Link-based Approach to Web Service Discovery

Gopinath Ganapathy and Chellammal Surianarayanan

***Abstract*—The requirement of automated web service discovery arises in many Service Oriented Computing applications. According to the unique characteristics of rapidly growing huge number of web services over Internet and the various standards in which the services are being described which make automated service discovery challenging and cumbersome task, a study has been made to check whether link based approaches can be employed to extract the context of a web service and to find similar services during discovery. Initial results with typical data set show that a back-link based approach can very well be used to extract service context and improve the efficiency of service discovery by providing context information.**

Secondly, a backward or reverse approach is used to check the applicability of simple co-citation algorithm in finding similarity among web services. With a typical similar set of services, the 'recall' and number of false positives of the proposed method is 22.2% and 1. The observed false positive is due to the fact that some of the back-links pointing to services might be hub pages and those should not be considered for similarity during matching. With this resolution, number of false positives is 0. The number of false negatives is found as 0 as expected. Though the recall seems to be low as compared to a typical full key word matching method which has a recall of 88.8%, the back-links of services may be used in finding best referrers. It is claimed here that the best referrers might be used for filtering of discovered services based on hub score.

***Index Terms*—co-citation based service discovery, link based service context extraction, link based service similarity.**

I. Introduction

In Service Oriented Computing, despite considerable research efforts towards the standardization of web service protocols and languages to ease integration and composition of existing heterogeneous business solutions, automated web service discovery still remains a complex task. The complexity is mainly due to the rapid increase in number of web services available on the Web and the various concept models like Web Service Description Language (WSDL), Web Ontology Language for Services (OWL-S), and Web Service Description Language Semantics (WSDL-S), Web Service Modeling Language (WSML), etc with which the services are being exposed. Another major issue that exists with automation is that the discovery mechanism using Universal Description Discovery and Integration (UDDI). The UDDI registry based discovery supports only exact keyword based matching and category based querying which are inadequate to automate the service discovery. In view of the difficulties with UDDI registry method as described in [1] and [2], Web Service Description Language (WSDL) based discovery techniques have come up. In [2] a WSDL based approach had been described for template based web service discovery. The above methods serve well only for manual service discovery and these basic components of web services are not enough to deal with modern requirements, namely dynamic composition, flexible discovery, good initialization and selection of appropriate services [3].

In dynamic service composition where services are discovered and composed dynamically, traditional syntactic approaches meet serious difficulties when sufficiently enough multiple parties are involved [4]. With the advent of ontology and various semantic service matching algorithms [5], web services can be unambiguously interpreted and automatically discovered. How service capabilities can be described semantically using OWL-S has been discussed in [6]. Several research efforts have been proposed to extend the current service descriptions with explicit semantics to improve the service discovery and composition significantly. Even the accuracy of semantic methods is being improved with domain specific ontologies along with domain independent ontologies as in [7]. Another similar approach which uses domain independent English Thesaurus and domain specific ontology had been proposed in [8] for efficient search for services in repositories. In spite of high accuracy of semantic service discovery, at large, major critical issues that exist with web service composition are the number of services involved in meeting the user goal and scalability, as composing two services is not the same as composing ten services. In real time composite services like classic holiday booking scenario, the number of services interacting will be large and how a composition approach scale with number of services involved in meeting user goal is important. Compositional scalability of current approaches like Business Process Execution Language (BPEL) and OWL-S is only 'average' [9]. For any semantic reasoner such as RACER, the time required to discover a service with ten concepts is in the order of 4 to 5 seconds [10].

To overcome the scalability issue execution time involved in matching with semantic service discovery, presently efforts are being performed towards approaches such as OWLS-MX, a hybrid semantic web service matchmaker which uses both logic based reasoning and Information Retrieval (IR) techniques. Experiments with OWLS-MX matchmaker proved that hybrid approaches showed better performance that pure description logic based approaches. In [11], an algorithm had been proposed which uses a hybrid approach of semantic matching for service, operation and

Manuscript received March 18, 2010.
Gopinath Ganapathy is with the School of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, TN, 620 023, India (phone: 91-431-2331-663; fax: 91-431-2331-662; e-mail: gganapathy@gmail.com)
Chellammal Surianarayanan, Member IAENG (106383) is with the School of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, TN, 620 023, India. (e-mail: chelsrsd@rediffmail.com).

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

parameters and syntactic matching for inputs and outputs to produce better results. A goal oriented interactive service composition which takes into account the current context of the goal and user decision during composition had been proposed in [12]. A quality driven global planning composition model had been proposed in [13] which enables the achievement of the overall duration of the composite service execution should be minimized.

As Web is huge and rapidly growing, hyperlink structure based methods such as Selective Hypertext Induced Topic Search (SelHITS) which calculates hub and authority values [14] and PageSim, a novel link based similarity measure for information retrieval [15] are being explored for information retrieval. Further, link based similarity based methods are proposed in [16] and [17] to fight spam web pages. A survey with link based approaches show that they are extensively used in the area of information retrieval for finding similarity measure, spam web pages and ranking search results. In complement to applying link based methods to information retrieval, in this work a preliminary study has been performed for the first time to find whether link based approaches are useful in extracting context of a web service and in finding similarity among web services.

This paper is organized as follows. Section 1 presents introduction with current issues in automated service discovery and dynamic composition, Section 2 presents related work, Section 3 details context extraction, Section 4 describes how a simple co-citation based method can be applied to find similarity among web services, Section 5 discusses performance evaluations and Section 6 details the results and concludes the work.

II. Related Work

In [18] a circular two way context analysis has been performed in which each web service context is evaluated according to its proximity to other services and the proximity of each of the other services to the current service. It uses UDDI- Meta data of a service as textual initial context, constructs tokens from this initial context and extracts multiple contexts from World Wide Web with the constructed tokens. Results of the work indicate that context overlap can be used to identify candidate services for composition. A link based Page Rank ranking model for defining importance of services has been proposed in [19].

III. Context Extraction

This approach is based on the fact that every web service exposes its URL irrespective of the description standard. For example in the case of WSDL, the service Uniform Resource Locator (URL) is exposed using ‘service’ tag as given below

```
<service name="EmailValidation">
  <port name=" EmailValidation"
    binding="tns:XWebEmailValidation">
      <address location="http://ws.xwebservices.com/X
       WebEmailValidation/V2/EmailValidation.asmx"/>
  </port>
</service>
```

This URL forms the seed for context extraction. The back-links of the URL are found and then the relevant context is extracted by analyzing the contents of the back-links. The proposed method performs context extraction using the following method.

Method: generateContext

Input: Service description (for example a WSDL file)
Output: contextString
contextString=NULL
Step 1: If any <documentation> tag exists in the service description
For each <documentation> tag in ‘portType’ and ‘operation’
  Extract the contents of <documentation> tag
    contextString = contextString + portType/operation name + contents of <documentation> tag
End For
Return contextString
Step 2: If <documentation> tag is not present in the service description
1. Construct an initialContext with service and operation names and with individual words contained in them, such as initialContext extracted from the portType, ‘sendMessage’ is that initialContext={‘sendMessage’, ‘send’, ‘Message’}
2. Find the URL of the service from service description
3. Find the back-links of the URL
4. For each back-link
    If the word count of the contents in a back-link is less than typically say 50
      Add the contents as it is to the contextString
    Else
      Extract the context information by finding any word from ‘initialContext’ exists in the contents of back-link and if it exists then the sentence containing that word is added to the contextString
    EndIf
  End For
Return ‘contextString’

To test whether the proposed context extraction method produces useful results, two different WSDL files, one is with and another is without <documentation> tag have been chosen from “www.remotemethods.com” as test data. The test data include “http://www.abctext.com/webservices/SMS.asmx?WSDL”and “http://ws.xwebservices.com/XWebEmailValidation/V2/XWebEmailValidation.wsdl”. Firstly, as explained in the previous section, context extraction has been employed to “http://www.abctext.com/webservices/SMS.asmx?WSDL” as follows. This WSDL file has been scanned for documentation tag in ‘portType’ and ‘operation’ elements. It is seen that the above file does not contain documentation tag for portType. The contents of documentation tag for different operations are given below in Table 1

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 1. Context extracted from <documentation> tag

#	Operation name	Contents of documentation tag
1	QuickSend	Allows you to send a single SMS message instantly to many recipients
2	Send	Allows you to send multiple SMS messages, each with its own optional delivery time, to many recipients. String interface for non .Net clients
3	GetCreditBalance	Returns the number of credits remaining
4	GetMessageReceipientStatus	Returns the status of a message for a specified recipient number and user phone message id
5	IsSenderIdAvailable	Returns a boolean confirming whether a SenderId is available

The 'contextString' has been constructed with the above operation names and the contents of their documentation tags. Secondly, context extraction has been employed to "http://ws.xwebservices.com/XWebEmailValidation/V2/XWebEmailValidation.wsdl" as given below. For simplicity this service which contains only one operation has been taken up for explanation. The above WSDL file does not contain any <documentation> tag. From this file, the name 'portType' is found as "XWebEmailValidationInterface" and the name of the 'operation' is found as 'ValidateEmail'. With these names, the 'initialContext' has been arrived with removal of common terms like 'get', 'post', 'soap', 'http', interface'. The 'initialContext 'arrived is given as

initialContext = {XWebEmailValidation, ValidateEmail' 'x', 'web', 'email, 'validation, 'validate'}. The URL of the service has been found from WSDL file. With the help Yahoo Application Programming Interface (API), the back-links of the above service URL have been found and given below in Table 2

Table 2 Back-links of service URL

#	Back-links of service URL
1	http://www.xwebservices.com/Web_Services/XWebEmailValidation/
2	http://ws.xwebservices.com/XWebEmailValidation/V2/EmailValidation.asmx?op=ValidateEmail
3	http://service-repository.com/service/wsdl?id=98336
4	http://www.service-repository.com/service/newversioncheck?id=720900
5	http://www.esigma.com/telos/discover/servicedetail.html?reg=uddi:webservices.esigma.com:uddi&key=uddi:ws:xwebservices:com:XWebEmailValidation:V2:XWebEmailValidation
6	http://www.esigma.com/telos/discover/servicedetail.html?reg=uddi:webservices.esigma.com:uddi&key=uddi:ws:xwebservices:com:XWebEmailValidation:V2:XWebEmailValidation&rt=1

Each back-link has been explored for its contents. If any of the terms from 'initialContext' is found in the contents of a back-link, then the sentence which contains the term is added to the 'contextString'. With this step, contents extracted from the back-links are given below along with the terms from 'initialContext' given in bold in Table 3

Table 3 Contents of back-links

#	Contents extracted from back-links
1	the **XWebEmailValidation** Web Service provides all the necessary **Email Validation** Software API functionality for a low, flat, monthly fee through a well-established, well-adopted standard of exchanging information
2	**Validate**s a specific **Email** address
3	-
4	**EmailValidation**
5	XML/SOAP based Web Service which **validate**s **Email** addresses for client applications Date added: 2005-04-27 Avg Response Time: 202.04
6	XML/SOAP based Web Service which **validate**s **Email** addresses for client applications Date added: 2005-04-27 Avg Response Time: 202.04

IV. Link Based Similarity For Web Services

It is proposed to check whether a simple co-citation approach, depicted in Fig. 1, can be used for finding similar web services.

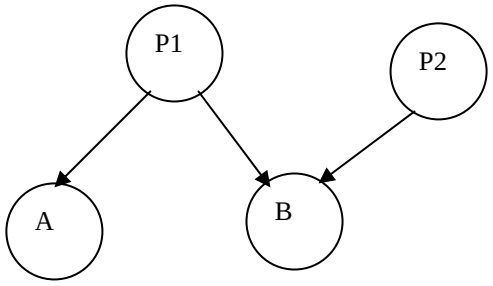

Fig 1 Co-citation approach for web service similarity

With this approach, two web services say A, B are considered to be similar if they have common parents or back-links. In order to retrieve all the related services during a service discovery irrespective of similarity score, it is proposed that even if one common back-link exists between two services, then they are treated as similar services. In Fig. 1, the Web

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

services A and B are considered as related services as they have common parent or back-link 'P1'.

To test whether co-citation based approach can be employed to find the similarity between web services, a reverse approach with the following steps has been taken up.

1. Consider a set of similar web services
2. Find the back-links of each web service
3. Compare the services and find common parents or back-links
4. Consider the services with one or more common parent as similar services.

To test the proposed steps, a typical data set with nine similar Short Message Service (SMS) services has been constructed as test data. The WSDL URLs of the test data along with their back-links are given below in the Table 4

Table 4 Service URLs and their back-links

#	Service URL	Back-links of the service URL
1	http://www.aswinanand.com/sendsms.php?wsdl	1. http://www.xmethods.net/ve2/ViewListing.po?key=427466 2. http://www.nabble.com/-jira--Created:-(JCR-2312)-Issues-with-Search-td25504980.html 3. http://social.msdn.microsoft.com/forums/en- US/vssmartdevicesvbcs/thread/881300ba-9e18-402b-816e-ab7b5537f830 4. http://forums.asp.net/p/1336060/2692273.aspx#2692273 5. http://issues.apache.org/jira/browse/JCR-2312 6. http://campus.en.kku.ac.th/forums/index.php?action=profile;u=285;sa=showPosts 7. http://social.msdn.microsoft.com/Forums/ko-KR/vssmartdevicesvbcs/thread/881300ba-9e18-402b-816e-ab7b5537f830 8. http://www.mail-archive.com/ dev@jackrabbit.apache.org/msg19141.html 9. http://www.mail-archive.com/ dev@jackrabbit.apache.org/msg19125.html
2	http://www.webservicex.com/SendSMS.asmx?WSDL	1. http://www.xmethods.org/ve2/ViewListing.po?key=uuid583A0319-E0E 2. http://sourceforge.net/mailarchive/forum.php?forum_name=pywebsvcs-checkins&max_rows=25&style=nested&viewmonth=200411 F-4F11-79E7-68BA3E099A9B 3. http://social.msdn.microsoft.com/forums/en-US/vssmartdevicesvbcs/thread/881300ba-9e18-402b-816e-ab7b5537f830 4. http://forums.asp.net/p/1336060/2692273.aspx#2692273
3	http://ws.acrosscommunications.com/SMS.asmx?WSDL	1.http://www.xmethods.com/ve2/ViewListing.po?key=uuid:E9BC8BF4-34A7-A291-1620-364A4A0205D0 2.http://sourceforge.net/mailarchive/forum.php?forum_name=pywebsvcs-checkins&max_rows=25&style=nested&viewmonth=200411
4	http://www.abctext.com/webservices/SMS.asmx?WSDL	1. http://www.xmethods.com/ve2/ViewListing.po?key=uuid:39EFD748-3EE6-7D0F-1046-1404411ED503 2. http:// sourceforge.net/mailarchive/orum.php?forum_name=pywebsvcs-checkins&max_rows=25&style=nested&viewmonth=200411 3. http://lists.w3.org/Archives/Public/www-ws-arch/2003Feb/0191.html 4. http://lists.w3.org/Archives/Public/www-ws-arch/2003Feb/0199.html
5	http://www.esendex.com/secure/messenger/soap/SendService.asmx?wsdl	1. http://www.xmethods.net/ve2/ViewListing.po?key=uuid:32E1822A-C028-9740-CBE6-6AC94139A552 2. http://forums.oracle.com/forums/thread.jspa?threadID=876082&tstart=1 3. http://spreadsheets.google.com/pub?key=pJgpFmDI-D1F-pajZsYLdDw&gid=1 4. http://www.apifinder.com/APIFinder/APIDisplay/29006 5. http://www.xmethods.net/ve2/ViewListing.po?key=427062 6. http://www.webservicelist.com/api/web_services_communication_esendex_sms_serviceapi_websitedesign239658.html
6	http://ws.cdyne.com/SmsWS/SMS.asmx?wsdl	1. http://www.xmethods.net/ve2/ViewListing.po?key=420986 2. http://fmnexus.com/groups/ws/blog/ 3. http://fmnexus.com/groups/ws/webLog
7	http://www.sms.mio.it/webservices/sendmessages.asmx?WSDL	1. http://www.xmethods.net/ve2/ViewListing.po?key=uuid:87D23AE6-4B48-66B1-228F-6CD8BB7EA144 2. http://www.xmethods.net/ve2/ViewListing.po?key=uuid:A472C5F5-F6CA-E425-5F2A-F09C5F8A4998 3. http://wiki.eclipse.org/WSDL_Test_Plan 4. http://wiki.eclipse.org/index.php/WSI_Test_Plan
8	http://www.info-me-sms.it/ws.php?wsdl	1. http://www.xmethods.com/ve2/ViewListing.po?key=uuid:A97C9F9C-C0D0-F746-A23D-3827752D556D
9	http://ws.strikeiron.com/globalsmspro2_5?WSDL	1. http://pbdj.sys-con.com/node/358725 2. http://www.xmethods.net/ve2/ViewListing.po?key=uuid:296A5C9A-14EF-28C2-4638-E753584EC7E2 3. http://www.sys-con.com/node/358725 4. http://forums.java.net/jive/thread.jspa?messageID=168945&tstart=0 5. http://www.esigma.com/telos/disco

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

		ver/servicedetail.html?reg=uddi:webserv ices.esigma.com:uddi&key=uddi:ws:strikeiron:com:globalsmspro2_56. http://www.esigma.com/telos/discover/servicedetail.html?reg=uddi:webservices.esigma.com:uddi&key=uddi:ws:strikeiron:com:globalsmspro2_5&rt=1

V. Performance Evaluation

When the proposed approach is employed to the above data, it is found that the first two services have two common parents namely, "http://social.msdn.microsoft.com/forums/en-US/vssmartdevicesvbcs/thread/881300ba-9e18-402b-816e-ab7b5537f830" and "http://forums.asp.net/p/1336060/2692273.aspx#2692273". So, the first two services are considered as similar. With the same analogy, services 2, 3 and 4 are found as similar services as they have the common parent, "http:// sourceforge.net/mailarchive/orum.php?forum_name=pywebsvcs-checkins&max_rows=25&style=nested&viewmonth=200411".

As a reverse or backward approach is followed to test the applicability of link based similarity, the generic evaluation measure 'precision' cannot used for evaluation as the experiment is started with initial set of similar services. The next essential measure to be analyzed is 'recall'. The performance of the proposed method is evaluated using the measures, 'recall', 'false positives' and 'false negatives'

Total number similar services chosen = 9

Number of sets identified as similar by proposed method = 2

Recall = Number of relevant services retrieved/Total relevant services

Recall = 2/9 = 0.222*100 = 22.2%

When an analysis is done for false positives cases in the above result, it is found that,

Number of false positives =1

According to the proposed approach as well as by inference from Table 4, one can find that the services 2, 3, and 4 are similar because they have one common parent, namely "http:// sourceforge.net/mailarchive/orum.php?forum_name=pywebsvcs-checkins&max_rows=25&style=nested&viewmonth=200411".

When the above link is analyzed by its contents and found that it is essentially a hub page and pointing to various web services. As hub pages do not distribute relevant feature or similar feature, such links should not be considered while finding similar web services.

When the results are analyzed for false negative cases, it is found that,

Number of false negatives found = 0

To compare the results with a standard procedure, a simple key word based matching chosen. It is assumed that a search is requested with the most frequently used key word for a SMS related service; the key word might be {'send sms'}. A search method is assumed that a service will be retrieved when all the words in the key word should appear at least once in the description of service. Each WSDL file is fully scanned for its contents and with the above method the number of relevant services returned by text matcher with full key word matching,

Recall= Number of relevant services retrieved /Total relevant services = 8/9

Recall = 88.8%

False positive cases = 0

False negatives = 0

The results obtained with link based similarity are compared key word based similarity and the performance comparison is given in Table 5

Table 5 Performance comparison

Evaluation measure	Link based similarity	Key word based similarity
Recall	22.2%	88.8%
False positives	1	0
False negatives	0	0
Correctness	50% (for the test data)	100% (for the test data)

From the Table 5 it is found that the recall of link based similarity is very low when compared with key word based search. Further even in the recall of 22.2%, the correctness of link based similarity measure is given as

Correctness = false positives/relevant retrieved

Correctness = 1/2=0.5 = 50%

Whereas the Correctness of keyword based method for the test data = 8/8 = 1= 100%.

VI. Results and Conclusion

Firstly, a back-link based context extraction technique has been suggested to extract context information of a web service and initial results with typical test data indicate that proposed approach works fine. In contrast to the work in [18] where multiple contexts have been arrived from Web, in the present work, the context search is narrowed down only to the relevant back-links. This will improve the performance of context extraction. Further, in the present work, the initial context/seed page arrived does not depend on the way in which services are described. Irrespective of the service description, the URL of the service is always known and the link based context extraction can behave independent of the type of service description

Secondly a link based co-citation algorithm has been employed in web service to find its usefulness in discovering similar services. A backward or reverse method has been proposed with a typical similar SMS services. The recall obtained with the proposed method is only 22.2% whereas the expected recall is 100% as the test is carried out with set similar services. Initially one false positive case has been found in the results of proposed method. This is due to the existence of hub pages occurring in the back-links. Hub pages will not propagate feature to their out-links. Still an analysis with back links shows that the proposed method is useful in identifying better referrers with high hub score and might be used for either filtering of discovered services or ranking the discovered services based on hub score.

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

REFERENCES

[1] Colin Atkinson, Philipp Bostan, Oliver Hummel and Dietmar Stoll, "A Practical Approach to Web Service Discovery and Retrieval" 2007 IEEE International Conference on Web Services (ICWS 2007), Salt Lake City, UT, pp. 241-248

[2] Kyong-Ha Lee, Mi-young Lee, Yun-Young Hwang, Kyu-Chul Lee, "A Framework for XML Web Services Retrieval with Ranking" 2007 IEEE International Conference on Multimedia and Ubiquitous Engineering (MUE'07), pp.773-778

[3] Vladislava Grigorova, "Semantic Description of Web Services and Possibilities of BPEL4WS", International Journal Information Theories & Applications, 2006, Vol.13, No.2, pp. 183-187

[4] Qingcong Lv, Jiliang Zhou, Qiying Cao, "Service Matching Mechanisms in Pervasive Computing Environments", International Workshop on Intelligent Systems and Applications, Wuhan, May 2009, pp. 1-4

[5] Umesh Bellur, Harin Vadodaria and Amit Gupta, "Semantic Matchmaking Algorithms", Advances in Greedy Algorithms, Source: Advances in Greedy Algorithms, Book edited by: Witold Bednorz, ISBN 978-953-7619-27-5, I-Tech, Vienna, Austria, November 2008, pp. 586

[6] Ruiqiang Guo, Jiajin Le, XiaoLing Xia, "Capability Matching of Web Services Based on OWL-S", Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA' 05), IEEE Computer Society, August 2005, pp. 653-657.

[7] Rama Akkiraju1, Biplav Srivastava, Anca-Andreea Ivan1, Richard Goodwin, Tanveer Syeda-Mahmood, "SEMAPLAN: Combining Planning with Semantic Matching to Achieve Web Service Composition", IEEE International Conference on Web Services (ICWS'06), Chicago, September 2006, pp. 37-44

[8] Tanveer Syeda-Mahmood, Gauri Shah Rama Akkiraju, Anca-Andrea Ivan,Richard Goodwin, "Searching Service Repositories by Combining Semantic and Ontological Matching", Proceedings of the IEEE International Conference on Web Services (ICWS'05), Orlando, Florida, July 2005, pp. 13-20

[9] Nikola Milanovic and Miroslaw Malek, "Current Solutions for Web Service Composition", IEEE Internet Computing, IEEE Computer Society, Nov-Dec 2004, pp. 51-59

[10] Sonia Ben Mokhtar, Anupam Kaul, Nikolaos Georgantas and Valerie Issarny, "Towards Efficient Matching of Semantic Web Service Capabilities", International Workshop on Web Services Modeling and Testing (WS-MaTe 2006)

[11] Pierluigi Plebani and Barbara Pernici, "URBE: Web service Retrieval based on Similarity Evaluation", IEEE Transactions on Knowledge and Data Engineering, 2009, Vol.21, Issue: 11, pp. 1629-1642

[12] Evren Sirin, Bijan Parsia, and James Hendler, "Filtering and Selecting Semantic Web Services with Interactive Composition Techniques", IEEE Intelligent Systems, 2004, Vol. 19, No. 4, pp. 42-49

[13] Lianghao Zeng, Boualem Benatallah, Marlon Dumas, Jayant Kalagnanam, Quan Z Sheng, "Quality Driven Web Services Composition", Proceeding of the 12th International conference on World Wide Web, (WWW2003), Budapest, Hungary, ACM, 2003, pp. 411-421

[14] Amit C. Awekar, Pabitra Mitra, Jaewoo Kang, "Selective Hypertext Induced Topic Search", Proceedings of the 15th International conference on World Wide Web(WWW2006), Edinburgh, Scotland, ACM, May 2006, pp. 1023-24

[15] Zhenjiang Lin, Irwin King, and Michael R. Lyu, "PageSim: A Novel Link-based Similarity Measure for the World Wide Web", Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings)(WI'06), Hong Kong, China, December 2006, pp. 687-693

[16] Andras A. Benczur Karoly Csalogany Tamas Sarlas, "Link Based Similarity Search to Fight Web Spam", AIRWEB'06, August 10, 2006, Seattle, Washington, USA.

[17] Dustdar, S and Schreiner, W., "A Survey on Web Services Composition", International Journal of Web and Grid Services (IJWGS) , Inderscience Publishers, Vol. 1, No. 1, 2005, pp.1-30

[18] Aviv Segev, "Circular context-based semantic matching to identify web service composition", Proceedings of the 2008 International workshop on Context enabled source and service selection, integration and adaptation ACM International Conference Proceeding Series, 2008, Vol.292, Article No.:7

[19] Camelia Constantin, Bernd Amann, David Cross Amblard, "A Link-based Ranking Model for Services", Springer-Verlag, Berlin Hiedelberg, R. Meersman, Z. Tari et al. (Eds), OTM 2006, LNCS 4275, 2006, pp. 327-344

[20] Qi Chen, Song-Nian Yu, Sisi Cheng, "Link Variable TrustRank for Fighting Web Spam", Proceedings of the 2008 International Conference on Computer Science and Software Engineering", IEEE Computer Society, 2008, Vol. 4, pp. 1004-1007

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)