of the recommendation-based association wR_{12} and wR_{21} are given by the following equation:

$$wR_{ij}(WS_i, WS_j) = \frac{|WS_J\, selection|}{|WS_i\, participation|} \qquad (2)$$

$|WS_i\ participation|$ and $|WS_j\ selection|$ represent the number of times that WS_i participated in scenarios composition and number of times that WS_j has been appointed by WS_i to participate in these scenarios of composition. This equation (2) is inspired by the works of Maamar et al. in [8] and Metrouh et al. in [12].

5.2 Algorithm for Social Web services discovery

Formally, we define the graph *G* as a triple *(S,C,R)*, where *S* is the set of vertices, *C* is the set of edges of *C*ollaboration and *R* is the set of arcs of *R*ecommendation. Furthermore, the graphs formed by *S* and *C*, on the one hand, and *S* and *R*, on the other hand, are respectively called *cG* (*G*raph of *c*ollaboration) and *rG* (*G*raph of *r*ecommendation) of *G*. We also define an undirected graph *rG'=(S,R')* corresponding to the directed graph *rG=(S,R)* whose edges *R'* are valued by the equation (3). WS_i and WS_j are two Web services of the graph *rG'*. The weight $wR(WS_i,WS_j)$ of the edge (WS_i,WS_j) is given by the following equation:

$$wR(WS_i, WS_j) = \frac{wR_i(WS_i, WS_j) + wR_j(WS_j, WS_i)}{2} \qquad (3)$$

ALGORITHM Social Web services Discovery

{

INPUTS

G = (S, C, R), where *S*: the set of nodes or Web services, *C*: the set of arcs collaboration and *R*: the set of arcs of recommendation

cG= (S, C) graph of the collaboration-based associations;

rG= (S, R) graph of the recommendation-based associations;

rG'= (S, R') undirected a nd weighted graph resulting from the directed and weighted graph *rG (S, R)*;

OUTPUT *WS_{dis} set of the discovered social Web services;*

$WS_{dis} := \emptyset$;
Mark the first vertex *x* of the graph *cG=(S, C)* associated with the community extracted from the user request;
Mark the vertex *x* in the graph *rG'=(S, R')*;
$WS_{dis} \leftarrow x$;

WHILE exists an adjacent vertex not marked at a vertex in *rG'= (S, R')* **DO**

Select a vertex *y* not marked and adjacent at a marked vertex *x*;
IF *(x, y)* is the outgoing edge of the highest weight **AND** $y \in$ to the tree associated with the community extracted from the user request of the graph *cG*
THEN
$WS_{dis} \leftarrow WS_{dis} \cup \{y\}$;
Mark *y*;
END IF

END WHILE

Return SW_{dis};

}

Figure 2. Algorithm for social Web services discovery process based on communities.

The contribution of this algorithm compared to our previous work [12], is that the collaboration-based associations are characterized by weights which allow us to sort the Web services in a community of Web services in a descending order. This will allow us to consider only the Web services with high weights, once the social network reached a large size. This will also avoid us a random selection of the first Web service in response to a client query, proposed in [12] and thus improve the process of Web services discovery.

The idea of the proposed algorithm illustrated in Figure 2 is to maintain a connected subgraph, by connecting a new vertex at each step. An idea largely inspired by Prim's algorithm which is a greedy algorithm that finds a minimum spanning tree for a connected weighted undirected graph. At each iteration, the algorithm will add to the tree a vertex, until it covers all the vertices of the graph *rG'*. The vertices must belong also added to the tree associated with the community extracted from the graph of collaboration *cG*. The proposed algorithm constructs a spanning tree only on the connected component of the vertex initially marked in the graph *rG'*. If at one stage a set *V* of vertices are connected together, to select the next vertex to be connected, in a spanning tree, there exists necessarily an edge which connects one of the vertices of *V* with a vertex apart from *V*. To build a maximum spanning tree, simply select from the outgoing edges witch has the highest weight. To detect the outgoing edges, we can progressively mark the already connected vertices. An outgoing edge then connects necessarily a marked vertex and an unmarked vertex.

6. IMPLEMENTATION AND EVALUATION

In this section, we describe the experimental results we obtained by running our algorithm. The algorithm is implemented in Java v1.5 and run under the operating system Windows XP Professional. CPU times were obtained on a 3.6 GHz Pentium 4 with 1 GB of RAM. The following table "Table 1" shows the time taken by our algorithm to return the set of the discovered Web services. The Graphs chosen are random in the sense that the number of vertices and the number of edges are chosen arbitrarily. It is the same for the weights of the recommendation-based association in the graph *rG'*, since we need data at the beginning which avoids us the problem of "cold start" not addressed in the context of this paper. The implementation provided in this paper uses a priority queue though it can be easily extended using heap implementation. The time is measured in milliseconds.

Table 1. Time taken to return the set of the discovered Web services

No	Graph *G*		Graph *cG*		Graph *rG'*		Time Taken (Milliseconds)
	Nodes	Edges	Nodes	Edges	Nodes	Edges	
1	240	350	150	250	50	150	135
2	350	550	200	300	100	230	395
3	400	750	250	350	150	340	1008
4	600	850	400	450	200	510	1579
5	650	950	450	500	300	750	3958
6	850	1700	600	1500	400	1100	6097
7	950	2200	700	1800	500	1500	12242
8	1100	2800	800	2400	600	1900	16692
9	1250	3300	1000	2800	700	2200	18839
10	1450	3500	1200	3000	800	2400	25905
11	1650	4000	1400	3500	1000	2700	50457
12	1850	4500	1600	4000	1200	3000	68403
13	2100	5000	1800	4500	1400	3600	134211

The priority queue node contains the vertex name witch refers a Web service, its parent and the weight of the edge between the parent and the vertex. The weight of the edge is given by the equation (3).

Calculating running time:

Total time = Time to build the priority queue + Time to go through the vertex listing and update all items in the queue till all the vertices of the connected subgraph of recommendation *rG* are in the tree + time to check that the vertex belongs to the connected subgraph of collaboration *cG*

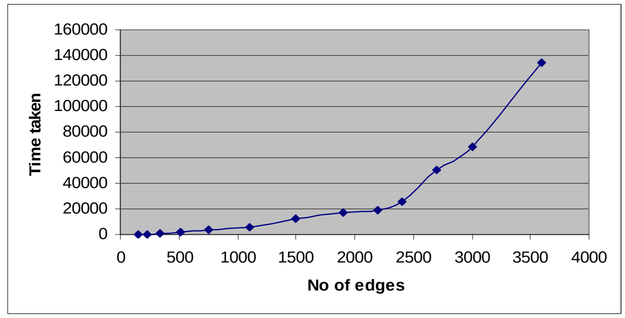


Figure 3. Time Vs N° of edges in the graph of recommendation

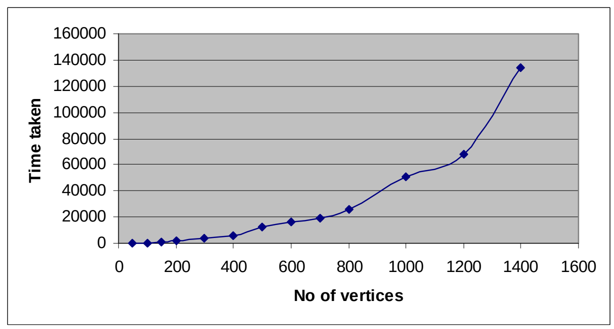


Figure 4. Time Vs N° of vertices in the graph of recommendation

Till 200 vertices the graph in "Figure 4" is almost linear and then it forms the "nlogn" curve. From the analysis we can easily conclude that the proposed algorithm takes more time as the number of vertices increase in the graph of recommendation rG. This could be influenced by the way in which was implemented (using a linked list version of priority queue and not a heap). The number of edges in the graph of recommendation rG also has an important influence over the running time "Figure 3".

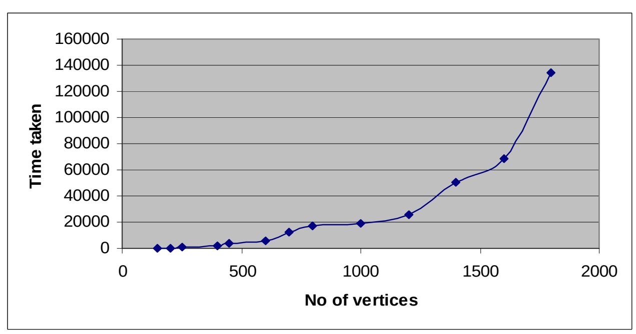


Figure 5. Time Vs N° of vertices in the graph of collaboration

We can notice also in "Figure 5" that the number of vertices in the graph of collaboration cG has no important influence on the running time. Seen its simple structure and seen that it allows only verifying that the vertex belongs effectively to the community extracted from the user request. The number of edges is also without important influence on the running time. The edges considered in the graph of recommendation rG constitute a tree of simple structure as shown in "Figure 1".

7. CONCLUSION AND FUTURE WORKS

In this paper, we propose an algorithm for social Web services discovery in the context of the Web 2.0. We have defined two types of associations in a social network of Web services. Collaboration-based associations formed from Web services communities, characterized by weights and sorted in the descending order and recommendation-based associations. The combined exploitation of these two types of associations allowed us to reduce considerably the task of Web services discovery. Different experiments were carried out to illustrate our approach for social Web services discovery.

The current work could be extended by introducing a semantic dimension in defining Web services communities. We could also review the recommendation-based associations and suggest that the users take part more in the process of recommendation. We should also face another problem, the "cold start" that occurs at the beginning of the use of the system, in critical situations where the system lacks data to make a custom filtering quality.

8. REFERENCES

[1] Akkiraju, R., Farrell, J., Miller, J., Nagarajan, M., Schmidt, M.T., Sheth, A. and Verma, K. 2005. Web service semantics – WSDL-S. W3C Member Submission 7, November 2005.

[2] Aroyo, L., De Meo, P. and Ursino, D. 2010. Trust and reputation in social Internetworking systems. In *Proceedings of the Workshop on Adaptation in Social and Semantic Web* (Aachen, Germany. June 21st, 2010). SASWeb 2010. CEUR Workshop Proceedings, 590, 42–53.

[3] Farrell, J. and Lausen, H. 2007. Semantic annotations for WSDL and XML schema. *W3C recommendation*, Available from: <http://www.w3.org/TR/sawsdl/>, last visited in June 1st, 2013.

[4] of Semantic Web Service Discovery Methods. In *Proceedings of the 6th World Congress on Services* (SERVICES-2010), 176-7 (2010). IEEE Computer Society Press. ISBN: 978-0-7695-4129-7.

[5] Maamar, Z., dos Santos, P. B.,Wives, L. K., Badr, Y., Faci, N. and de Oliveira, J. P. M. 2011. Using Social Networks for Web Services Discovery. *IEEE Internet Computing.* 15, 4 (2011), 48-54.

[6] Maamar, Z., Faci, N., Ejub, K., Pljasković, A., Ugljanin, E. and Fahim Akhter, F. 2013. From Social Web Services to Social Communities of Web Services. *Advances in Information Sciences* (AIS). 1, 1 (2013). 22-37.

[7] Maamar, Z., Lahkima, M., Benslimane, D., Thirand, P., Sattanathan, S. 2007. Web Services Communities: Concepts & Operations. In *Procedings of the 3rd International Conference on Web Information Systems and Technologies* (Bacelona, Spain. March 3–6, 2007). WEBIST'07. 323-327.

[8] Maamar, Z., Wives, L.K., Badr, Y. and Elnaffar, S. 2009. Even Web Services Can Socialize: A New Service-Oriented Social Networking Model. In *Proceedings of the International Conference on Intelligent Networking and Collaborative Systems* (Barcelona, Spain, Nov. 8-9, 2009). INCOS '09. 24-30.

[9] Maamar, Z., Wives, L. K., Badr, Y., Elnaffar, S., Boukadi, K., Faci, N. 2011. LinkedWS: A novel Web services discovery model based on the Metaphor of social networks. *Simulation Modelling Practice and Theory.*19,1 (2011), 121-132.

[10] Malik, Z., Akbar, I. and Bouguettaya, A. 2009. Web Services Reputation Assessment Using a Hidden Markov Model. In *Proceedings of the 7th International Conference on Service Oriented Computing*. Lecture Notes in Computer Science. 5900 (2009). 576-591.

[11] METEOR-S. 2002. Semantic Web Services and Processes. LSDIS Lab, University of Georgia. Available from: http://lsdis.cs.uga.edu/projects/meteor-s/, last visited in June 1st, 2013.

[12] Metrouh, A., Seridi-Bouchelaghem, H. and Mokhati, F. 2012. Web Services Discovery - A Novel Social Networks Approach based on Communities. In *Procedings of the 14th International Conference of the Entreprise Information Systems* (Wroclaw, Poland, 28 June – 01 July, 2012). ICEIS 2012. 2, 316-319.

[13] Metrouh, A. and Monfort, V. 2013. Selecting Web Services "On the Fly" According to Dynamic Social Communities Creation. In *Proceedings of the 2nd International Workshop on Web Intelligence* (Angers, France, July 5, 2003). WEBI 2013. 82-89.

[14] Papazoglou, M.P., Traverso, P., Dustdar, S. and Leymann, F. 2007. Service oriented computing: State of the art and research challenges. *Computer*. 40, 11 (2007). 38-45.

[15] Papazoglou, M.P., Traverso, P., Dustdar, S. and Leymann, F. 2008. Service oriented computing: a research roadmap. *International Journal of Cooperative Information Systems*. 17, 2 (2008), 223-55.

[16] Shekarpour, S. and Katebi, S.D. 2009. Modeling and evaluation of trust with an extension in semantic Web. ***J. Web Sem.*** 8, 1 (2010). 26-36. DOI= http://dx.doi.org/10.1016/j.websem.2009.11.003

[17] Staab, S., van der Aalst W., and Benjamins, V. R.. 2003. Web services: been there, done that?. *IEEE Intelligent Systems*. 18, 1 (2003). 72–85.

[18] W3C. 2002. Web Services Description Requirements. W3C Working Draft, <http://www.w3.org/TR/ws-desc-reqs/>.

[19] Yahyaouia, H., Maamar, Z., Lim, E., Thiran, P. 2013. Towards a community-based, social network-driven framework for Web services management. *Future Generation Computer Systems*. 29, 6 (2013). 1363-137.