A Novel Approach: A Hybrid Semantic Matchmaker For Service Discovery In Service Oriented Architecture

Soodeh Pakari[1], Esmaeel Kheirkhah[2], Mehrdad Jalali[3]

[1]Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran

[2]Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran

[3]Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran

ABSTRACT

Since service-oriented architectures make the commercial systems more reliable and reusable, they have gained more popularity in industry and scientific community in recent decades. Service-oriented architectures bring flexibility and reusability to software design. Due to the increasing number of services on the Web, finding a service which is suited to user requirements is crucial. The process of finding suitable services to user request is one of the main purposes of service-oriented architectures. Many methods have been proposed for service discovery in service-oriented architectures that try to fulfil user requirements and offer suitable services to user request; however the proposed methods do not have enough precision for discovering suitable services. In this paper, we propose a method for service discovery which offers more accurate services according to user request. The proposed method is a hybrid semantic matchmaker for service discovery in service oriented architecture. By providing accurate services suitable to user requests, we have greatly increased the reusability rate and reduced the time and cost of software development

.

KEYWORDS

Service Oriented Architecture, Service Discovery, Ontology

1. INTRODUCTION

One of the main activities in developing Service-Oriented Architectures is to identify and discover services which fulfill the requirements of a software product. This paper specifically focuses on the problem of service discovery, as an increasing number of companies and organizations would benefit from service discovery in Service-Oriented Architectures.

Service discovery approaches can help organizations to identify their required services and use them in the development process. This can avoid re-implementing services which already exist, resulting in increasing re-usability. To our knowledge, the proposed methods in the literature by far do not provide adequate precision for discovering suitable services. In this paper, we propose a method for service discovery that offers more suitable services according to user's request.

DOI : 10.5121/ijnsa.2014.6104

One of the benefits of Service-Oriented Architectures is the use of services which are provided outside of the organization, resulting in cost saving and acceleration of the development process [1]. Hence, relevant mechanisms and methods are needed to identify services to help the organizations. This is when service discovery issue arises; i.e. matching customer requirements with the information provided in service description by service provider [2]. In most cases, service discovery is performed after service identifying. In service identifying, with analyzing system requirements, it is recognized which services are needed. In the next step, either services are produced or using service discovery methods, services corresponding to identified services can be discovered. Development of Service-Oriented Architectures and an increasing number of service providers have led to a situation where new methods are required to describe and discover services. Different approaches for service discovery have been proposed and employed that try to offer suitable services to the user. Here, we review some of these methods.

2. RELATED WORKS

Here, we discuss semantic service discovery methods, semantic annotations was invented to automation the entire lifecycle of a service from providing to invoke and run it. Specially, important role of it in service discovery will be affected every aspect of this process involves the architecture, algorithms and tools. Semantic service discovery approaches try to perform similarity matching based on semantic annotation and ontology matching. The semantic matching is able to overcome the insufficiencies of syntactic discovery, but these approaches are much more complex than key word-based approaches. There are three categories of these approaches, domain ontology-based approaches, public ontology-based approaches, semantic and syntax-based approaches. . We discuss methods of each sub-group, and then subsequently compare the sub-groups in terms of their merits and drawbacks.

2.1 Domain ontology-based approaches

Methods proposed in [3], [4], [5], [6], [7], [8], [9], and [10] have used domain ontology for service discovery, there are some advantages and disadvantages of domain ontology based approach that we mention them briefly:

Advantages:

- The aim of Semantic Web Service technology is to reduce the manual discovery and usage of Web Services, by allowing software agents and applications to automatically identify, integrate, and execute these Web Services to achieve the user purposes [11].
- With increase in number of similar Web Services, one of the important issues is to discover relevant Web Service with regard to user specification. Relevancy of Web Service discovery can be improved by augmenting semantics through expressive formats like OWL [12].
- Pledge the automation of Web Service discovery process [13].

Disadvantages:

- Some proposed discovery methods are based on a user request that is expressed in a specific semantic description language like OWL-S, WSMO, or WSDL-S. As a result, end user should have intimate knowledge of Semantic Web Services and related description and implementation details which makes their usage difficult for end users [11].

- It would limit the discovery process to specific advertised services. Also, from the service requestor's perspective, the requestor may not be aware of all the knowledge that constitutes the domain ontology. Specifically, the service requestor may not be aware of all the terms related to the service request. As a result, many services relevant to the request may not be discovered in the service discovery process [11].
- Another limitation of some proposed methods consist on their semantic matching approaches. In fact, both service provider and service requester use domain ontologies to build semantic service description file. Most of proposed approaches assume that both service provider and service requester use the same ontology domain to describe service but is not applicable in real-world scenario. To overcome this ontology heterogeneity, it is needed to utilize ontology mapping techniques to coordinate the differences between these ontologies to support interoperability [11].

.

2.2 public ontology-based approaches

[14] ‹[15] ‹[16], and [2] are in this group, now we explain some advantages and disadvantages of these approaches.

Advantages:

- Most important advantage of public ontology-based approaches is to allow developers to enhance Web Services with semantic information without semantic annotation against an ontology. These methods are different from traditional, ontology-based researches, which require significant cost and effort for semantic annotation and ontology management [14].
- WordNet, It is not domain specific and eliminates the semantic annotation cost of services [14].

Disadvantages:

- In general, WordNet is too fine-granular for many purposes [17].
- There are WordNet versions for a large number of languages, but there is no real multi-lingual WordNet. The different WordNet differ in coverage, format, and availability [17].

2.3 Semantic and syntax-based approaches

[18] ‹[19] ‹[20], and [21] are in this group, both semantic and syntax based approaches have some advantages and soma disadvantages, with semantic and syntax based approaches we want to use advantages of both and decrease their disadvantages, we mention some disadvantages of them here.

Disadvantages of syntax-based approaches:

- Not suited for automatic processing [13].
- Still required human interaction [13].
- Syntax-based approaches do not allow re-planning a Web Services workflow on the way in case a service fails [12].

- They can't choose the most similar service among a large set of available and semantically similar services [12].
- Another limitation of keyword search is that user can't describe the search request more precisely than keyword. Furthermore, keywords do not suffice for accurately specifying user's information needs[21].

Advantages of syntax based approaches:

- Simple and widely used technique [13].
- Standards like UDDI exist [13].
- They offer a simple syntax in terms of a list of key word phrases that users can use their own words to express their information requirement. Also, keyword-based search is more familiar to the user [11].

Table 1: merits and shortcomings of service discovery methods

	merits	***shortcomings***
Domain ontology-based approaches	1-Minimize the manual discovery and usage of Web Service by allowing software agents to automatically and dynamically discover Web Services 2-Effective and reliable technique 3-Relevancy of Web Service discovery can be improved by augmenting semantics through expressive formats like OWL	1-they require the end user to have intimate knowledge of Semantic Web Services and related description and implementation details which makes their usage difficult for end users 2- The discovery scope of these approaches is often limited to some Web Services that are published in a specific description standard. 3- The service requestor may not be aware of all the terms related to the service request. 4- We need to use ontology mapping techniques to coordinate the differences between these ontologies to support interoperability.
Public ontology-based approaches	1-It allows developers to enhance Web Services with semantic information without semantic annotation against an ontology. 2- WordNet, It is not domain specific and eliminates the semantic annotation cost of services. 3- Terms and concepts in WordNet has its	1-Different parts of WordNet have different granularity for the description of word senses. In general, WordNet is too fine-granular for many purposes. 2- There is no real multi-lingual WordNet.

	precise semantic	
Syntax and semantic-based approaches	1-Simple and widely used technique. 2- Standards like UDDI exist. 3- keyword-based search is more familiar to the user.	1- Still required human interaction 2- can't choose most similar service among a large set of available and semantically similar services

3. PROPOSED APPROACH

As we mentioned in section 2, every method has some advantages and some disadvantages, because of this, we hybrid them to have a better result. In our approach we use logical-based method for domain ontology and structural-based method for public ontology (WordNet), also we use syntactic method for service discovery, we explain each in detail

3.1 Syntactic similarity with Jaro-Winkler strategy

This algorithm calculates a normalized score for similarity between two strings. Calculation is based on the number of matching characters in two strings and also number of transpositions.
The Jaro-Winkler Distance is really an addition to the Jaro Distance, represented mathematically below:

$$d_j = \begin{cases} 0 & if\ m = 0 \\ \frac{1}{3\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right)} & otherwise \end{cases} \qquad (1)$$

The mathematical equation states that the Jaro distance is 1/3 multiplied by the number of differences (m) divided by the length of the first string (S_1) plus the number of differences (m) divided by the length of the second string (S_2) and lastly plus the number of differences (m) minus the number of transpositions (t) divided by the number of differences (m). The 1/3 is very important because the max the number inside the parentheses can reach is 3, but the distance wants a number between 0 and 1.

A difference between the strings is any character that is not the same to another character within a "match distance" in the opposing string. Below is the formula for determining the "match distance".

$$\left\lfloor \frac{\max(|s_1|,|s_2|}{2} \right\rfloor \qquad (2)$$

The "match distance" is defined as the maximum of the two string lengths divided by 2 and then subtract 1.

Lastly, the Jaro-Winkler distance is a modification of the Jaro distance. The formula is given below.

$$d_w = d_j + (lp(1 - d_j)) \tag{3}$$

The Jaro-Winkler formula uses the Jaro distance, but adds the result of l, which is the length of the common prefix shared by the two strings with a maximum of 3. Multiply that by the p constant, which is usually 0.1 multiplied by the quantity of 1 minus the Jaro distance [22].

3.2. Structural Similarity based on WordNet

We use WordNet as a public ontology for computing semantic and structural similarity between concepts, for this purpose, we use WuAndPalmer algorithm [23]. The formula to calculate similarity between concepts is:

wp = (2*depth(lcs(c_1,c_2))) / (depth(synset1) + depth(synset2)) (4)
Where LCS is least common subsumer of c_1, c_2.

Wu and Palmer algorithm calculates semantic similarity between concepts and it consider only Is-A relation and it suggests a similarity measurement which computes the similarity between two concepts, considering path length between concepts in WordNet taxonomy. The calculation is according to formula (5):

$$sim_{wp}(c_1, c_2) = \frac{2C}{A+B+2C} \tag{5}$$

$$A = len(c_1, lcs(c_1, c_2))$$

$$B = len(c_2, lcs(c_1, c_2))$$

$$C = depth(lcs(c_1, c_2))$$

A+B is the path length from c_1 to c_2, and C is the path depth in taxonomy [24].

3.3. Semantic similarity base on domain ontology

We have a domain ontology here; first we find two concepts in ontology, then find their relation, this relation can be “subclass of”, “superclass of”, “equivalent”, “disjoint”, we give a weight to each relation and then calculate weighted average. For computing final similarity between concepts, we average between three amounts of mentioned similarity.

3.4 Hybrid approach:

After computing three similarities (syntactic similarity, structural similarity, and semantic similarity), we hybrid them with a weighted averaging to have a better result. The hybrid formula is:

Final-similarity= (0.4* semantic similarity + 0.3*structural similarity+0.3*syntactical similarity)/3 (6)

With this similarity computation, we can improve the result of matchmaking and more accurate services will be found.

This approach applies to OWL-S service descriptions and for each owls file, we use only service profile. The fields which are compared are Input, Output, Result, Precondition, Service name, and Service description.
Using this kind of similarity measure, the precision of discovered services will improve because we consider three kind of similarity measure (syntactical similarity, structural similarity, semantic similarity).

Semantic matchmaking matches registered services S with desired service description Q according to figure1.

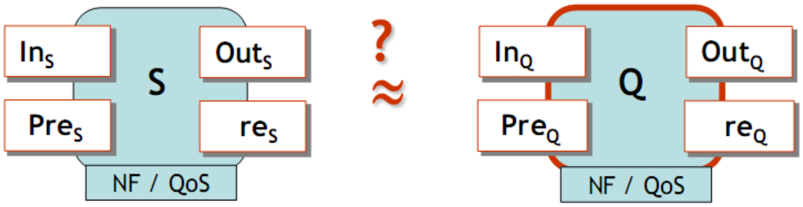


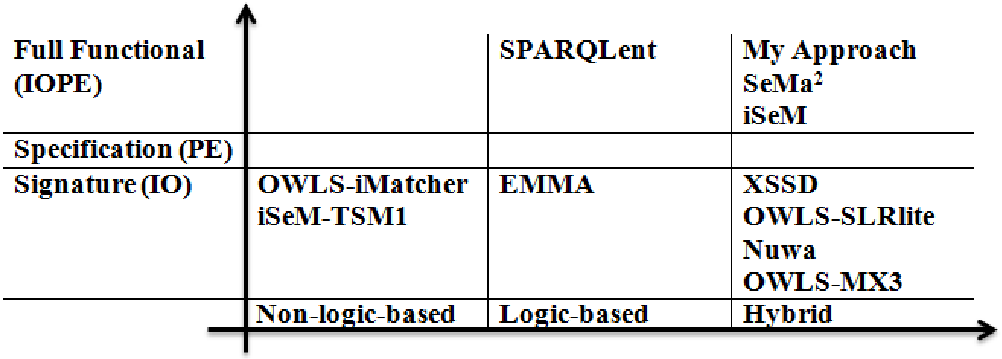
Figure 1: matching of registered service and desired service

3.5 comparing with other approaches

To compare our work with other matchmakers, we can consider matchmakers in S3[1] contest [25]. The S3 contest evaluates the retrieval performance of Semantic Web service matchmakers over given test collections for most prominent semantic service formats such as OWL-S, WSML and the standard SA-WSDL. In S3 contest classifies all participants of the contest according to a classification of semantic service matchmakers.

Current semantic service matchmakers including those which participated in the S3 contest can be classified according to (a) the kind of semantic selection they perform, and (b) what parts of the semantic service description they exploit for this purpose. In particular, we may distinguish between means of logic-based, non-logic-based and hybrid semantic service selection based on the service signature (input/output, IO), the specification (preconditions/effects, PE), the full functional profile (IOPE).The classification of all contest participants is summarized in figure2: Since all of them perform semantic matching on Either the service signature or the full profile, the figure does not show the remaining levels of the dimension of semantic service description parts. Further, the majority of contested matchmakers performs hybrid semantic matching of service annotations.

[1] Semantic Service Selection



Full Functional (IOPE)		**SPARQLent**	**My Approach** **SeMa^2** **iSeM**
Specification (PE)			
Signature (IO)	**OWLS-iMatcher** **iSeM-TSM1**	**EMMA**	**XSSD** **OWLS-SLRlite** **Nuwa** **OWLS-MX3**
	Non-logic-based	**Logic-based**	**Hybrid**

Figure 2: classification of semantic matchmakers in S3 contest

OWL-S Service Matchmakers

1. iSeM 1.1 (DFKI, Germany)
2. OWLS-MX3 (DFKI, Germany)
3. SeMa2 v2 (TU Berlin, Germany)
4. Nuwa-OWLS (URJC Madrid, Spain)
5. OWLS-iMatcher (U Zurich, Switzerland)
6. SPARQLent (HP, Italy)
7. OWLS-SLR (Aristotle U of Thessaloniki, Greece)
8. XSSD (Beihang U, China)
9. EMMA (U Seville, Spain)
10. iSeM-TSM1 (Payame Noor U, Iran / DFKI)

iSeM 1.1

–Selection: Hybrid; Signature (I/O), Specification (P/E), Service description tag
- Logic-based matching
- Non-logic-based matching

iSeM-TSM1

–Selection: Non-logic-based; Signature (I/O), Service description tag
- Non-logic-based matching

Nuwa-OWLS

–Selection: Hybrid; Signature (I/O), Service description text
- Logic-based matching
- Non-logic-based matching

- **SeMa2 v2**

–Selection: Hybrid; Signature (I/O), Specification (P/E)
- Logic-based matching
- Non-logic-based matching

- **OWLS-SLR lite**

– Selection: Hybrid; Signature (I/O), Non-functional parameters
- Logic-based match
- Non-logic-based match

- **OWLS-iMatcher**

– Selection: Syntactic; Signature (I/O)
- Non-logic-based

- **OWLS-MX3**

– Selection: Hybrid, adaptive; Signature (I/O)
- Logic-based match
- Non-logic-based match

- **SPARQLent**

– Selection: Logic-Based; Signature (I/O), Specification (P/E)
- Logic-based match

- **XSSD**

– Selection: Hybrid; Signature (I/O), Service description tag
- Logic-based match
- Non-logic-based match

- **EMMA**

– Selection: Logic-based semantic; Signature (I/O)
- Logic-based pre-filtering [25]

As you see in figure2, our approach is a hybrid matchmaking and also use full functional (IOPE) and also service discription and service name in matching. Because this paper is a conceptual framework, we will have a toy example to show the result.

3.5. An example for evaluating proposed approach

We have two service descriptions in OWL-S, namely, "*Buy Airplane Ticket*" and "*Purchase Flight Ticket*" and want to calculate similarity between them according to previous sections. We use *input*, *output*, *result*, *precondition*, *service name*, *service description* of each OWL-S files to calculate similarity.

These fields are as bellow for "Buy Airplane Ticket":

Input :FlightNumber

Output :Confirmation

Precondition :FlightAvailable

Result :PlaneTicketIsBought

Service name :Buy Airplane Ticket

Service description :This service buys a ticket for a specified flight

And for "Purchase Flight Ticket" are:

Input :flight

Output :acknowledgement

Precondition :flight available و credit card charged

Result :flight ticket purchased

Service name :Purchase Flight Ticket

Service description :This web service will purchase a ticket for a specified flight

Now, according to previous sections, we calculate the similarity between corresponding fields and then according to formula (6), final similarity is calculated. The similarity between these two service descriptions is 0.803. The calculated value is reasonable, and as you see, the proposed approach will discover similar services and offers them to the user as candidate services. This approach will increase precision in service discovery process.

4. CONCLUSIONS

In this work, by hybriding the proposed methods such as domain ontology-based approaches for logical subsumption and public ontology-based approaches that we have used WordNet to consider structural similarity between concepts, and also syntactic similarity, we proposed a new approach that by utilizing the advantages of mentioned methods and hybrid them achieve more accurate results and discover more suitable services.

REFERENCES

[1] T. Erl, Service-Oriented Architecture: Concepts, Technology, and Design, Upper Saddle River, NJ: Prentice Hall PRT, 2005.

[2] J. Becker, O. Mueller, M. Woditsch, "an ontology-based natural language service discovery engine-design and experimental evaluation," in the proceeding of the European conference on information systems, Pretoria, South Africa, 2010.

[3] V. Oleshchuk, "ontology-based service matching and discovery," in 6th international conference on intelligent data acquisition and advanced computing systems, sept 2011.

[4] A. V. Paliwal, "semantic-based automated service discovery," IEEE transaction on service computing, vol. 5, no. 2, pp. 260-275, April-June 2012.

[5] Y. Shi, G. Li, J. Li, "framework of semantic web service discovery based on ontology mapping," in international conference on research challenges in computer science, Dec 2009.

[6] K. Zamanifar, A. Zohali, N. Nematbakhsh, "matching model for semantic web service discovery," in international conference on advanced information networking and applications workshops, May 2009.

[7] B. D. Martino, "semantic web service discovery based on structural ontology matching," international journal of web and grid services, vol. 5, no. 1, pp. 46-65, 2009.

[8] L. Zhou, "an approach of semantic web service discovery," in international conference on communications and mobile computing, April 2010.

[9] Ch. KE, Zh. Huang, "self-adaptive semantic web service matching method," in Knowledge-based systems, Nov 2012.

[10] A. B. Bener, V. Ozadali, E. S. Ilhan, "semantic matchmaker with precondition and effect matching using SWRL," Expert systems with applications, vol. 36, no. 5, pp. 9371-9377, July 2009.

[11] A. Adala, N. Tabbane, S. Tabbane, "A Framework for Automatic Web Service Discovery Based on Semantics and NLP Techniques," advanced in multimedia-special issue on web services in multimedia communication, vol. 2011, no. 1, January 2011.

[12] R. Suganyakala, M. Aarthilakshmi, G. R. Karpagam, S. Maheswari, "ontology based Qos driven web service discovery," international journal of computer science issues, vol. 8, no. 4, pp. 191-198, July 2011.

[13] A. Pradnya Khutade, B. Rashmi Phalnikar, "Qos-based web service discovery using OO concepts," international journal of advanced technology & engineering research, vol. 2, no. 6, pp. 81-86, Nov. 2012.

[14] R. Karimpour, F. Taghiyareh, "conceptual discovery of web services using WordNet," in Service Computing conference, Singapore, Dec. 2009.

[15] G. Ganapathy, C. Surianarayanan, "an approach to identify candidate services for semantic web service discovery," in IEEE international conference on service-oriented computing and applications, Perth,WA, Dec. 2010.

[16] Y. Peng, Ch. Wu, "automatic semantic web service discovery based on assignment algorithm," in 2nd international conference on computer engineering and technology, Chengdu, April 2010.

[17] M. Pinkal, A. Koller, semantic theory, 2005.

[18] A. Faroog, R. Arshad, "an efficient technique for web services identification," international journal of multidisciplianary sciences and engineering, vol. 2, no. 1, pp. 26-30, March 2011.

[19] C. B. Merla, "context-aware match-making in semantic web service discovery," international journal of advanced engineering sciences and technologies, vol. 9, no. 2, pp. 243-247, 2011.

[20] M. Klusch, P. Kapahnke, B. Fries, "hybrid semantic web service retrieval: a case study with OWLS-MX," in IEEE international conference on semantic computing, Santa Clara, CA, Aug. 2008.

[21] J. R. Raj and T.Sasipraba, "web service discovery based on computation of semantic similarity distance and Qos normalization," Indian journal of computer science and engineering, vol. 3, no. 2, pp. 235-239, May 2012.

[22] R. Bayer, "StrCmp," 2013. [Online]. Available: people.rit.edu/rmb5229/320/project3/jaro-winkler.html.

[23] Z. Wu, M. Palmer, "Verb semantics and lexical selection," in In: Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics, 1994.

[24] T. V. d. Cruys, "Evaluation of Wordnet-based similarity," in Mining for Meaning, 2010.

[25] M. Klusch, "5th International Semantic Service Selection Contest - Performance Evaluation of Semantic Service Matchmakers -," DFKI, Germany, April 12, 2013.