Future Generation Computer Systems 48 (2015) 73–81

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Contextual service discovery using term expansion and binding coverage analysis

Shang-Pin Ma [a,*], Ci-Wei Lan [b], Chia-Hsueh Li [a]

[a] Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung 202, Taiwan

[b] Global Business Solution Center, IBM CDL, Taipei 110, Taiwan

HIGHLIGHTS

- We propose a contextual approach to discovering web services.
- The approach mitigates issues of wording bias and ambiguity by term expansion.
- The approach ensures that discovered services are compatible with binding context.
- The approach can find out qualified services for constructing a composite service.

ARTICLE INFO

Article history:
Received 17 March 2014
Received in revised form
21 August 2014
Accepted 19 September 2014
Available online 5 October 2014

Keywords:
Service discovery
Term expansion
Service context

ABSTRACT

Business cloud emerges as a new solution for providing on-demand business services over Internet. Standardized service technologies drive widespread adoption of clouds and enable computation evolution towards service-oriented paradigm. While diversified cloud services are readily available today, how to discover desired services fitting into user's context becomes a practical challenge. In this paper, we propose a Contextual Service Discovery (CSD) approach to help find out qualified services in accordance with binding context on the user side. Query descriptions and binding information are analyzed as a set of meaningful terms. We designed a term expansion mechanism to improve matchmaking performance by mitigating issues of wording bias and ambiguity. Besides, binding coverage analysis between multiple services and a given query is conducted to ensure that matched services are compatible with user's contextual expectation. The experimental results show that the proposed approach is with better performance than other alternatives under Top-*N* precision and recall metrics.

1. Introduction

The well-constructed Internet infrastructure brings new evolution of computing from monolithic machine towards connected servers. Various innovative services as well as business models are emerging, for example, cloud-based computing, storage, database, networking, parallel processing, messaging, monitoring and social analytics [1,2]. Pay-as-you-go pricing makes cloud computing embraced by all levels of organization which can reach the worldwide user base without expensive upfront cost. Both small businesses and large-scale enterprises can benefit from various infrastructures, platforms and software as a service [3]. The trend of business computing is shifting from IT computation outsourcing upward to business-specific service integration [4–6]. How to keep pace of the ever-changing business environment with well-aligned internal and external cloud services becomes a new challenge today.

The service computing paradigm creates unprecedented opportunities for modern business collaborations. Based on standardized interface, technology details are encapsulated in high-level business service descriptions and application integration is easier than before in terms of platform heterogeneities. People are able to focus on business process and work out corresponding executable applications through service discovery as well as composition. Particularly services can be provided by different business partners. In terms of service composition, not only the functionalities but also service's input and output have to be matched against each other. Nevertheless service discovery for a service composition is not a trivial process. In [7–12], they used a text-based method by employing information retrieval techniques to compare keywords

* Corresponding author. Tel.: +886 224622192.
E-mail addresses: shangpin.ma@gmail.com, albert@ntou.edu.tw (S.-P. Ma), lanciwei@tw.ibm.com (C.-W. Lan).
URL: http://soselab.cs.ntou.edu.tw (S.-P. Ma).

http://dx.doi.org/10.1016/j.future.2014.09.013
0167-739X/

against service descriptions. Others used another semantics-based method [13–17] by ontological reasoning to measure matching degree among queries and services. However, effectiveness of service discovery with prior work is limited to service granularity issue in which services may be described similarly but having distinct function scope with varied binding signatures. For example, two services may be named as payment service identically while one may deal with local currency purchase and the other provides credit card based international transaction processing. Those two services definitely have different requirements of input parameter and produce distinct output results. In this paper, contextual service discovery refers to the process of identifying candidate services based on literally service descriptions and programming specifications of service's input and output. Besides, binding context is defined as the set of a service's input and output programming specifications. Contextual service discovery is a practical challenge that has to deal with linguistic service description matchmaking and binding context subsumption judgment at the same time.

In this paper, we proposed a Contextual Service Discovery (CSD) approach using term expansion and binding context analysis techniques. We aim at solving service discovery challenge especially for running a service composition wherein service's input and output specifications need to be considered. To mitigate linguistic imprecision between a service description and a user query, we adopt a thesaurus to make wordings consistently and to expand terms of user query as well. Furthermore domain ontologies are employed to perform semantic expansion so that discovery exploration will be more extensive. In addition to service descriptions, the notion of binding context represented by I/O coverage is considered, i.e. input and output parameters in service binding specification. The experiments show that proposed CSD approach reaches better precision and recall results in comparison with solely semantic expansion method and other non-contextual benchmarks. Major contributions of proposed approach include: (a) performing a novel contextual service discovery rather than simply text matching: performing service matchmaking with not only literal service description comparisons but also service's binding specification evaluations. The aforementioned challenge is practically significant while doing service discovery for a service composition. Discovering a service within composition scenarios needs to consider the binding compatibility of a set of services rather than a singular service; and (b) providing a flexible matchmaking scheme in which a user is free to conduct service discovery with fully or partially expanded query terms. By configuring parameterized settings, a user is able to select fully semantic expansion when a specific ontology for application scenario is available and to moderate term expansion otherwise for example.

The remainder of this paper is organized as follows: Section 2 reviews and analyzes prior studies of Web Service discovery. Section 3 addresses details of the proposed CSD approach. Section 4 presents empirical experiments. Conclusions as well as future work are presented in the final section.

2. Related work

In this section, we investigate a number of studies related to Web Service discovery and illustrate the features of our approach.

Text-based service discovery is one of most popular matchmaking architecture where information retrieval techniques are employed to compare service descriptions against a user's query. Stroulia and Wang [18] used well-known TF/IDF (term frequency/inverse document frequency) [8] and WordNet [19] ontology to calculate similarities between two WSDLs. They used the structure of data types, operations and content semantics therein for service discovery. Dong et al. learned from hypertext search and proposed the Woogle [7] search engine. Similarity comparison is measured by a TF/IDF-enhanced clustering algorithm of terms in query and service. Hao et al. [20] devised an IR (Information Retrieval) based mechanism for discovering and ranking Web Services. To achieve better retrieval performance, authors proposed the notion of preference degree for Web Services by calculating two indicators: service relevance and service importance. The service importance is based on the proposed schema tree matching algorithm to catch both structure and semantic similarity. Plebani et al. [10] presented an evaluation method called URBE (UDDI Registry By Example). URBE analyzes structures of WSDL documents and analyzes terms used in WSDL to measure the similarity between multiple Web Service interfaces. URBE also utilizes WordNet and domain ontologies to realize a semantic-oriented variant for enhancing precision.

On the other hand, people also tried to use domain ontology for service description, search, and composition semantically. Verma et al. [13,16] used a semantic annotation method (SAWSDL) to incorporate semantics in WSDL for improvements of service search accuracy. Martin et al. [17] utilized OWL-S [21], the successor to DAML-S (DAML-based Web Service ontology), to describe the Web Service capabilities semantically and store semantic information in UDDI. [22,15,14] proposed methods to organize queries and Web Services using WSMO (Web Services Modeling Ontology). They treated predefined goals as the templates of queries, and use predefined service profiles to describe Web Services.

Currently, context-aware service discovery attracts more and more attention. As stated in [23], the concept of context is too broad so that context could be anything related to the user and applications. Rong and Liu [23] proposed that the context awareness is one of solutions for bridging the gap between the issued service request and the real use requirement. Authors also classified context-aware Web Service matchmaking methods into four categories: personal profile oriented context, usage history oriented context, process oriented context, and other contexts. Spanoudakis et al. [24] presented a platform to perform service discovery based on structural and behavioral service models as well as context. In this approach, context information is obtained through calling "context operations". In [25], Medjahed and Atif proposed a matching framework for Web Service composition. Authors developed an ontology-based categorization of contextual information in the Web Service domain, defined a two-level mechanism for modeling Web Service contexts by RDF (Resource Description Framework) and WS-Policy, and devised a context-matching mechanism based on the proposed "context community" that can group context policies related to the same category. Since the context information is any additional information that can be used to improve the behavior of a service in a situation [26], we focus on the binding context that can improve the search accuracy when binding component services in a composite service [27] or a service-oriented application.

In order to illustrate the features of our contextual service discovery approach, we compare and analyze the above-mentioned related service matching approaches along with four dimensions (see Table 1): the external representation of service interfaces, the internal representation of service interfaces, the representation of context information, and service matching methods.

In most of the text-based approaches, the external representation of service interface is the standard WSDL. In ontology-based approaches, the service interface follows ontology-enabled formats, such as OWL-S, SAWSDL (Semantic Annotations for WSDL) or WSMO (Web Service Modeling Ontology) to express its semantics. The service providers usually need to spend considerable effort to develop ontology-based service descriptions. For most of the current context-aware approaches, the representation of service interface is not the main focus. Our proposed approach mainly