Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 46 (2015) 876 – 883

Procedia Computer Science

International Conference on Information and Communication Technologies (ICICT 2014)

A Normalized Approach for Service Discovery

G. Sambasivam*[a], V. Ravisankar[b], T. Vengattaraman[a], R. Baskaran[c] and P. Dhavachelvan[a]

[a]*Department of Computer Science,Pondicherry University, Pondicherry, India,*
[b]*Research Scholar, Department of Computer Science, Bharathiar University, Tamilnadu, India*
[c]*Department of Computer Science and Engineering, Anna University, Tamilnadu, India*

Abstract

In today's world web services are the known perception to all the users who uses the internet. The Web Service process involves service discovery, selection and ranking. Discovery is the process of matchmaking of user query with advertisements in the repository. Our motivation is to develop a model for web service discovery with the combined approach of service selection and ranking. In this paper, we have proposed a technique for web service discovery process, combining the keyword search and semantic search and ranking the services. The implementation results show that the proposed model performs better for the web service discovery process.

Peer-review under responsibility of organizing committee of the International Conference on Information and Communication Technologies (ICICT 2014)

*Keywords:*Web Service Discovery; QoS; Keyword; Semantic;Ranking.

1. Introduction

Many Researchers and academician focuses on the web service technology, because Nowadays more and more and web services are given by the service providers. So choosing a web services based on functional requirements alone does not satisfy user requirements. The Quality of Service (QoS) has been the non-functional parameter has to be considered for the selecting the best service. So based upon QoS user can get accurate and best services for their businesses. For our experimentation purposes we have used the real world data sets. Many websites like XMethods.net etc. providing a repository for web services. In most of our study we have analyzed that the discovery process is based on the user's requirements.

* Corresponding author. Tel.: +91-9944756318.
E-mail address: gsambu@gmail.com

Peer-review under responsibility of organizing committee of the International Conference on Information and Communication Technologies (ICICT 2014)
doi:10.1016/j.procs.2015.02.157

Several algorithms have been proposed by many researchers for service discovery, selection and ranking. The traditional discovery process uses only keyword search, the search is not capable of finding all the related services for users[17, 19]. In the keyword search process there is possibility of omitting the services according the user requirements. So we propose a model with a combination of keyword search and semantic search. And we rank the services based on normalization of the QoS parameter Values. The Fig.1. shows the standard web service model, It mainly constitute of three components: Service Provider, Service Consumer and UDDI (Universal Description, Discovery and Integration).

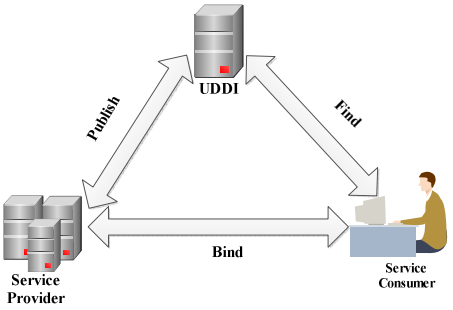

Fig.1. Web Service Model

Service provider who registers the service in UDDI with all the details like location, provider name etc…the details will be stored in the UDDI repository[14,21]. The consumer who requires services will query the UDDI repository according to his needs. Then the UDDI accept the user query and searches the appropriate services from the registry. Then the matched results will be provided to the service consumer. The rest of the paper is structured as follows. In Section II, discusses about the related works helps for this study. Section III, we discussed about the proposed System and explained. In section IV, the proposed algorithm is given and explained. In section V we have given the experimentation and results. Finally, Section VI conclusion of the work is given.

2. Related Works

They mainly focus on the non-functional parameters and rank the service according to the trust and Quality of service of the service providers[12, 16]. Another thing they are claiming that decentralized approach is not that much efficient[1].They proposed the web service discovery model with QoS based on the user preference and they introduce a three layer matching mechanism[2].

The propose a architecture for web service discovery based on agent and QoS matching, selection and ranking. The model also gets user required Quality of service parameters for accuracy and efficiency[20, 13]. They have mainly focus on the dynamic web service discovery with QoS based on SOA model, the technique followed by them is finite state automata to avoid incorrect transitions [4].

They proposed the semantic model and develop a P2P[11, 15] approach for distributed web service discovery approach, because the old traditional UDDI is inefficient and wsdl does not include semantic specification[5].

They have proposed a multi agent based semantic web service discovery model[3, 4] the broker specification are not changed during the service discovery. The model analyses the user request and match the query in the registry[6, 7].They discuss about the overview and limitations of context aware methods in service discovery and they pointed out the context information also. They said that the Context definition could not be still identified[20].

They modelled specialized search engine for service discovery which the engine extract the information from the Web Service Description languages(WSDL) from tags and annotations[8].They analyze the drawback of the broker based UDDI and they suggest some of the plan for to move forward the existing standards[9,18].They propose multi-criteria approach in service discovery with based on user preferences and they include semantic and less number of QoS parameters is used[10].Based on the discussion on the related works, it motivated us to do the research work.

3. Proposed System

The service discovery is a mining process due to the vast service repository maintenance. There is a need for betterment in the service discovery process. It's identified from the literature study and a ranking based approach is proposed as shown in Fig.2.

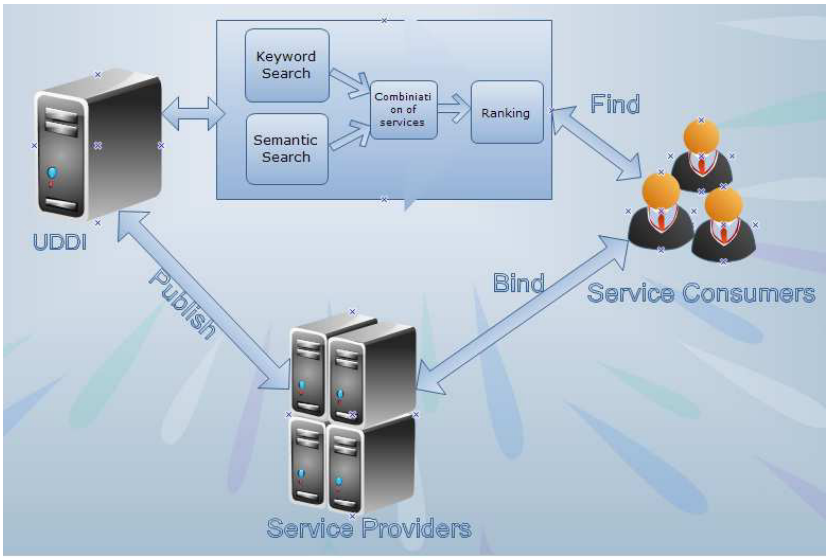

Fig.2. Web Service Discovery and Ranking Model

The proposed system consists of service provider, service consumer and UDDI (Universal Description, Discovery and Integration). The WSDSR module consists of keyword and semantic search and a ranking. First the keyword based search will happen and then semantic based search after that the both results are union together, a collection of services will be obtained. After that for ranking the QoS parameters values will be normalized to a same scale. After the normalization process the services will be ranked, the resultant services will be provided to the end user.

Calculating normalization of QoS is given below

$$\text{Normalization for Positive QoS (i)} = \frac{QoS_i - QoSi_{min}}{QoSi_{max} - QoSi_{min}} \quad (1)$$

$$\text{Normalization for Negative QoS (i)} = \frac{QoS_{max} - QoS_i}{QoSi_{max} - QoSi_{min}} \quad (2)$$

4. WSDSR Algorithm

Step 1: Input: Name of the web service (S)
User Requirements (UR)
Output: Retrieved Collection of Service with Quality of Services (QoS) with ranking
(WS_QoS)
Step 2: Select Web Service (WS) from the repository (R) with keyword Search || Semantic
Search
Keyword_ Match= (Select Services w.r.t. to Keyword match)
For (i=0; i< Keyword_ Match.length;i++)
Keyword Search []={K_WS$_i$, K_WS$_{....}$K_WSn }w.r.t keyword search
End

```
//Retrieving the matched services using Semantic Search using WordNet
//Semantic Match SM:=(Select S w.r.t to semantic meaning using wordnet)
For (i=0; i< Semantic Match;i++)
Keyword Search []={K_WSi, K_WS.....K_WSn }w.r.t keyword search
End
Keyword_Search []={K_WSi, K_WS.....K_WSn }w.r.t keyword search
Semantic_Search[] ={ S_WSi, S_WS.....S_WSn }w.r.t semantic search
Step 3: Combine both the search results
Final_Collections [WSi to WSn][WQoSi to WQoSn]= { Keyword_Search }
U { Semantic_Search }
End
Step 4: Normalizing the QosParmeters Values from the Final_Collections
QoS Values Ranges between 0 to 1
Services [services][QoS]=Normalized values w.r.t to services
Step 5: Summing up the QoS Normalized Values for each services
For (i=0;i<Services.length;i++)
QoS[i] =.Services i[Qos1 + Qos2 +…. Qosn]
End
Step 6: Ranking the Services
For (i=0;i<QoS.length;i++)
Sort the values in ascending order from
WS_QoS= QoS [i] toQoS[n]
Display the sorted list.
End Return (WS_QoS)
```

4.1. Explanation of the Algorithm

The Web Service Discovery, Selection and Ranking algorithm (WSDSR) first gets input from the user and the based upon the input it retrieves the services from the repository. After that a set of services retrieved from the repository. The algorithm first matches based upon the keyword preferred by the user. In the second step the algorithm performs a semantic matching with the used of Word Net. Then both keyword and semantic results are united together eliminating the same services.

Then the final retrieved services will be obtained. Normalization is the process of scaling the values from 0 to 1.The retrieved Quality of service parameters values are initially are not normalized, we have to normalize value to perform calculations with respect the Quality of Services. Then results then will be undergone to the process of normalization. The equation (1) and (2) is used for calculation of normalization. The QoS attribute are normalized according to the positive and negative attributes. The QoS parameters such as Availability, Throughtput are considered as positive parameters and attributes such a Cost and Response time are considered as negative attributes. The consumer requires more in positive attributes values and fewer values in negative attributes. After normalization process the QoS values will be normalized in same scale. The we have to add the all the QoS values for each services, according the values tit is sorted in ascending order and ranking of the services will be obtained. The list of ranked service will be provided to the user.

5. Experimental Results

We have implemented our proposed work in Net Beans IDE 7.3 with java swing as front end and Oracle 10g as back end. Our system has 1000 web services and using that we have implemented our proposed approach and have obtained the results as shown below in the table 1 and table 2.

Table 1. Result for Semantic Search

S.No	Service ID	Service Name	Description of the web service	Location (url)	Availability *percentage(%)*	Response Time (ms)	Throughput (*No of Times/second*)	Price ($)
1	1061	Sms	Used for Sending Short message service	http://www.2sms.com/soap/2smsmessaging.wsdl	89	303	12	700
2	1075	AWSECommerceService	Messaging	http://www.smartsms.se/IPX/services/FileShop/fshandler.asmx?wsdl	80	482	16	800
3	587	ExchangeRateService	Messaging	http://www.smsdome.com/portalvbvs/services/smsgateway.asmx?wsdl	84	126	10	250
4	875	ExchangeRateService	Message	http://www.jaredmonaco.com/ATTSMS.asmx?WSDL	91	302	15	230
5	789	FpML Validation	Messaging	http://www.directsdi.com.au/info/api/SmsGateway-HTTP.wsdl	42	407	20	350
6	368	IXML WS service	Messaging	http://www.info-me-sms.it/ws.php?wsdl	83	136	31	510
7	748	Location	Messaging	http://www.info-me-sms.it/sms.wsdl	98	210	24	120
8	986	MatchService	Messaging	http://ws.acrosscommunications.com/SMS.asmx?WSDL	90	335	12	650
9	896	prophecyService	Message	http://sms.cellcom.co.il/SmsGate/SmsGate2.asmx?WSDL	89	196	10	400
10	796	wordcountService	Messaging	http://www.scottnichol.com/samples/hellowsdl2.php?wsdl	84	496	18	700

Table 2. QoS Based Evaluation Using Proposed System

Availability of service			Response Time			Throughput			Price		
Semantic	Keyword	SKR	Semantic	Keyword	SKR	Semantic	Keyword	SKR	Semantic	Keyword	SKR
64	41	89	286	269	303	9	19	12	772	972	700
55	36	80	462	442	482	14	24	16	892	1092	800
59	44	84	104	81	126	12	22	10	265	565	250
66	29	91	283	269	302	8	18	15	232	432	230
17	24	42	384	365	407	17	27	20	367	767	350
58	40	83	112	97	136	19	18	31	552	852	510
73	61	98	187	177	210	12	21	24	162	262	120
65	38	90	314	294	335	10	15	12	647	947	650
64	31	89	179	153	196	9	14	10	496	896	400
59	44	84	471	454	496	26	11	18	747	1047	700

Table 3. Ranking of services

S.No	Service ID	Service Name	Description of the web service	Location (url)	Availability ***percentage(%)***	Response Time (ms)	Throughput (*No of Times/second)*	Price ($)	Ranking
1	1061	Sms	Used for Sending Short message service	http://www.2sms.com/soap/2smsmessaging.wsdl	89	303	12	700	1
2	1075	AWSECommerceService	Messaging	http://www.smartsms.se/IPX/services/FileShop/fshandler.asmx?wsdl	80	482	16	800	2
3	587	ExchangeRateService	Messaging	http://www.smsdome.com/portalvbvs/services/smsgateway.asmx?wsdl	84	126	10	250	3
4	875	ExchangeRateService	Message	http://www.jaredmonaco.com/ATTSMS.asmx?WSDL	91	302	15	230	4
5	789	FpML Validation	Messaging	http://www.directsms.com.au/info/api/SmsGateway-HTTP.wsdl	42	407	20	350	5
6	368	IXML WS service	Messaging	http://www.info-me-sms.it/ws.php?wsdl	83	136	31	510	6
7	748	Location	Messaging	http://www.info-me-sms.it/sms.wsdl	98	210	24	120	7
8	986	MatchService	Messaging	http://ws.acrosscommunications.com/SMS.asmx?WSDL	90	335	12	650	8
9	896	prophecyService	Message	http://sms.cellcom.co.il/SmsGate/SmsGate2.asmx?WSDL	89	196	10	400	9
10	796	wordcountService	Messaging	http://www.scottnichol.com/samples/hellowsdl2.php?wsdl	84	496	18	700	10
11	1061	Sms	Used for Sending Short message service	http://www.2sms.com/soap/2smsmessaging.wsdl	89	303	12	700	11
12	1075	SMS	Messaging	http://www.smartsms.se/IPX/services/FileShop/fshandler.asmx?wsdl	80	482	16	800	12
13	587	SMS Search	Messaging	http://www.smsdome.com/portalvbvs/services/smsgateway.asmx?wsdl	84	126	10	250	13
14	875	SMS	Message	http://www.jaredmonaco.com/ATTSMS.asmx?WSDL	91	302	15	230	14
15	789	SMS	Messaging	http://www.directsms.com.au/info/api/SmsGateway-HTTP.wsdl	42	407	20	350	15

Table 3 show the results of combined approach of keyword search and semantic search and the table 2 shows the results of web services after the process of normalization and ranking.

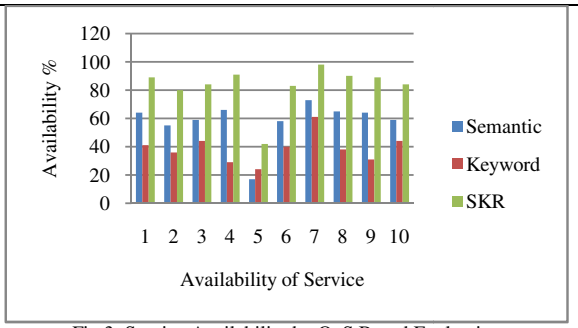

Fig.3. Service Availability by QoS Based Evaluation

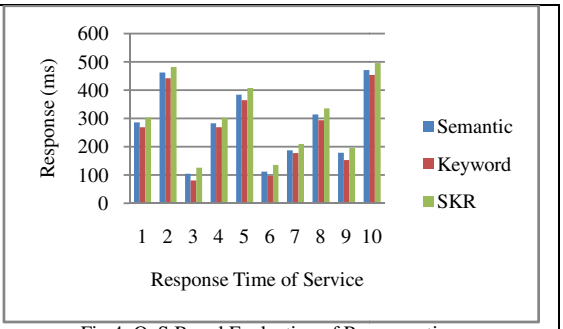

Fig.4. QoS Based Evaluation of Response time

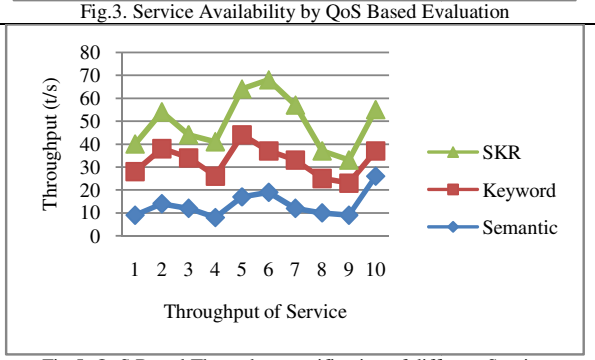

Fig.5. QoS Based Throughput verification of different Service

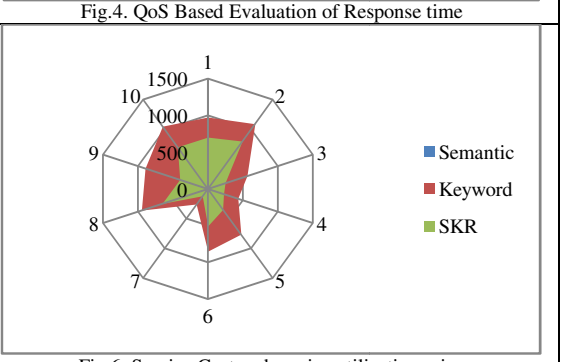

Fig.6. Service Cost and service utilization price

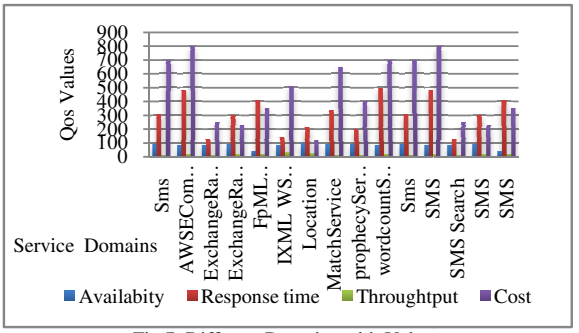

Fig.7. Different Domains with Values

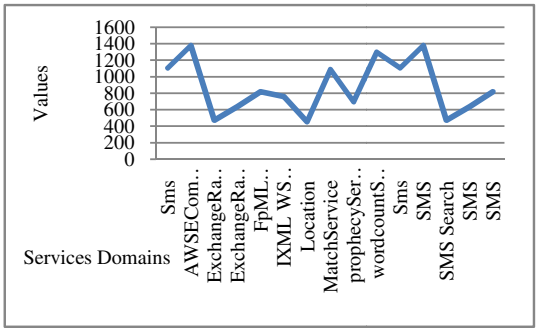

Fig.8. Ranking of the Services

5.1. Result Analysis

The figure result evaluations show clearly that the QoS based service discovery is improved and the service reliability also maintained in the proposed system. Its variations are tabulated in Table 3 the corresponding results are tabulated in Fig.3. identifies the availability of the service and its variability described under different techniques. Fig.4. describes the response time and its corresponding service mitigations are illustrated. Moreover, in Fig.5 the throughput of different service providers and the failure rate also drastically decreases in the proposed system. Similarly in Fig.6 its corresponding cost of search and maintenance also includes with the improved approach in the present paper. However, in Fig.7. and Fig.8. describes the service availability, response, throughput and cost of service under different domains and its ranking respectively.

6. Conclusion

In this paper the problems in web service discovery and ranking are analysed and we have proposed a new WSDSR algorithm and implemented. Our proposed approach with keyword with semantic search is efficient. The implementation results show that this approach is better implementation of the web service discovery and ranking. Results shows that the proposed approach yields better results. In future, we will enhance the algorithm with more number of QoS parameters and makes them efficient.

Acknowledgment

This work is a part of the Research Project sponsored under the Major Project Scheme, UGC, India, Reference No: F.No.41-616/2012 (SR)/Dt.18-07-2012. The authors would like to express their thanks for the financial supports offered by the Sponsored Agency.

References

1. FatihEmekci, Ozgur D. Sahin, DivyakantAgrawal ,Amr El Abbadi.A Peer-to-Peer Framework for Web Service Discovery with Ranking. *Proceedings of the IEEE International Conference on Web Services (ICWS'04.)* DOI: 10.1109/ICWS.2004.1314739, 2004; p.192-199.
2. KaiZheng, Hailing Xiong.Semantic Web Service Discovery method Based on User Preference and QoS. 2nd International Conference on Consumer *Electronics, Communications and Networks (CECNet)*, 2012; p.3502-3506.
3. Dhavachelvan.P, Uma.G.V. Multi-agent-based integrated framework for intra-class testing of object-oriented software. *Applied Soft Computing Journal*, 2004; **5 (2),** p.205-222. DOI: 10.1016/j.asoc.2004.04.004.
4. T.Rajendran, P.Balasubramanie. An Optimal Agent-Based Architecture for Dynamic Web Service Discovery with QoS.*Second International conference on Computing, Communication and Networking Technologies*, p.1-7. DOI: 10.1109/ICCCNT.2010.5591673.
5. Murugaiyan.S.R, Sambasivam.G, Vengattaraman.T, Dhavachelvan.P, Venkatachalapathy.V.S.K. An improved matchmaking technique for Web service discovery. *International Journal of Applied Engineering Research*, 2014; 9(13), p.2359-2368.
6. Chiranjeevi Manike, P.P.S Naik, Kavadi Durga Prasad. An Efficient Framework for High-Quality Web Service Discovery.*CSI Sixth International Conference on Software Engineering (CONSEG)* 2012; p.1-7. DOI: 10.1109/CONSEG.2012.6349480.
7. Mohan.R, Murugaiyan.S.R, Sambasivam.G, Vengattaraman.T, Dhavachelvan.P. User preferential model for semantic web service discovery based on concept lattices. *International Journal of Engineering and Technology*, 2014; 5 (4), p.3470-3483.
8. ZhenqiWang,Yuanyuan. An Approach for Semantic Web Service Discovery Based on P2P Network. *4th International Conference on Wireless Communications, Networking and Mobile Computing*- WiCOM '08. p. 1-4, DOI: 10.1109/WiCom.2008.1230.
9. Balaji.N, Sambasivam.G, Basha.M.S.S, Vengattaraman.T, Dhavachelvan.P. SLA based architecture for web service selection and ranking with QoS. *International Conference on Human Computer Interactions*, ICHCI 2013, DOI: 10.1109/ICHCI-IEEE.2013.6887783.
10. AzadehGhariNeiat, MehranMohsenzadeh, RanaForsati, Amir Masoud Rahmani. An Agent- based Semantic Web Service Discovery Framework. *International Conference on Computer Modeling and Simulation*, p.194-198, DOI: 978-0-7695-3562-3/09.
11. P. Victer Paul, D. Rajaguru, N. Saravanan, R. Baskaran and P. Dhavachelvan. Efficient service cache management in mobile P2P networks. *Future Generation Computer Systems*, Elsevier, 2013; **29 (6)**, p.1505–1521.
12. Ourania Hatzi, Georgios Batistatos, Mara Nikolaidou, Dimosthenis Anagnostopoulos. A Specialized Search Engine for Web Service Discovery. *IEEE 19th International Conference on Web Services (ICWS)*. DOI: 10.1109/ICWS.2007.12.
13. Colin Atkinson, Philipp Bostan, Oliver Hummel and Dietmar Stoll. A Practical Approach to Web Service Discovery and Retrieval. *IEEE International Conference on Web Services* (ICWS 2007), p.24-248.
14. P. Victer Paul, N. Saravanan, S.K.V. Jayakumar, P. Dhavachelvan and R. Baskaran. QoS enhancements for global replication management in peer to peer networks. *Future Generation Computer Systems*, Elsevier, 2012; 28 (3), p. 573–582.
15. Kritikos, K.; Plexousakis, D. Evaluation of QoS-Based Web Service Matchmaking Algorithms. *IEEE Congress on Services*; 6(11), p.567,574. doi: 10.1109/SERVICES-1.2008.
16. GaoShu, Omer F. Rana, Nick J. Avis, Chen Dingfang. Ontology-based semantic matchmaking approach. *Advances in Engineering Software*.2007; **38**, p. 59-67.
17. N. Balaji, S.R. Murugaiyan, M.S. SaleemBasha, R. Baskaran and P. Dhavachelvan. Enhancements for UDDI using User Preferential Web Service Selection Model based on SLA. *International Journal of Engineering and Technology*, 2013; 5(5), p.4057 – 4067, ISSN: 0975-4024.
18. Agarwal, S Studer, R. Automatic Matchmaking of Web Services, *International Conference on Web Services-ICWS '06*, vol. no. pp.45-54, doi: 10.1109/ICWS.2006.35.
19. D.Chandramohan, T. Vengattaraman and P.Dhavachelvan. Data Privacy Breach Prevention Framework for the Cloud Service. *John. Wiley: Security Comm. Networks* (2014), DOI:10.1002/sec.1054.
20. Hong Qing Yu; Dietze, S.; Benn, N., Autonomous Matchmaking Web Services. *International Conference on Computer Information Systems and Industrial Management Applications (CISIM),* 2010; 8(10), p.420-425. doi: 10.1109/CISIM.2010.5643504.(4-7).
21. Murugaiyan.S.R, D.Chandramohan, T. Vengattaraman and P.Dhavachelvan. A Generic Privacy Breach Preventing Methodology for Cloud Service. *International Journal of Grid and High Performance Computing.* 2014;6(3), p. 56-88. DOI: 10.4018/ijghpc.2014070104.