Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 12, December 2013, pg.283 – 287

SURVEY ARTICLE

A SURVEY ON GENERIC QUERY MODEL FOR THE HETEROGENEOUS SERVICES TO MAKE UNIFIED DISCOVERY

C.AGJELIA LYDIA[1], K.S.KANNAN[2]

PG STUDENT, ASSISTANT PROFESSOR

NPR COLLEGE OF ENGINEERING AND TECHNOLOGY, TAMILNADU, INDIA

EMAIL: *agjelia@gmail.com, saikannan2012@gmail.com*

Abstract: Web service discovery involves service categorization and enhancement of the service request. It incorporates the clustering for accurately classifying the web services based on service functionality validates the effectiveness and feasibility of the proposed approach. Existing service discovery mechanisms lacks the flexibility and scalability of web services and limited to keyword based match making process results in lesser accuracy in terms of similarity metrics. Proposes the generic query mechanism called Proteus which incorporates the unified discovery of heterogeneous services. Web service discovery the web service providers publishes the service and the client uses the service. Publishing a Web service involves creating a software artifact and making it accessible to potential consumers. Service Providers maximizes the Web service endpoint with an interface description using the Web Services Description Language. The provider can explicitly register a service with a Web Services Registry such as Universal Description Discovery and Integration. The service users can search Web Services manually or automatically. The current UDDI search mechanism can only focus on a single search criterion.

Index Terms-- Web services publishing; web service discovery; service discovery process and methodology

1 INTRODUCTION

A large number of web services structure a service oriented architecture and facilitate the creation of distributed applications over the web. These web services offer various functionalities in the areas of communications, data enhancement e-commerce, marketing, utilities among others. Some of the web services are published and invoked in-house by various organizations. These web services may be used for business applications, or in government and military. However, this requires careful selection and composition of appropriate web services. The web services within the service registry (UDDI) have predefined categories that are specified by the service providers. As a result, similar services may be listed under different categories.

Given the large number of web services and the distribution of similar services in multiple categories in the existing UDDI infrastructure, it is difficult to find services that satisfy the desired functionality. Such service discovery may involve searching a large number of categories to find appropriate services. Therefore, there is a need to categorize web services based on their functional semantics rather than based on the classifications of service providers. web services add a new level of functionality to the Web, a step toward an open environment of distributed applications. Although current Web service technologies built around SOAP and the Web Services Description Language (WSDL) form a solid foundation, scaling will be difficult without a proper degree of automation. In large-scale, open, and heterogeneous environments.

Web services' success depends on resolving the fundamental challenges that existing integration technologies don't address namely, search, integration, and mediation.XML descriptions of Web services support integration in rigid workflow or services configurations, but automation requires more than a description of the data structure and syntax. The Web supports distributed computing, and the term service-oriented architecture (SOA) was created in particular as a generalization of Web services technologies. The Web is already a vast data repository, and Web services are growing very quickly. To make sense of all this data and these services, the Semantic Web builds on the foundations of logic and knowledge representation to help computers find the right information for their users. Finding and combining information is only part of the vision, however. Computers must also be able to find and combine services on the Web to free users' hands and make the Web of services scale together with the Web of data.

A web service comprises of several operations given in WSDL and communicates with users and other services through input/output SOAP packages. The growing number of services raises a challenging problem: locating the desired service. To address this problem, search engines aiming at keyword search have been developed. As web services described by WSDL contain very few text fragments traditional IR techniques such as TF/IDF and LSI, which rely on term frequency to capture the underlying semantics in WSDL, do not apply well. Considering the problem above, Woogle and URBE have been built. To capture the semantics of terms within WSDL of a service, developers in propose an approach which clusters parameter names in the collection of web services into semantically meaningful concepts, and then applies TF/IDF on the concepts to obtain the similarity of services. The work in utilizes external knowledge, the lexical database, Wordnet, to obtain the semantic distance of terms extracted from the operations, input/output parameters of WSDL. A service is then considered as a bag of terms. The similarity of services is thus measured upon these semantic distances.

Web services are software entities that have a well define interface and perform a specific task. Typical examples include services returning information to the user, such as news or weather forecast services, or services altering the world state, such as online shopping or booking services. A web service is formally described in a standardized language (WSDL). The service description may include the names and types of input and output parameters, preconditions and effects, as well as Quality of Service (QoS) attributes, such as price, execution time, availability, and reputation.

As web services and service providers proliferate, there will be a large number of candidate, and likely competing, services for fulfilling a desired task. Hence, effective service discovery mechanisms are required for identifying and retrieving the most appropriate services. Assume the existence of a repository that contains a large number of advertised service descriptions. In a typical scenario, a user provides a complete definition of the requested service, and issues a discovery query. The repository, then, employs a matchmaking algorithm to identify services relevant to the user's request. Note that perfect matches, i.e., services with the same description as the request, are seldom found. Furthermore, even when a perfect match exists, it may not constitute the most desirable option, e.g., the service is currently unavailable. For these reasons, given a request, the matchmaking algorithm needs to consider a potentially large number of partial matches, and to select the best candidates among them.

2 LITERATURE SURVEY

2.1 SEMANTIC ANNOTATIONS FOR WSDL AND XML SCHEMA

Semantic Annotations for WSDL and XML Schema (SAWSDL) points to semantics from the Web Services Description Language (WSDL).Represent the semantics, need to reach for knowledge representation languages.

The major technologies for Web services are SOAP and WSDL. SAWSDL extends WSDL with pointers to semantics that are crucial for achieving automation. Adding semantics to Web services mainly aims to automate certain tasks that must be performed with services before or during invocation.

Based on various efforts in SWS and service-oriented computing communities (such as OWL-S and WSMO), the generally accepted tasks are discovery, negotiation, filtering, selection, and invocation. Set of extension attributes which can be associated to the elements of a WSDL definition. Different tasks require different semantic descriptions. Describe four aspects of services semantically. Information semantics define an information model (an ontology together with instance data).

Functional semantics describe the service capability that is, what the service can offer its users. These are used for service discovery, comparing a user's need and the functional descriptions of available Web services. Service composition also uses functional semantics (together with information semantics) when creating a plan for a given goal.

Nonfunctional semantics define additional constraints and policies over service functionality that functional semantics don't capture. Nonfunctional semantics are needed to match discovered services against users preferences and constraints.

Behavioral semantics describe a service's public and private behavior. A description of the public behavior is a protocol that each client must follow in order to consume the service's functionality. It guides negotiation with the service as well as its invocation. Service discovery or composition might also incorporate compatibility checks or process mediation between the client's and the service's expected behaviors. Semantic Annotations for WSDL based semantic templates can be used for dynamic binding. It can be replaced with actual services either at run-time or deployment time to achieve dynamic binding. SAWSDL support for data mediation. User specified mappings from Web service message element to semantic model concept. It doesn't recommend annotating interfaces with nonfunctional properties.

2.2 SEMANTICS-BASED AUTOMATED SERVICE DISCOVERY

Semantic-based categorization of web services is performed at the UDDI that involves semantics augmented classification of web services into functional categories. The proposed approach focuses on semantic-based service categorization of the web services and selection of services based on semantic service description rather than syntactic keyword matching. Approach needs to be generic and should not be tied to a specific description language. The semantic categorization of UDDI wherein we combine ontologies with an established hierarchical clustering methodology, following the service description vector building process.

Semantic categorization key steps are First step to build the web service description vectors. Second step to append relevant ontology concepts and delete irrelevant terms based on the ranking of semantic relationships among the terms. Final step to perform mine web service collection utilizing hierarchical clustering and associate an upper ontology concept for each cluster and the relevant ontology concept for the corresponding subcluster. This leads to better service discovery by matching the service request with an appropriate service description.

Service selection then consists of parameters-based service refinement and semantic similarity-based matching. Better service discovery by matching the service request with an appropriate service description. Increase precision levels, recall levels, and the relevance scores of the retrieved services. Service discovery may involve searching a large number of categories to find appropriate services. The key steps of service selection approach are Retrieve associated parameters forming the association pattern item set.

Prune the association patterns collection. It is difficult to find services that satisfy the desired functionality An efficient matching of the enhanced service request with the retrieved service descriptions is achieved utilizing Latent Semantic Indexing (LSI).semantic-based service selection, we employ ontology linking (semantic web) and LSI thus extending the indexing procedure from solely syntactical information to a semantic level. It is focused only on semantics data rather than heterogeneous web services.

2.3 CONTEXT-AWARE SIMILARITY SEARCH OF WEB SERVICE

Web services environment ambiguity lies a major problem so to avoid that context aware similarity search is deployed to improve the accuracy in similarity metrics. Growing number of services need efficiently locating the desired web services. The similarity metric plays an important role in service search. However, most solutions only exploit information available in the descriptions of services and employ a single similarity metric. The descriptions of services vary in their level of semantic ambiguity. It is hardly to apply one metric upon all services to filtering out those required by users.

Services with more ambiguity, the metric with more restriction should be applied. As to those services sharing the same context and thus less ambiguity, the metric with less restriction is more appropriate. Show information exploitation of context of services to improve the searching accuracy. First employ ideas from the semantic link network to infer the ambiguity and relatedness of the context of services, then utilize two kinds of similarity metrics on services with different contexts.

The goal to estimate the ambiguity and the relatedness of service sets and purpose is to link services with the same context together employ the technique in the semantic link network which is able to build the association relations of services and form the graph of the related service set. Then based on the graph we can estimate the ambiguity and relatedness of the service set. The conservative metric is on those services with more ambiguity, while the relaxed metric is for those with less ambiguity. The experiments show that solution outperforms some searching methods. The similarity measure to resolve the problem caused by the implicit connections of words of services. Better performance than the original algorithms in both the service classification and query.

Proposed framework not having enhanced similarity in terms of precision and recall measures and also lacks accuracy when having changed similarity measures. The similarity search of services, firstly it should decide the context of the request, the context of services that users wish to find. It can depend on users' choice from the already existing contexts or just be decided automatically by the terms provided by users. Here the search policy adopt is the automatic way.

2.4 RANKING AND CLUSTERING WEB SERVICES USING MULTICRITERIA DOMINANCE RELATIONSHIPS

The web is increasingly used not only to find answers to specific information needs but also to carry out various tasks, enhancing the capabilities of current web search engines with effective and efficient techniques for web service retrieval and selection becomes an important issue. Existing service matchmakers typically determine the relevance between a web service advertisement and a service request by computing an overall score that aggregates individual matching scores among the various parameters in their descriptions.

Two main drawbacks characterize such approaches. First there is no single matching criterion that is optimal for determining the similarity between parameters. Second, the reduction of individual scores to an overall similarity leads to significant information loss. Determining appropriate weights for these intermediate scores requires knowledge of user preferences, which is often not possible or easy to acquire. Instead, using a typical aggregation function, such as the average or the minimum of the degrees of match across the service parameters.

Consequently several services those having a single unmatched parameter, may be excluded from the result set, while being potentially good candidate and present two complementary approaches that overcome the aforementioned deficiencies. First methodology for ranking the relevant services for a given request, introducing objective measures based on dominance relationships defined among the services. Second methodology to investigate methods for clustering the relevant services in a way that reveals and reflects the different trade-offs between the matched parameters.

The effectiveness and the efficiency of techniques and algorithms through extensive experimental evaluation on both real requests and relevance sets, as well as on synthetic scenarios. Ranking and clustering of web service based on multiple matching criteria without aggregating the match scores of individual service parameters. It validates the effectiveness and efficiency. It is not focused on the generic query model for supporting any kind of heterogeneous service descriptions rather than focused on ranking and clustering criteria for the data semantics.

3 CONCLUSION

From this paper with perform literature survey on A Generic Query Model for the Heterogeneous Services to Make Unified Discovery.

REFERENCES

[1] X.Dong, A.Halevy, J.Madhavan, E.Nemes, and J.Zhang,"Similarity Search for Web Services," Proc. 13th Int'l Conf. Very Large Data Bases, pp. 372-383. 2004.
[2]J. Farrell and H. Lausen, "Semantic Annotations for WSDL and XML Schema," http://www.w3.org/TR/sawsdl/, Aug. 2007.
[3]Paliwal, A.V. CIMIC, Rutgers Univ., North Brunswick, NJ, USA Shafiq, B. ; Vaidya, J. ; Hui Xiong ;Adam,N. "Semantics-Based Automated Service Discovery" Services Computing, IEEE Transactions on (Volume:5 , Issue: 2) April-June 2012.
[4]Skoutas, D. L3S Res. Center, Hannover, Germany Sacharidis, D. ; Simitsis, A. ; Sellis, T. "Ranking and Clustering Web Services Using Multicriteria Dominance Relationships" Services Computing, IEEE Transactions on (Volume:3 , Issue: 3 : July-Sept. 2010).