Proposed Conceptual Model for Semantically Enabled Web Services Based On QoS

Anjan Bandyopadhyaya\textsuperscript{a}, Tapas Si\textsuperscript{b}, Partha Protim Mondal\textsuperscript{c}, Saurav Mallik\textsuperscript{d}

\textsuperscript{a}Asst.Prof (C.S.E.), Mallabhum Institute Of Technology, India
\textsuperscript{b}Asst.Prof (C.S.E), Bankura Unnayani Institute Of Engineering, India
\textsuperscript{c}Asst.Prof (C.S.E), Mallabhum Institute Of Technology, India
\textsuperscript{d}Asst.Prof (C.S.E), Mallabhum Institute Of Technology, India

Abstract

Presently, the service discovery is the process of discovering one or more documents that describe in a particular service. Most of the service discovery mechanisms perform syntactic matching. That means it supports the key word based search. This often leads to poor service discovery result. Another drawback of the existing service discovery mechanism is that the query service matching score is calculated and taking an account only the keywords from the user query's and the terms is the service descriptions. The keyword based service discovery mechanism supported by Universal Description Discovery and Integration (UDDI) and most of the existing service search engines like Yahoo and Google suffer from keyword based search. Firstly it is difficult for the user to obtain the desired services because the number of the retrieved services with respect to the keywords may be huge. Secondly, keywords are insufficient in expressing the semantic concepts and semantically different concepts could possess identical representation (homonyms), which will further low accuracy. This warrants the need to establish an effective and reliable process of web service discovery. This research has emerged to develop methods and model to improve the accuracy of the best web service discovery to match the best service. This paper presents a survey of web service discovery systems, focusing on systems that support either semantics based or syntactic based approach.

1. Introduction

The process of discovering and invoking relevant services should be hidden from the users' point of view. In order to realize this scenario, provide mechanisms smart service discovery based on the current situation of the user (e.g., user's location, his interest, user's environment characteristics etc.). Most of the existing service discovery mechanisms retrieve services descriptions that contain particular keywords.
from the user’s query. The query keywords might be semantically similar but syntactically different from the terms in service descriptions e.g. ‘investment’ and ‘saving’ (synonyms). The query keywords also might be syntactically equivalent but semantically different from the terms in the service description. Another problem with keyword-based service discovery approaches is that they cannot completely capture the semantics of user’s query because they do not consider the relations between the keywords. By definition, context is a situation of an entity (person, place or object) that is relevant to the interaction between a user and an application [9]. Therefore, considering the context in the query-service matching process can improve the quality of the retrieved results. One possible solution for this problem is to use ontology-based retrieval. In this approach, ontologies are used for classification of the services based on their properties. This enables retrieval based on service types rather than keywords. This paper presents a novel approach for service discovery that uses ontologies to capture the semantics of the user’s query, of the services and of the contextual information that is considered relevant in the matching process.

2. Literature Review

2.1. Existing Web Site Model

As the number of functional similar Web service increases in the Internet, QoS based Web service selection has become a hot research. In literature [3], it provides a Web Service QoS(WS-QoS) architecture that enables QoS aware service specifications as well as the broker based Web service selection model that enables an efficient QoS-aware service selection. Literature [4] designs the service selection algorithms to meet end-to-end QoS constraints. It models the problem as the Multiple Choice Knapsack Problem and provides efficient solutions. Literature [5] proposes an extended implementation of the UDDI specification combines the semantic based Web service match with the recommendation system and provides a Web service selection framework. Discovery of Web services is becoming of an immense interest to many governmental and business organizations, and there have been numerous efforts that attempted to find alternative means to improve the discovery of Web services. the peer-to-peer framework for ranking of Web services [6] by concentrating more on the functionality served by a Web service, providing keyword-based search engine for querying Web services [7], search engine for Web services (SWoogle) [8], and semantic discovery and ranking of Web services using signatures found in WSDL documents through a sequence of types in its inputs and output variables and using a partial matching technique [9]. However, these approaches provide a very limited set of search capabilities (i.e. search by business name, business location, etc), and do not take into consideration the process behaviour which is becoming an essential part of the functionality of services (i.e. how service functionality is served).

2.2. The Arrangement of Web Services

Each Web service has an associated XML-based document called WSDL. WSDL file describes Web service functionality and interface information as shown in fig.2. The service implementation definition describes how a service interface is implemented by a given service provider, and the service interface definition contains the business category Information and interface specifications that are registered as UDDI tModels. Each Web service consists of a set of operations. Regardless of the invocation information in WSDL that is useless for similarity matching, such as the binding and the port, we can identify three types of metadata from WSDL.

- **Name and text description:** A Web Service is described by a name, a text description in the WSDL file, and a description that is put in the UDDI registry.
- **Operation descriptions**: Each operation is described by a name and a text description in the WSDL file.

- **Input/output descriptions**: Each input and output of an operation contains a set of parameters. For each parameter, WSDL file describes the name, data type (if the parameter is of array type). Parameters may be organized in a hierarchy by using complex types.

### 3. Proposed Conceptual Model for Web Services

#### 3.1. Proposed Work

In order to improve the service discovery satisfaction, we propose to add clients’ interests into QoS and form a new extended QoS model that at least take care of the following functions:

1. Receive and Retrieve a client's request.
2. Offer the client's request to the UDDI Registry.
3. Collect the client's interest.
4. Collect Qos (Time, Cost, Reliability, availability) of WS Providers.
5. Select the best Web service for the client as a ranking basis.
6. Match Web Service based on the collected and extended QoS.
7. Maintain the best quality of the services.
8. Select the best Web service for the client.

Figure 1 depicts the conceptual model for web service.

![Conceptual model for web service](image)

#### 3.2. Proposed Model

Service discovery is meant for searching for services of required qualities in service catalog on the basis of services’ “metadata”. In this proposed model, a well-organized Web service has been selected from the set of mapped services through ranking scheme. The architecture of proposed model is described below:
- **Requester query**: The user who is an intermediate level application developer has to query the model by posting required service’s name through keyword with or without QoS functionalities as scalability, accuracy, availability, cost, reusability etc. If provided these QoS functionalities are taken as additional input for service selection.

- **Information Resource**: An information search can be done either syntactic or semantic based. In the syntactical case a “keyword” based match is applied and, in the semantic search a concept based match is employed on the basis of the concepts defined by the ontologies used.

- **UDDI Service Registry**: UDDI registry is commonly used data store of Web services. It contains services from various domain provided by various providers with their own functionality and originality. The services retrieved are converted into ontologies using semi/fully automated converter and stored in library for future reference.

- **Web Service Store**: This is the location where the ontologies defining the concepts needed for service definitions are stored. There are two ways of ontology definition. The one is to develop ontologies from scratch. And the other is to re-use the ontologies already existing on the Web. The second option requires ontology evaluation to determine the most suitable ontologies to be used.

- **Web Service Extraction**: Services that have most similar correspondence to the user’s request are extracted from the registry. This process has been done through the usage of the synonym set of users request utilizing the service from knowledge resource.

- **Mapping Patterns**: Mapping pattern is a template that is used to define the data structure for mapping two ontologies. The data structure differs according to the pattern. Each pattern has its own functionalities.

4. Discussion of the Proposed Work:-

The proposed semantic web service creation is elaborated with the help of the web service creation. In this proposed work, the proposed model is implemented into the web service. For better explanation, the following related example is given.

In this particular example, Concept, Attribute, Relation and Axiom are defined as follows:

- **Concept**: conceptual entity of the domain
- **Attribute**: property of a concept
- **Relation**: relationship between concepts or properties
- **Axiom**: coherent description between Concepts / Properties / Relations via logical expressions

The propositional logic is given below:

- \( \text{holds(Professor, Lecture)} \Rightarrow \text{Lecture.topic} \)
- \( \text{Lecture.topic} \in \text{Professor.researchField} \)

According to the given propositional logic, Figure 2 depicts the demonstration of proposed semantic relationship. In this figure students and professor both are person, here professor have research field but students have no research field. So we can semantically identify easily both the students and professor.
5. Conclusion

This paper presents the conceptual model for web service creation in syntactic or semantic way. In this paper we have tried to provide a glimpse of the huge spectrum of work investigated by researchers globally in the field of Discovering the Right Services either semantically or syntactically and highlight the advantages and disadvantages of each system.

References