Efficient Discovery of Semantic Web Services

Shalini.L, Femila Goldy.R

PG Student, Dept. of Computer Science and Engineering, Anand Institute of Higher Technology,

Kazhipattur, Chennai, Tamil Nadu, India.

Assistant Professor, Dept.of Computer Science and Engineering, Anand Institute of Higher Technology,

Kazhipattur, Chennai, Tamil Nadu, India.

Abstract—In this paper, our work is mainly focused on to discover the semantic web services efficiently and also for making the use of semantic web service easier. Semantic Web Service is a revolutionary technology which is growing popularly, due to the increased level of machine understandability. One of the major open problem of semantic web service discovery is approximated matching. We propose semantic web service discovery (SWSD) framework using Sense based match making mechanism that provides flexibility for searching exact word. It incorporates NLP techniques for disambiguating words, meanings and establishing a context for a set of words. This mechanism improves the quality of discovery process which enhances the user selection and also helps to express the desired semantic Web services. The experimental test shows the positive impact on the discovery process.

Keywords—Semantic Web Service, Ontology, Match Making, Natural Language Processing.

I.INTRODUCTION

The Semantic web depends on machine interaction to acquire accuracy over information retrieval. The Conventional web resources includes syntactic discovery which could bring out only the related results not the relevant results. Thus it remains less efficient and needs an upgrade to the Semantic web. The main challenges in the area of Semantic Web services are composition, discovery and trust. Semantic Web Services aim at addressing this challenge on the basis of comprehensive, machine interpretable semantic descriptions. The factors that affect the discovery process are the ability of service providers to describe their services, service requestors to describe their requirements and the effectiveness of the service matchmaking algorithm. The mere existence of semantic descriptions does not improve service discovery, the semantic information need to be incorporated during the matchmaking process. In order to serve this purpose, matchmaking approaches have been used.

The proposed web service discovery framework is an approach for Web service description based on Natural Language Processing Technique done by means of semantic languages matching with the user requested query. This framework can also be able to establish a broader search field by making use of related concepts from the ontologies for identifying the context in which the Web service is operating. This approach place the services in an appropriate order for composition based on the similarity measures.

II.BACKGROUND

The various techniques are involved in the semantic web services for the selection, composition and discovery of services. In this section, we describe some concepts definitions and methodologies utilized in our framework.

A.   Web Service Description Language

WSDL is a description language for Web services, i.e., a formalism, which represents a Web service on an abstract level as well as its concrete definition. WSDL is XML-based and independent of the underlying programming language of the service implementation and the platform the service is running on. It encapsulates the service functionalities so that a service consumer does not have to handle different technological infrastructures at the service provider's side. The abstract part of a WSDL document advertises what a service does while the concrete part defines how a service can be consumed and where it is located. Thus, a WSDL document defines a contract between service consumer and provider with regard to functional and selected non-functional aspects of the service. WSDL is purely syntax-based, i.e., service functionalities, message types are only defined on a syntactic level.

B. Semantics of web services

Semantics of information refers to the meaning and use of information. Semantic Web content can be divided into data and metadata. While the data represents the actual content, metadata can be defined as data or information about data. In general, the application of metadata descriptions is twofold: On the one hand, an abstraction of the representational details can be achieved while capturing the information content, and on the other hand, domain knowledge can be associated with the data allowing to make inferences such as the relevance of data or relationships to other pieces of information.

The key to achieve machine processability of content is the grounding of terms used in metadata descriptions in well-defined, standardized vocabularies. Apart from enhancing already existing Web service description components by meanings, the integration of semantics into Web service standards resulted in the definition of new elements, which have not been regarded in non-semantic Web service standards.

C. Kinds of semantics for Web services

a) *Data semantics* formally define data in input and output messages of Web services.

b) *Functional semantics* formally define the capabilities of Web services, i.e., by defining preconditions and effects and semantically annotating interfaces and operations.

c) *Non-functional semantics* reference QoS and general policy requirements/constraints.

d) *Execution semantics* describe the execution of services and operations.

D. Ontologies

In Semantic Web research, ontologies provide the foundation for machine processable data and allow exchange of information between people and machines by both syntactic and semantic means. The usage of ontologies has got a long tradition as means to describe the knowledge about a particular domain. "Ontology is a formal, explicit specification of a shared conceptualization. "Ontology is formally specified i.e., it makes use of a defined ontology language. Second, conceptualization refers to an abstraction of a domain which includes the relevant concepts in that domain. Third, Ontology is based on shared knowledge, i.e., it represents an agreed viewpoint.

E. Web Ontology Language

Web Ontology Language (OWL) provides the basic functionalities and properties of the Semantic Web. OWL-S is a heavy weight approach that defines a full featured, OWL-based ontology for the semantic description of services. The goal of OWL-S is to support automatic Web service discovery, invocation, composition and interoperation. It follows a top-down approach and describes the semantic characteristics of a service, which may then be grounded in a functional description.

The basic components involved in the web ontology languages are service profile, service model and service grounding. OWL-S may also be regarded as a complement to WSDL, not as a substitute. On enabling

M.R. Thansekhar and N. Balaji (Eds.): ICIET'14

rich semantics to the web resources through use of Web Ontology Language Services (OWL-S), the machine interoperability to understand the user's intention on information retrieval can be enhanced. The semantic concepts to be formally defined in an OWL DL based ontology. This restriction is a common assumption in conjunction with the use of matchmaking.
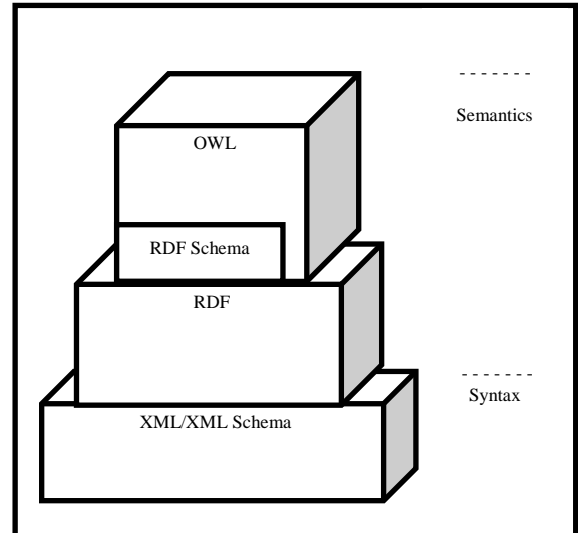

Fig: 1 Ontology language in Semantic Web

III. RELATED WORK

Patil et al [1] presented MWSAF framework for semi-automatically marking up Web Service descriptions with ontologies and they have developed algorithms for the measure of average concept and the average service match. But it shown low accuracy even though its work evaluated with a set of only 24 services divided in to two categories. It did not provide any evaluation data concerning the mapping process. G. Meditskos et al [6] describe and evaluated a Web service discovery framework using OWL-S advertisements, combined with the distinction between service and Web service of the WSMO Discovery Framework. Their main aim is to find the initial set of candidate Web services for a specific request. They also discussed about the techniques such as object-based structural matching, OWL-S profile, structural information and role-oriented matchmaking. But their framework with composition capabilities returns only single Web services and its framework is not efficient with more structural and information content similarity measures.

Klush et al [10] discussed about the performance of OWLS-MX in terms of its false positives and false negatives using the service retrieval test collection OWLS-TC 2.1. Based on this they implemented a hybrid matchmaker version OWLS-MX2 and it is compared with OWLS-X to show the improved precision in average. Hybrid semantic matching was performed by OWLS-MX2 has its own deficiencies in particular text matching failures. Sometimes matching may introduce misclassifications by its own and also it does not concern about the automated logic-based service composition planning.

G. Meditskos et al [5] proposed a new methodology for ontology concept matching and their work was mainly focused on measuring the similarity of objects based upon Object Oriented schema. They discussed about applying reasoning

techniques over semantically enhanced Web service requests and advertisements to facilitate efficient and accurate discovery of Web services. Also they discussed about the problem of semantic web services matchmaking from an Object Oriented perspective. It cannot able to apply direct object similarity measures on profile objects and it did not enrich their methodology with services in pre and post conditions. Paliwal et al [2] described about the approach for service categorization and semantic enhancement of the service request for the discovery of semantic based web services. Their categorization of services was performed at the universal description discovery and integration based on an ontology framework and they utilized the clustering technique for classifying the web services. Their proposed approach performs the two key steps related to automated service discovery, but they didn't investigate additional mapping tools to express a service request to search for relevant concepts and they assign only binary values to the ranking parameters.

Kritikos et al [9] discussed about OWL-Q, a rich, extensible and modular ontology language for QoS based WS description that complements OWL-S and semantic QoS metric matching algorithm. Based on this they have extended the discovery algorithm and the Constraint Satisfaction Problem-based approach for QoS based WS discovery. They proposed a modification to the metric matching algorithm and two novel semantic QoS based WS Discovery algorithms. But there may exist some over constrained problems like semi ring based constraint satisfaction.

Liu et al [3] described about the approach of similarity search and classification of service operations and proposed an approach of a new similarity metrics which utilizes the external knowledge to compute the semantic distance of terms from two compared services and the similarity of services are measured upon these distances. As the WSDL contain only very few text fragments the techniques which is used here is rely on term frequency and approaches utilizing the distance of terms do not perform well.

In recent work, Farrag et al. [4] introduced an algorithm for mapping WSDL descriptions to OWL-S descriptions by making use of an ontology-based approach. It introduces a new discovery mechanism that has the ability to provide optimal results for service request using mapping algorithm. This mechanism is distinguished by its service repository, which is built using the advertisements of semantic and non-SWS. It includes ontology search and standardization engine to describe any data type in the system which helps in providing standardization process. Used to assist the process of finding suitable ontology. However, this approach suffers from several limitations such as, not fully used semantic methods for discovery, only a part of semantic methods is used like ontology not lexical and sense based methods. Not contain automatic process for discovery. Mainly contain disambiguation problem.

IV.PROPOSED WORK

The semantic Web service descriptions use ontologies to describe the behavior of a Web service by applying reasoning over Web service semantics. In this way, the semantics described in ontologies enable systems to interpret what a Web service is doing stimulating automatic Web service discovery and composition.
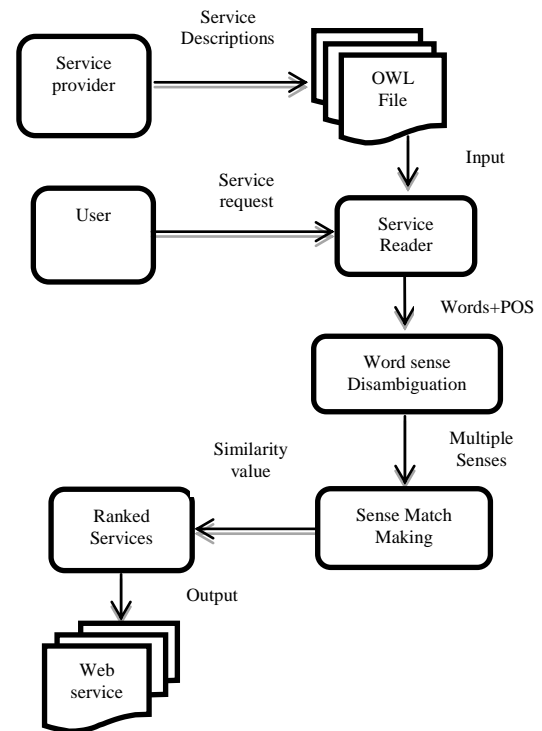

Fig: 2 Overall Process Design

The above figure represents the overall design of our proposed framework. In this process user request and service files are parsed and extract words from it. Then perform similarity metric measure based on sense based matching technique. The retrieved details are considered to be services and are stored in local server, which is taken for further process. The ontologies, however, are created by humans and therefore contain natural language. This allows humans to understand the concepts defined, but a system, in contrary to humans, can only understand ontology concepts and their relationships to a limited extend. Natural language processing (NLP) techniques can therefore help in better defining the context of a Web service. When using one holistic ontology, machines can discover and compose Web services automatically based on the semantics defined. Using one holistic ontology is, however, hardly reachable and therefore it is impossible to reason based only on formal logic. NLP techniques can help overcome the ambiguity problems between different ontologies that are being used by semantic Web service descriptions. We proposes a semantic Web service discovery framework based on Sense match making mechanism for finding semantic Web services by making use of natural language processing techniques (NLP). This method provides flexibility by not only searching for exact word matches, but also by looking for synonyms found in a

popular lexical database as WordNet. Matching the users search context with a Web service context by means of a similarity measure.

V.IMPLEMENTATION

The process is subdivided into major tasks such as Service Reading, word sense disambiguation, sense match making and service ranking.

*Service Reader***:** The first step in the process of searching for semantically described Web services is to implement readers. To enable a search engine to look through Web service descriptions written in different languages, it has to have several different Web service description readers. Service reader get Service Request from user and a set of Web services from the service provider that are described in semantic languages (OWL-S) as input. Readers parse a description and extract concepts, attributes, and relations from OWL-S. Each word in the sentences found in the non-functional descriptions, must be tagged with the right Part-of-Speech (POS) in order to be useful. Using a POS-tagger, nouns and verbs from sentences are extracted and used for the WSD.

Word Sense Disambiguation: A user can represent its goal by defining a set of words (nouns and verbs). As many words can have associated multiple meanings. Words extracted from the service reader are given as input to Word Sense Disambiguation. Disambiguation of word senses helps in finding the correct context. The disambiguation will be applied to the set of words gathered from the user input and from a semantic Web service description. The WSD task subsequently determines the senses of a set of words. Words will have different senses and each sense has its own identifier, a WordNet synset. For each synset, the similarity of the other words will be computed. Sense similarities are computed by making use of Information Content (IC), which depends on the probability of the occurrence of a particular sense in a large corpus:

$$IC\ (sense) = 1/\log P\ (sense) \qquad \ldots\ldots\ldots\ldots$$
(1)

The WSD disambiguates a word based on a previously disambiguated set of words and their related senses. Per word sense, a similarity with the senses from the context is calculated and the sense with the highest similarity is chosen. Subsequently, the word and its chosen sense are added to context and the process is iterated until there are no ambiguous words left.

Sense Matchmaking: This section starts with a high level view of the matching between the user input and the different levels of information extracted from a Web service description. After the disambiguation of all words gathered from the user input and a semantic Web service description, each word in the user query is assumed to be equally important and hence the user input contains one set of senses. However, a Web service description can contain words that better represent the context of the Web service than other words. Therefore, after the WSD, several sets of senses, each having a different weight for the matching process, are computed for a Web service. Synset resulting from the disambiguation process must

be matched in order to get a final similarity measure. Compute the lexical similarity using WordNet. By combining the synset similarity value and the lexical representation similarity using a weighted average, determine a final similarity value.

Input: User Request, Web Service files
Output: Relevant Services
Begin
Step 1: Read Services using Service Reader
Step 2: Get Multiple Sense using WordNet
Step 3: Compute Sense Similarity
Step 4: Compute Lexical Similarity
Step 5: Combine Step 3 & Step 4
Step 6: Compute Service Ranking
Step 7: Get highest ranked services
Step 8: If Step 7 is Empty then User Insert Query Related Services
Step 9: Update DB
End

Service Ranking: Get the final similarity value for each service & user request. Rank the service according to similarity value. Display the ranked service. Then the most relevant service related to user service request is found by defining the user minimum threshold. Compare each service similarity to minimum threshold. If service similarity is greater than the minimum threshold then the service is related to user request otherwise it is not related. Finally the most relevant services are displayed.

VI.CONCLUSION AND FUTURE WORK

Our proposed framework provides an approach for searching suitable Web services that are described using semantically enriched annotations. It formulates the problem of Service matching which makes use of sense based matchmaking algorithm and provides suitable services based upon user query. As a future work, the SWSD engine could be extended in such a way that it has the ability to read more annotation formats.

REFERENCES

[1] A.A. Patil, S.A. Oundhakar, A.P. Shethand, and K. Verma, "METEOR-S Web Services annotation Framework," Proc.13[th]Int'L Conf. World Wide Web (WWW), 2004.

[2] AabhasV.Paliwal, JaideepVaidya, Nabila dam, "Semantic –Based Automated Service Discovery "IEEE Transactions on Service Computing, 2012.

[3] Fangfang Liu, Yuliang Shi2 Jie Yu, Tianhong Wang, Jingzhe Wu, "Measuring Similarity of Web Services Based on WSDL", IEEE International Conference on Web Services, 2010.

[4] Farrag, T. A., Saleh, A. I., Ali, H. A.," Towards SWSs Discovery: Mapping from WSDL to OWL-S Based on Ontology Search and Standardization Engine". IEEE Transactions on Knowledge and Data Engineering, Vol. 25, no. 5, May 2013.

[5] G. Meditskos and N. Bassiliades, "Object-Oriented Similarity Measures for Semantic Web Service Matchmaking," Proc. Fifth European Conf. Web Services (ECOWS '07), 2007.

[6] G. Meditskos and N. Bassiliades, "Structural and Role-Oriented Web Service Discovery with Taxonomies in OWL-S" IEEE Trans. Knowledge and Data Eng., vol. 22, no. 2, pp. 278-290, Feb. 2010.

[7] JebersonRetna Raj, Dr.T.Sasipraba, "Web Service Discovery Based On Computation of Semantic Similarity Distance and QOS

Normalization ", Indian Journal of Computer Science and Engineering (IJCSE), 2012.

[8] Jing Dong, Yongtao Sun, Sheng Yang, "OWL-S Ontology Framework Extension for Dynamic Web Service Composition, "IEEE Transactions on Knowledge and Data Engineering, 2009.

[9] K. Kritikos and D. Plexousakis, "OWL-Q for Semantic QoS-Based Web Service Description and Discovery," Proc.Workshop Service Matchmaking and Resource (SMRR), 2007.

[10] M. Klusch, B. Fries, and K. Sycara, "OWLS-MX: A Hybrid Semantic Web Service Matchmaker for OWL-S Services" J. Web Semantics: Science, Services and Agents on the World Wide Web, vol. 7, no. 2, Apr. 2009.

[11] Puwei Wang, Zhi Jin, "Building toward capability Specifications of Web Services Based on an Environment Ontology ", IEEE Transactions on Knowledge and Data Engineering, 2008.

[12] Walenz, B., Didion, J, JWNL: Java WordNet Library. From: http://sourceforge.net/projects/jwordnet/, 2011