IJCTA, 9(28) 2016, pp. 71-78

HYBRID APPROACH FOR AN EFFICIENT DISCOVERY OF WEB SERVICES

G.Senthil Kumar* T.S.Shiny Angel* A.Maria Nancy* and S.Veena*

Abstract: The Internet, with its ease of use and wide applications across many domains has made itself the go-to destination for many consumer services. While there is abundant web service published with at least one apt service to any client request, there is a need for an easy and efficient discovery system, which can retrieve the required service. Since most of the consumers are unaware of the syntax and structure of service description, it is necessary that the consumers have the ease of providing their requirements in natural English language. In this paper we propose a framework for semantic matchmaking among the processed user query and the published services to attain the service most similar to the user requirements. It involves assigning a semantic sense to the user query and mapping it to the service ontology.

Keywords: web services, ontology, mapping, service-discovery, semantics, matchmaking, natural language processing

1. INTRODUCTION

Web service discovery with limited published services was predominantly keyword mapping and retrieval based [1] where the descriptions of the services were looked for their functionality as a keyword and all the services wherein the keyword was found retrieved[2]. The description is defined in WSDL [3] and is directed upon by using URI's [4]. As the user base increased and correspondingly more services were published, the traditional semantic approach for discovery turned out to be ineffective. The problem arises when the user's query and the target service does not have similar keywords.

To be able to discover a service by its functionality classification and class relations, the semantic approach to discovery was introduced [5]. Web services, with their syntactic description, are now related to ontology [6] wherein they are segregated by the service type class, the object properties, and the data properties. As the user is unaware of semantic knowledge and the structural relations, we require a procedure for mapping the query in natural language [7] to the description and the category of the service.

In this paper, we suggest a hybrid framework for syntactical search with a semantic mapping add-on. In the semantic web, the service description includes the functionality definition and the structural semantic relations for every service. Here, we try to relate query and the target service description both syntactically and semantically. The parsed query is cut down to words and is assigned with ontology so that the query, ontology set can relate to the service by its name and its functionality.

In section I, we have discussed the discovery processes method followed, the drawbacks and the improvements made to them. Also, it gives a summary of what proposed framework would do. Section II shows the previous foundational work done on the semantic concepts and the discovery process efficiency. Section III depicts the process flow of the overall architecture of the proposed framework. The client and the side service modules were explained separately. Within this section, Sub-Sections A through D shows how the client-side query is processed and how the word sense ontology is assigned to it. Sub-Sections E

* Assistant Professor SRM University Chennai
Email: chenthi2004@gmail.com, angel_d_leno@yahoo.co.in, mariananancy.a@ktr.srmuniv.ac.in, veena.s@ktr.srmuniv.ac.in

through I shows how the service description is parsed, ontology assigned and mapping is made. Finally, we conclude with the brief summary and the advantages of the proposed framework.

2. RELATED WORK

The need for the use of semantics started almost a decade ago when the discovery process needed more efficiency. Okkyung Choi Et al [8] proposed the use of semantic knowledge and processing it. As the semantic relativeness increased, more and more effective and efficient proposals came for the discovery process. Zhang Jia Jie Et al [9] proposed the use of more than just the syntactic relatables while identifying the services. Also, to try and make the user query more sense full, Mikalai Yatskevich and Fausto Giunchigl Et al [10] proposed element level matchmaking by which suggests the use of WordNet lexical database to increase the scope of user query while mapping.

3. METHODOLOGY

Overall Architecture

The proposed framework initially takes the user query expressed in natural language and parsed it to obtain the relevant keywords and the multiple senses to them. The parsing procedure would include spell checking, stop word removal, Parts-of-speech assigning, tokenization, lemmatization and finally name entity recognition. The name-entities recognised would be separated from the query word set for identifying the services by their name. The remaining word set is sent to the Word-sense disambiguation module where different synonyms of the query words are derived. The word query set is then assigned with its ontology. We use WordNet to assign structural meaning to the word set which can be tried and mapped onto the service ontology.

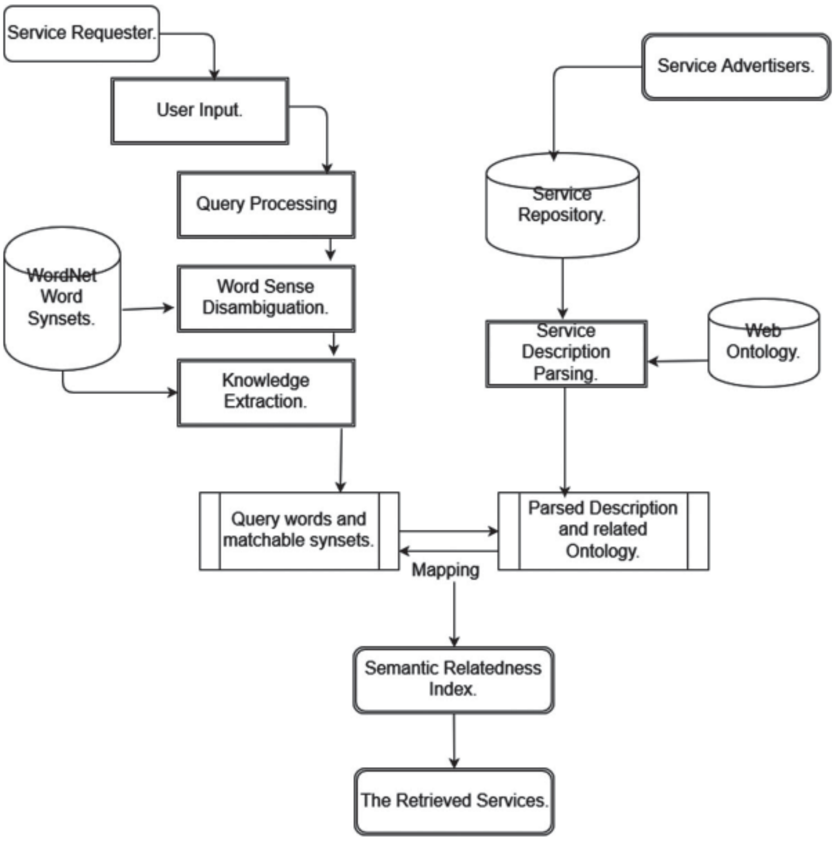

Figure 1. Proposed framework for hybrid matching.

Figure 1 shows the processing flow of how matchable elements from both the query and the service description are obtained. The treatment on both client and the server side is technically similar. It involves parsing, retrieving the required keywords and assigning the respective ontologies to them. The ontology is derived from WordNet for queries and the ontology base on the web. Finally, the description sets along

With their semantic sets [11] are tried and mapped to retrieve the best relatable service.

Query Processing.

The user query is processed and parsed to extract relatable words from it. The processing includes the following:

- Spelling correction is performed to check the linguistic accuracy of the user query. Suggestions are provided, and corrections are implemented.
- Tokenization is conducted to split the query into tokens, which correspond to the words.
- Stop Word Removal: To be able to process the words which are required, we remove the stop words such as *is, the, it etc.*. From the query
- Part-Of-Speech tagging would assign the part of speech to the word to map it to the service. The noun and verb pair would collectively be mapped to the service functionality, and the adjectives are related to the functional properties of the service.
- Lemmatization is done to obtain different inflict able forms of the word. A word 'publishing' can be replaced with a same sensed word 'publish' to increase the relatalbility of the word query set.
- Name entity recogniser would separate the Name entities from the query. This separate set of name entities can be mapped to service name or advertised names.

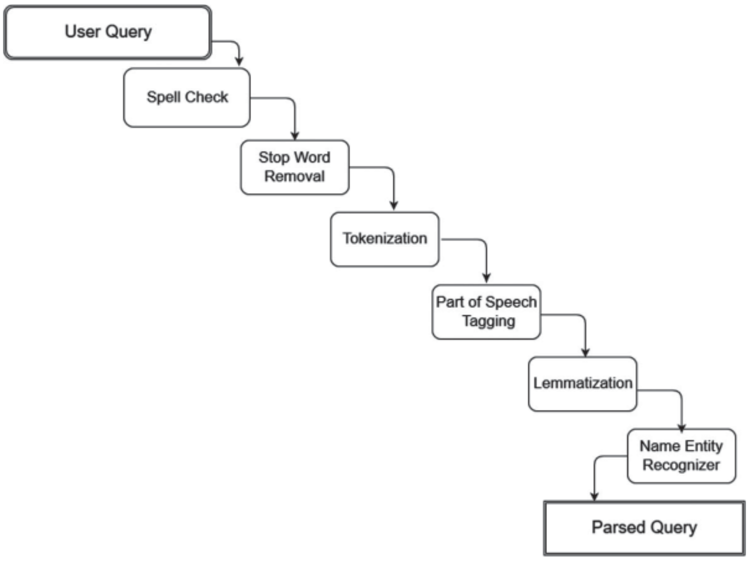

Figure 2. The steps of query processing.

Word Sense Disambiguation (WSD).

This module assigns the most synonimally similar sense as per the context to every word to bridge the gap between different descriptions of the same functionality on both client and server side. A Parts-of-Speech tagged query would be the beneficiary as it clearly specifies the context to the query.

The synset, most applicable sense is extracted from WordNet and is assigned.

Knowledge Extraction.

This module provides the ontology structure to the query word set to try and map it to the semantic knowledge representation of the service. The ontology is extracted from the WordNet's word-tree [12] structure, and different relatables of the word from WordNet are tried and mapped to the different structural class properties assigned to the server side.

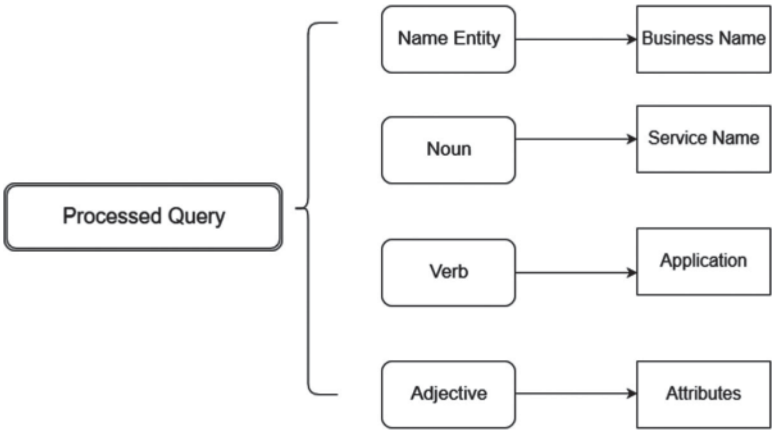

Figure 3. Shows the relatable entities in a user query.

The following are the different structural relatables provided by the WordNet:

- ***Hyponyms:*** Links broader synsets to more accurate ones.
- ***Eg:*** Chair: Armchair, Barber Chair, etc..
- ***Direct:*** Relates next subordinate branch of the word.
- ***Full:*** Relates all the branches to the word.
- ***Hypernym:*** Indicates the opposite of Hyponym, relates subordinates to superordinates.
- ***Direct:*** Relates to the immediate Superordinate to the word.
- ***Inherited:*** Relates all super-ordinates till the root.
- ***Meronym:*** Part-whole relation. Characteristic of a Super-ordinate is inherited onto the subordinate. Not the other way. (Subordinates may have specific characteristics)

Service Description Reading

To be able to map the request query and its WordNet Ontology to the advertised services, the service description should be retrieved, parsed and then assigned to the web service ontology assigned to it from the ontology database. Description of the service usually contains the data specifications and service performable operations and their related data. As the user would not be aware of the syntactic description of the service, the usable data is the operation description. Before any mapping is done, the description is parsed and service ontology is assigned to each of the service being looked for.

WSDL to OWL-S Conversion.

To increase the scope of the framework and to be able to retrieve from much more description files, we have to convert the existing WSDL-defined files into OWL-S files.

We use the 'Owl-S' Edit tool for the conversion.

The conversion would result in the following OWL-S files

1. Profile
2. Process
3. Service
4. Grounding.

Much of the description of the operations of the services resides in the profile and the grounding files. So, while parsing and matching, we give priority to the profile and the process files.

Description sets Mapping

In a conventional keyword mapping, the query words are tried and mapped to the parsed service description, and if an exact match is found, that particular service is retrieved. But in real time mapping, keywords have a narrow match and retrieve probability, and we cannot expect the user expressed necessity query to match specifically to the advertisement described service operations. This is where ontology helps.

Since difference exists among users side and server side expression of the service, we try and relate the ontology derived from both the sides. The idea is to gather the parsed query and its set of WordNet ontology to the advertisement defined service description and its ontology. The mapping would follow the order given below.

(a) ‘Namespaces’, if present would be tried and mapped with the ‘Business entity’ or the ‘Advertisers name’ in the service.

(b) The noun and verb pair would be tried and assigned to the operation description of the service. Usually the user mentions the functional necessity with verbs, nouns of the noun-verb pairs.

(c) The Sister Terms and the Domain Term Category are tried and mapped to the service description. This can be expected to help in reducing discovery complexity as it might bridge the expression gap between the client and the server side.

(d) The Hypernyms to the parsed query are tried and mapped to the Super-ordinates or the class related service of the ontology. Since the specificity of the service increases as we go deeper into the ontology branch, we choose rather to map a broader sense of the user query to the targeted services.

(e) The hyponyms are tried and mapped to the deeper or specific class objects of the ontology. This might not be much useful since mapping two sets with much-specified characteristics is not very helpful. We look for the immediate subordinate of the word while mapping.

(f) The holonyms are tried to be mapped with the ontology set related to the service. Since this is a part-whole relation, there is a chance of obtaining similar functionality or operation from the related services.

(g) Cross-sense-Words are tagged along to the words before beginning the mapping procedure.

Scemantic Relatedness.

While comparing the parsed user query to the service description, retrieval of the service is based on the extent to which the query sense is matching to the service description. To see the closeness or the relatability index, we choose to use the ‘nlkt’ synsets in python.

The verb-noun pairs and the functionality describing key-words of the target service are first looked for the semantic similarity.

We can import the synsets for each of the words and then calculate the relatedness. The following piece of code shows the index calculation.

```
>>> c = wordnet.synset ('client.n.01')
>>> s = wordnet.synset ('service.n.01')
>>> c. wup_similarity(s)
```

Mapping Algorithm.

```
Function MainLoop
Input: Query, Descriptions
Output: mappedList
mappedlist = null
for all Si in Descriptions do
if ( outputSem!=1) then
mappedList.append(Si);
end if
end for
return (mappedlist);
end Function
Function OutputSemMatch
Input: Query ontology set Qoutput = [x1, x2,.., xN],
Descriptions ontology set Doutput = [y1, y2,.., yM]
Output: outputSem;

for all xi in Qoutput do {
for all yj in Doutput do {
if (xi equivalent to yj) then
match1 = Exact
match2 += {yj}

else if (yj subsumes xi) then
match1 = Plugin
match2 += {yj}

else if (xi subsumes yj) then
match1 = Subsumes
match2 += {yj}
```

```
else
match1 = Fail
match2 = 0
end if
end for
match1= Max {match}
end for
outputSem= Min {match1}
return outputSem;
return {match2}
end Function
```

4. EXPERIMENTAL RESULTS.

The implementation of hybrid matching showed positive results. Each query from the user could now retrieve much more relatable services from the repository than the conventional keyword search. Also, many services without relatable descriptions could now be discovered.

Query	*Services Retrieved due to syntactic inference*	*Services Retrieved due to semantic inference*
"Transaction"	Epay.wsdl,	wallet.wsdl
"Retail"	Retail01.wsdl	Retail02.wsdl
"Offer Deals"	Deal.wsdl	Coupon.wsdl
"Feedback retriever"	Feedback.wsdl	Rating.wsdl

5. EXPERIMENTAL ANALYSIS

While publishing, the service advertiser may or may not define the operations in the service by mappable natural language. The syntactic matching algorithm would overlook this description. The hybrid framework implementation would relate to the misinterpreted services.

Example: While checking for a 'Retail Service' business service, the syntactic matching could only relate to the file "retail.wsdl.' Since the operations in the service were 'bid retail', 'Order stock 'etc. This framework related to the ontology and retrieve a 'retail2.wsdl.' service with operations defined as 'op1', 'op2', etc.. With the syntactic approach, it is not possible to retrieve the 'retrive2.wsd.' files with the abruptly defined operations.

6. CONCLUSION

In this paper, we have presented a hybrid discovery framework uses the two main concepts namely the English language and the ontological semantics over the conventional syntactic keyword discovery. The language semantics make the user query more sense full and allows the discovery framework to broaden its search spectrum. The ontology derived semantics help identify poorly or inaptly described services in the web. The combined approach would overcome the differences between the client and the advertiser in describing the service.

With the hybrid approach, discovery process identifying the semantically related classes and the operationally and functionally similar services. Also, the synonymity gap in the description is closed in by borrowing synsets of the query and the parsed description words.

References

1. K. Pu, V.Hristidis, N.Koudas "Syntactic rule-based approach to web service composition". 22nd International conference on Data Engineering (ICDE'06) 3rd to 7th April 2006.
2. Haiming liu, "Syntactic service match making based on fuzzy semantics of ontology" published in Electronics Communication and Control (ICECC). 9-11-2012.
3. W3C Web Services Description Language, http://www.w3.org/TR/wsdl.
4. URI (Uniform Resource Identifier) http://www.searchsoa.com/definition/URI.
5. T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," Scientific American, 2001, http://www.w3.org/2001/sw
6. W3C OWL Web Ontology Language. http://www.w3.org/TR/OWL.
7. NLP (Natural Language Processing for python) http://www.nltk.org.
8. Okkyung choi, "Semantic we bservice discovery system with QoS for enhanced web service quality." 4th international conference CDVE 2007.
9. Zhang Jia Jie 'Web Service Discovery Based on Semantic Matchmaking with UDDI' 2005.
10. Yatskevich and Fausto Giunchigl "Element level syntactic Matching using WordNet" 2007.
11. J. Javier Samper, F. Javier Adell, Leo van den Berg and J. José Martinez, "Improving semantic web service discovery," Journal of Networks, Vol. 3, No. 1, January 2008.
12. R. Navigli, P. Velardi, "Structural Semantic Interconnections: a Knowledge-Based Approach to Word Sense Disambiguation," IEEE.