Sense Match Making Approach for Semantic Web Service Discovery

[1]G.Bharath, [2]P.Deivanai
[1]M.Tech Student, [2]Assistant Professor
[1,2]Department of Software Engineering, SRM University, Chennai, India
[1]bharathwinner1@gmail.com [2]devanai.divya@gmail.com

***Abstract*- As the available number of web services are increasing, the problem of discovering and selecting the most suitable service has become complex. Hence, a new discovery mechanism is needed as the existing discovery techniques fail to find the similarities between Web services capabilities. A new discovery framework is proposed in this paper, this framework includes techniques like part-of-speech tagging, lemmatization, and word sense disambiguation. After detecting the senses of relevant words gathered from Web service descriptions and the user's query, a matching process takes place and displays the relevant services. The proposed technique is implemented and its components are validated using some test samples. The results of this experiment promise the proposed framework will have a positive impact on the discovery process.**

***Key terms*: Semantic Web Service (SWS), ontology, Lemmatization, OWL-S, WSDL, WSMO, WSML.**

I. INTRODUCTION

Semantic Web Services (SWS)[1], like traditional web services, is the server end of a client-server system for machine-to-machine interaction in the World Wide Web. The main idea of the Semantic Web is to extend the current human-readable web by encoding some of the semantics of resources in a machine-processable form. Computers will be able to search, process, integrate, and present the content of these resources in a meaningful manner. As a Semantic Web component, Semantic Web Services use markups that make the data machine-readable in a detailed (as compared to human-readable HTML pages which are not usually easily understood by computer programs). As shown in Fig. 1, there are five steps to deal with any web service:

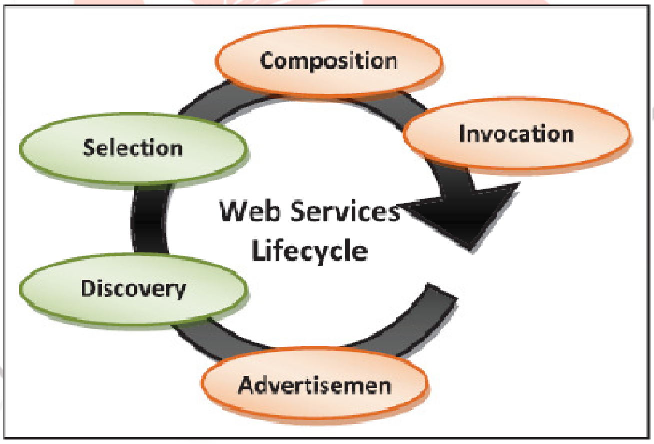

Fig.1. Web services life cycle.

The Advertisement is the process in which the service provider aims to publish information about the benefits of the service and how to use it. The Discovery is the process in which can be defined as the process of finding the list of services that can possibly satisfy the user requirements. The Selection is the process that targets to select the most suitable WS. This process is usually based on application-dependent metrics (e.g., QoS)[6].The Composition is the process that integrates the selected WSs into a compound process. The Invocation is the process that invokes a single WS or compound process by providing it with all the necessary inputs for its execution. An extended definition of the discovery process is mentioned in, which states that The discovery process is the process of finding a service that can possibly satisfy user requirements, choosing between several services, and composing Services to form a single service.

A. Web Ontology Language For Services

OWL-S [3] defines ontology to describe the properties and capabilities of web services in OWL. OWL-S is a W3C submission since 2004 and from that day on many researchers use it to describe SWSs. The OWL-S allows the process of automatic web services discovery[5], invocation, composition, and interoperation. Fig. 2 shows the basic components of the service class used to describe the web services using OWL-S[3,4]. There are three main components namely Service profile,
Service grounding, Service model.

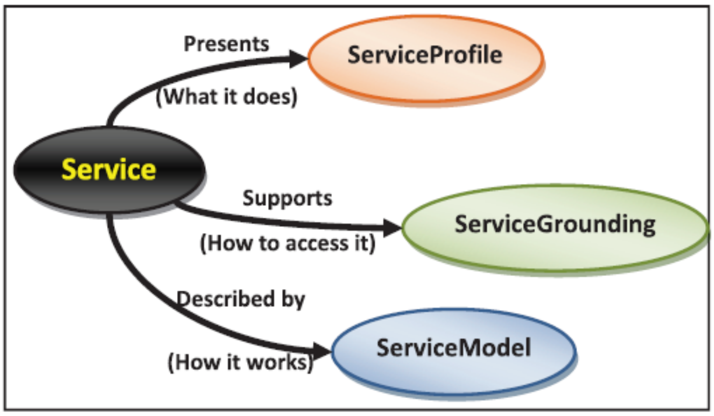

Fig. 2. Top level of the service ontology.

Service profile presents what it does and describes the function of the service and provides the all necessary information that helps in the discovery process . Service model presents how it works and describes all the processes the service is composed of, how these processes are executed, under which conditions they are executed. Service grounding describes how to access it plays the role of coordinator of the service usage.

II. RELATED WORK

A. Web Service Discovery Engines

We can distinguish between two types of approaches for Web service discovery, i.e., discovery based on clustering operation parameters on the one hand, and rich semantics on the other hand. This section continues by elaborating on existing or proposed Web service discovery systems for each of these two different approaches.

1. Web service discovery based on clustering operation parameters

One approach for Web service discovery is by searching for similarities among different WSDL servicedescriptions, enabling searching for substitutable and composable Web services as similar operations and services can be discovered based on operation parameters. Web service operation semantics can be extracted and employed for discovery purposes.

Woogle [9] is aWeb service search engine make use of clustering techniques for grouping operation parameters. For this, Woogle automatically defines the semantics of WSDL descriptions based on the operation parameters and uses these semantics to match operations. However, if independent ontologies which define the Web service semantics exist, the behavior of a Web service can be known without investigating parameter names and is therefore preferable to use.

Operation parameter clustering techniques are also used in Seekda![11], which extracts service semantics from WSDL files, enabling runtime exchange of similar and composable services. Seekda! is part of the Service-Finder [12] framework, which is a platform for service discovery where information about services is gathered from various sources like Web pages and blogs. The information is automatically added to a semantic model using automatic service annotation, realizing flexible discovery of services. Service-Finder uses service semantics for discovery and composition, but gathers this information dynamically and hence does not take into account predefined semantics.

2. Web service discovery using rich semantics

Another approach for semantic Web service discovery is the use of predefined ontologies. By identifying semantic similarities between ontologies, related semantic Web services can be discovered. To identify semantic similarities, Mediation Spaces can be used [10]. These Mediation Spaces mediate on data-level as well as semantic-level for discovery of related semantic Web services according to ontologies, other semantic Web services, or WSMO Goals.

GODO [13] does not search for similarWeb services, but instead employs a goal-driven approach. GODO has a repository with WSMO Goals and analyzes a user-described goal (in natural language). The WSMO Goal with the highest match will be sent to WSMX [14], which is an execution environment for WSMO service discovery and composition. The WSMX environment subsequently searches for a WSMO Web service that is linked to the given WSMO Goal via WSMO Mediators and returns the WSMO Web service. Although this approach makes good use of the capabilities of the WSMO framework, it cannot be applied for other semantic languages like OWL-S and WSMO-Lite, as they do not have goal representation elements.

II. PROPOSED METHODOLOGY

With the existence of Web services and Service Oriented Architecture (SOA)[6], Using SOA for Web services creates a wide network of services that collaborate in order to implement complex tasks. Currently, Web services are commonly described via narrative Web pages containing information about their operations in natural languages. These Web pages contains only plain text with no machine interpretable structure so it cannot be used by machines to automatically process the descriptive information about the Web service. To promote the automation of Web service discovery[5] and composition, a number of different semantic languages are created that allow describing functionality of services in a machine interpretable form, where current Web service descriptions contained only information about the bindings as a description and data types of a Web service functionality. Ontologies are used by the semantic Web service descriptions to describe the behavior of a Web services.

In this way, the semantics described in ontologies enable systems to explain what a Web service is doing, encouraging automatic Web service discovery and composition. The ontologies are created by humans so it contains natural language. So the concepts defined are easy to understand for humans, but a system 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. Service composition should be driven by people who know business processes and not by technicians. Thus, end users must be able to discover these Web services based on keywords written in human language. Therefore, a discovery mechanism must be developed.

A. Semantic Web Service Discovery

Sense matchmaking technique is proposed to overcome the drawbacks in current web services discovery process. Ontologies are used to describe the behavior of web services in semantic web service discovery process [1, 3]. In this way, the semantics described in ontologies enable systems for automatic Web service discovery and composition. The ontologies are created by humans and therefore contain natural language. So humans understand ontologies , but a system 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[3], machines can discover and compose Web services automatically based on the semantics defined. NLP techniques are used overcome the ambiguity problems between different ontologies that are being used by semantic Web service descriptions. In this paper a semantic Web service discovery framework for finding Web services by making use of natural language processing techniques(NLP) and Sense based match making mechanism is proposed. The flexibility of this method provides that not only searching for exact word matches, but also by searching for synonyms found in a popular lexical database as WordNet[4,8].

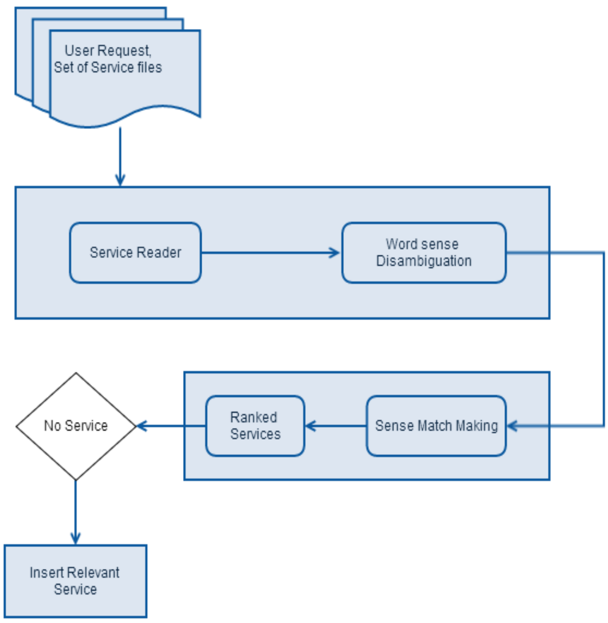

Fig 3. System architecture

1. Semantic Web Service Reader

Client gives request as input in terms of key words, Semantic Web Service Reader gets service request. It requires a set of web services that are described in semantic language that means wed ontology language for services(OWL-S)[2,4],by using this language service provider can describe the web services semantically. Semantic Web Service Reader can able to parse a description and extract concepts, attributes, and relations from ontology files (OWL-S)[3],each word in the content described in the ontology file must be tagged with the respective Parts-of-Speech by using POS tagger, it identifies the nouns and verbs from the sentences are extracted, this extracted nouns and verbs are used for web service discovery.

2. Word Sense Disambiguation

The process of disambiguation will be applied to the set of words gathered from the user input by using WordNet synset, for each synset the similarity of the words will be computed. This process helps in finding the correct context because many words have associated with multiple meanings and each word will have different senses and each sense has its own identifier.

3. Sense Match Making

Sense match making gathers the synsets results from the word sense disambiguation process and matched with other synsets were, one set of synsets are coming from the user input and several set of synsets are coming from the web services these two synsets are matched to find the final similarity value. By using the WordNet tool[4,8] lexical similarity value is calculated, here the weighted average of synset similarity value and the lexical similarity value is final similarity value.

4. Service Ranking

Final similarity value is calculated for each service and user request by sense matching module, based on the similarity value services are ranked and ranking is displayed

5. Relevant Services

After completing the ranking process based on the ranking it displays most relevant services for user request. the order of services displayed is based on the similarity value.

III. EXPERIMENTAL SET UP

This project Semantic Web Service discovery using matchmaking requires OWL files to run the project, for this project ontology files (OWL-S) are datasets. Any service provider cannot allow giving their ontology files because those files are assets for those particular organizations, some of the researchers created sample ontology files those are test collection of ontology files. These files are named as OWLS-TC_V2.2_revision_3, this is the open source dataset which defines seven concepts as subclasses of profile concept, these concepts are categories of the sets (Weapon, Economy, Travel, Food, Education, Medical and Communication) this datasets are collected from the link http://lpis.csd.auth.gr/systems/OWLS-SLR/datasets.html And some of the ontology files related to geography are collected from the online ontology library swoogle, hit is one of the open source for OWL files, www.swoogle.com is the link for swoogle. for this experiment six categories of OWL files are collected those are Geography, Restaurant, Job, Transport, Biology, Army, these files are saved in system from here admin can select a required file.
Stanford POS tagger is a Part-Of-Speech Tagger (POS Tagger) is a piece of software which is used to parse the content in the OWL files and assigns Parts-of-Speech to each word by using POS tagger, it identifies the nouns and verbs from the sentences are extracted. and WordNet[4] is a tool used for disambiguation process and to find synsets, it a lexical database for the English language which groups English words into sets of synonyms called synsets, provides short, general definitions, and records the various semantic relations between these synonym sets. As shown in Fig 4 here OWL file is taken as an input by web service reader and extract the data from the file to parse the data, to that data each word is tagged with respective parts-of-speech by using POS tagger, User request is taken as an input by word sense disambiguation and finds the synonyms for the input by using WordNet tool and the similarity of each synonym is calculated, based on the similarity values the services are displayed.

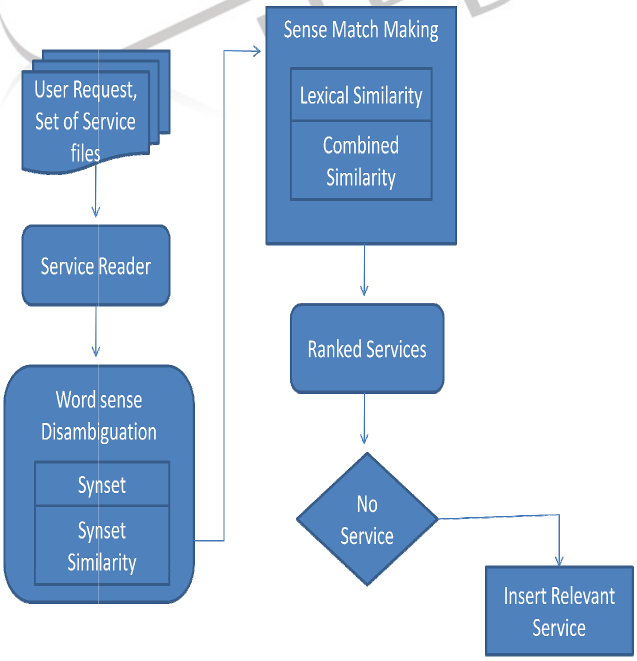

Fig 4. Experimental setup

A. Experimental Results

The aim of this project is to improve the discovery process in web services, currently discovery process in the web services displays the irrelevant services along with relevant services, this project proposed a new discovery process that displays only relevant services. 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. By using the WordNet tool, it finds the synonyms for the user request and search for the services to all the synonyms and calculate the similarity value, based on the similarity value the services are displayed.

Accuracy:

The services must be displayed accurately because this method not only searching for the exact key words given by user, but also searching for synonyms of the user request. So the displayed services contain only relevant services.

Automated:

The discovery process is fully automated, so user can use this process without having the knowledge about semantic languages (OWL-S).

IV. CONCLUSION AND FUTURE WORK

Proposed semantic wed service discovery is a keyword-based discovery process for searching web services and used fully semantic methods for discovery process, semantic methods like word sense disambiguation and WordNet is used in discovery process. This project makes an use of natural language processing techniques and a WordNet-based similarity measure for matching keywords. This approach is with two different similarity functions, one for lexical similarities and one for semantic similarities are applied for discovery of semantic Web services.

In future, the discovery process could be extended to more annotation formats, e.g., WSMO-Lite, SAWSDL. Here we have implemented semantic searching process for WSMO (OWL-S) but it can be further extended to WSML, which provides additional information about Web services with that information the content of the Web service will explained more detailed.

V. REFERENCES

[1] Tamer Ahmed Farrag, Ahmed Ibrahim Saleh, and Hesham Arafat Ali,"Toward 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

[2] Ayse B. Bener *, Volkan Ozadali, Erdem Savas Ilhan. "Semantic matchmaker with precondition and effect matching using SWRL" Elsevier (2009).

[3] Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., PatelSchneider, P. F., & Stein, L. A. OWL Web ontology language reference, www.w3.org, (2004).

[4] Budanitsky, A., & Hirst, G.Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. workshop on WordNet,(2001)

[5] Dietze, S., Gugliotta, A., & Domingue, J. Exploiting metrics for similaritybased semantic Web service discovery. IEEE International Conference on Web Services,(2009).

[6] Yang, S. J. H., Zhang, J., & Chen, I. Y. L. A JESS-enabled context elicitation system for providing context-aware Web services. Elsevier,(2008).

[7] Ernst, M.D., Lencevicius, R., Perkins, J.H.: Detection of Web Service Substitutability and Composability. In: International Workshop on Web Services — Modeling and Testing (WSMaTe 2006), pp. 123–135 (2006).

[8] Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: Wordnet: An on-line lexical database. International Journal of Lexicography 3(4), 235–244 (1990).

[9] Dong, X., Halevy, A., Madhavan, J., Nemes, E., Zhang, J.: Similarity Search forWeb Services.In: 30th International Conference on Very Large Data Bases (VLDB 2004), vol. 30, pp. 372–383 (2004).

[10] Dietze, S., Gugliotta, A., Domingue, J., 2009. Exploiting Metrics for Similarity-based Semantic Web Service Discovery. In: IEEE 7th International Conference on Web Services (ICWS 2009). IEEE Computer Society, pp. 327–334.

[11] Semantic Technology Institute, 2009. Seekda! From: http://seekda.com/.

[12] Cefriel, Seekda!, Ontoprise, University of Sheffield, 2009. Service-Finder. From: http://www.service-finder.eu/.

[13] Gomez, J. M., Rico, M., Garcia-Sanchez, F., Bejar, R. M., Bussler, C., 2004. GODO: Goal Driven Orchestration for SemanticWeb Services. In: 1st Workshop on Web Services Modeling Ontology Implementations (WIW 2004). Vol. 113. CEUR Workshop Proceedings.

[14] DERI Galway, 2008. Web Service Execution Environment. From: http://www.wsmx.org/.