2009 International Conference on Machine Learning and Computing
IPCSIT vol.3 (2011) © (2011) IACSIT Press, Singapore

A Unified CBR Approach for Web Services Discovery, Composition and Recommendation

Zhaohao Sun [1 +], Jun Han [2], Dianfu Ma [2]

[1] Graduate School of Information Technology and Mathematical Sciences
University of Ballarat, Mt Helen 3353 VIC Australia, z.sun@ballarat.edu.au

[2] School of Computer Science and Engineering, Beihang University, Beijing, 100191 China
jun_han@buaa.edu.cn, dfma@buaa.edu.cn

Abstract. This article proposes a system architecture of a case based web services reasoner, CWSR, looks at the correspondence relationship between the main activities of CBR and those of web services, and proposes a unified approach for case-based web service discovery, composition and recommendation. The proposed approach will facilitate the research and development of web services, e-services, intelligent systems and business intelligence.

Keywords: Web services, case-based reasoning (CBR), web services composition, business intelligence.

1. Introduction

Web services are playing an increasingly important role in e-commerce, business intelligence, and service computing [18]. Web services are the provision of services over electronic networks such as the Internet and wireless networks [11]. The key motive for rapid development of web services is the ability to discover services that fulfil users' demands, negotiate service contracts and have the services delivered where and when the users request them [17]. With dramatic development of the Internet and the web in the past decade, web services have been flourishing in e-commerce, business intelligence (BI) because they offer a number of strategic advantages such as mobility, flexibility, interactivity and interchangeability in comparison with traditional services [3]. Current research trend is to add intelligent techniques to web services to facilitate discovery, invocation, composition, and recommendation of web services [18].

Case based reasoning (CBR) has found many successful applications in business intelligence and multiagent web services [15], especially in matchmaking, discovery, brokering and composition of web services [8]. For example, Ladner et al [6] use a case-based classifier for web services discovery. However, there is no unified treatment for web services using CBR taking into account web services lifecycle although there have been a great number of researches on web service discovery and composition [20]. This article will alleviate the above mentioned issue by proposing a unified approach for case-based web service discovery, composition and recommendation. To this end, the remainder of this article is organized as follows: Section 2 reviews web services and some of its main activities. Section 3 proposes CWSR: a case-based web service reasoner. Section 4 examines the correspondence relationship between web services and CBR. Section 5 provides a unified treatment for case-based web service discovery, composition and recommendation. The final section ends the article with some concluding remarks and future work.

2. Web Services

[+] Corresponding author.
E-mail address: z.sun@ballarat.edu.au

Web services are defined from an e-commerce viewpoint at one extreme, in this case, web services are the same as e-services. At another extreme, web services are defined from a computer science viewpoint. For example, web services are defined as the network enabled reusable components that conform to an interface with standard description format and access protocols [21]. Between these two extremes, many different definitions have been proposed by different authors. For example, web services are self-contained, modular applications, accessible via the web, that provide a set of functionalities to businesses or individuals [17]. Further, there are also different levels for defining web services from a methodological viewpoint. For example, a web service is a way of publishing an explicit, machine-readable, common standard description of how to use a service and access it via another program using some standard message transports [14]. Others are at an intermediary level, for example, a web service is an operation typically addressed via a URI, declaratively described using widely accepted standards, and accessed via platform-independent XML-based messages [1]. A more technical definition of web services is as follows: A web service [1] is "a standardized way of integrating web-based applications using the XML, SOAP, WSDL, and UDDI open standards over an Internet protocol backbone. XML is used to tag the data. SOAP is used to transfer the data. WSDL is used for describing the e-service available. UDDI is used for listing what services are available." This article considers web services as simple, self contained applications that perform functions, from simple requests to complicated business processes.

Many activities of web services have drawn an increasing attention. For example, web service search, matching, retrieval, discovery, adaptation, reuse, consultation, personalization, composition and recommendation, negotiation, contracting and billing, etc. [12]. In what follows, we only review web service discovery, composition and recommendation in some detail.

Web service discovery is a process of finding most appropriate web services needed by a web service requester [12]. It identifies a new web service and detects an update to a previously discovered web service [6]. There have been a variety of techniques and approaches developed for web service discovery. For example, OWL-S (of W3C) provides classes that describe what the service does, how to ask for the service, what happens when the service is carried out, and how the service can be accessed [6].

Web service composition primarily concerns requests of web service users that cannot be satisfied by any available web services [9]. It combines a set of available web services to obtain a composite service that might be recommended to the users. Therefore, web service composition refers to the process of creating customised services from existing services by a process of dynamic discovery, integration and execution of those services in a deliberate order to satisfy user requirements [7]. Service composition can be either performed by composing elementary or composite services. Web service composition is an important topic for service computing, because composing web services to meet the requirement of the web service requester is the most important issue for web service providers and brokers.

Web service recommendation is to help web service requesters in selecting web services more suitable to their needs [8]. Web service recommendation is a significant challenge for web services, in particular for web service brokers. Web service recommendation can be improved through optimization, analysis, forecasting, reasoning and simulation [5]. Sun and Lau has examined case based web service recommendation [14]. However, how to integrate web service recommendation, composition and discovery in a unified way is still a big issue for web services. This article will address this issue in Section 5 from a CBR perspective.

3. CWSR: A Case Based Web Services Reasoner

Case-based reasoning (CBR) is a reasoning paradigm based on previous experiences or cases; that is, a CBR system solves new problems by adapting solutions that were used to successfully solve old problems [6, 13]. Ladner et al [6] use case-based classification for web service discovery by applying CBR to supervised classification tasks. Kwon [5] examines how to find the most similar web service case among cases using CBR. Limthanmaphon and Zhang [7] examine web service composition using CBR and present a model for web service composition. CBR has been successful in making recommendation of business activities such as in e-commerce to recommend different e-services with high QoS [14-16, 18]. However, how to discover, compose and recommend web services in a unified way remains open for CBR research. To address this issue, we first propose a case based web services reasoner (CWSR). The system architecture of the CWSR mainly consists of an interface agent, a global web service base (GWB) and a CBR inference engine (CBRIE), as shown in Figure 1. The interface agent consists of some kinds of natural language processing

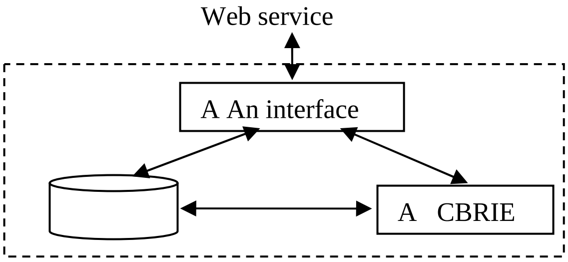

Fig. 1: A system architecture of the CWSR.

systems that allow the user to interact with the web service strategies [14]. The GWB consists of all the web service cases that the system collects periodically and the new web services discovered when the system is running. The CBRIE consists of the mechanism for manipulating the GWB to infer web services X based on CBR (X denotes discovery, composition, recommendation, etc) requested by the user. In order to implement this system architecture, we discuss why CBR can be applied in web services in the next section.

4. Web Services vs Case Based Reasoning

A case in CBR can be denoted as c = (p, q), where p is the structured problem description and q is the solution description [13]. In web services, a service case base stores the collection of service cases [7]. A service case, w = (d, s), consists of the service description d and its service solution (or functions) s as well as other information including functional dependency among web services [5]. The service description corresponds to the requirement of the service user, while the service solution corresponds to the answer to the requirement. In this way, a web service case in web services corresponds to a case in CBR.

When service definitions change or new providers and services are registered within the web services platform such as CWSR, the services need to be adaptive to the change in the environment with minimal user intervention [2]. In other words, web service adaptation is necessary for web services. In fact, case retrieval (search), reuse, revise (adaptation) and retention constitute the basic activities of CBR [13, 15]. Web service retrieval (search), reuse, adaptation, and retention in web services can then correspond to the activities of CBR. Therefore, CBR can be used for processing web service retrieval, reuse, adaptation, and retention. This implies that CBR is naturally applicable to web services. This is why CBR has been successfully applied to web service discovery, search and matching [6, 7].

5. A Unified Treatment of Case Based Web Services

The web service user's demand is normalized into a structured service description p'. Then the CWSR uses its similarity metric mechanism to retrieve its service case base GWB, which consists of service cases, each of these is denoted as $c = (p, q)$, where p is the structured service description and q is the service solution description. The inference engine of the CWSR performs similarity-based reasoning that can be formalized as [13]:

$$\frac{P', P' \approx P, P \to Q, Q \approx Q'}{\therefore Q'} \tag{1}$$

where P, P', Q , and Q' represent fuzzy compound propositions, $P' \approx P$ means that P and P' are similar. Q and Q' are also similar.

From a fuzzy CBR viewpoint, the service case retrieval process from web service search and matching is used to discover the following service cases from the web service case base GWB in the CWSR [13, 15]:

$$C(p') = \{c \mid c = (p, q), p \approx p'\} = \{c_1, c_2, ..., c_n\} \tag{2}$$

This is the result of case based web service discovery, where n is a positive integer, c_i, i = 1, 2, …, n are all service cases with their demand description p similar to the current demand description p'. Usually, $C(p') = \{c_1, c_2, ..., c_n\}$ satisfies the following property: for any integer i, $1 \le i < n$ and $c_i = (p_i, q_i)$,

$$s(p_i, p') \ge s(p_{i+1}, p') \quad (3)$$

where s(.) is a similarity metric, which measures the similarity between one service demand and another.

If n is small, the CWSR will directly recommend the web service solutions of $\{c_1, c_2, ..., c_n\}$, $\{q_1, q_2, ..., q_n\}$, to the web service requester through the interface agent. If n is very large, the CWSR will recommend the web service solutions of the first m cases of $\{c_1, c_2, ..., c_n\}$; that is, $\{q_1, q_2, ..., q_m\}$, to the requester, where $1 \le m < n$. This process is case-based web service recommendation [14].

After obtaining the recommended web services from the CWSR, the web service requester will evaluate them and select one of the following:

1. Accept one of the recommended web services, q_k, and contract it, where $1 \le k \le m$.
2. Adjust her/his demand descriptions p' and then send them to the CWSR.
3. Reject the recommended web services and leave the CWSR.

It is obvious that only the first two of these three choices require further discussion. For the first choice, the deal was successfully done and the CWSR routinely updates the service case $c_k = (p_k, q_k)$ in the GWB. At the same time, the CWSR has reused the service case successfully; that is, CWSR completes the process of case-based web service use and reuse. For the second choice, the demand adjustment is the process of demand adaptation that corresponds to problem adaptation [13]. After having adjusted the demand, the requester submits it to the CWSR, which will conduct web service retrieval, recommendation and reuse again. This process is case-based web service adaptation.

Further, if the web service adaptation is unsuccessful, the CWSR has to conduct case based web service composition. Assume that the web service requester's demand is normalized into a structured service description and service solution description $c = (p', q')$, and the CWSR has discovered m web services $\{c_1, c_2, ..., c_m\}$ (where m is the least positive number) such that

$$p' \subseteq p_1 \cup p_2 \cup ... \cup p_m \text{ and } q' \subseteq q_1 \cup q_2 \cup ... \cup q_m \quad (4)$$

where $\cup$ is the union operation of the set theory. This is a necessary condition for case based web service composition. Based on (4), the composite web service case $c = (p, q)$ is obtained through case based web service composition of the CWSR:

$$p = p_1 \oplus p_2 \oplus ... \oplus p_m \text{ and } q = q_1 \otimes q_2 \otimes ... \otimes q_m \quad (5)$$

where $\oplus$ and $\otimes$ are composition operations for web services. For example, when they are replaced by the ordinary (or fuzzy) union operation of set theory, the composite web service is the same as that discussed in DIANE [4] or similar to the composite web service in [5]. When they are replaced by the "independence" operation taking into account interdependent relationships among the services, the composite service is similar to that discussed in [7]. However, it is still a big issue for case based web service composition to use a more complicated composition operation to obtain a composite service case.

After obtaining a composite service case, the CWSR will recommend it to the service requester for acceptance. This goes to the early mentioned process for acceptance, adaptation or rejection.

So far, we have discussed case-based web service retrieval, discovery, adaptation, reuse, composition and recommendation in a unified way. We will not illustrate the above models any more owing to space limitation.

6. Conclusions and Future Work

The article proposed a system architecture of a case based web services reasoner, CWSR, discussed the correspondence relationship between the main activities of CBR and those of web services, and explored a unified treatment for case-based web service discovery, composition and recommendation. The proposed approach will facilitate the development of web services, intelligent systems and business intelligence.

Applying intelligent techniques to web services is still a new topic for business intelligence, AI and web services. In future work, we will develop a system prototype based on the proposed CWSR. We will also integrate web service discovery, composition and recommendation using the CWSR based on soft CBR.

7. References

[1] G. Alonso, F. Casati, H. Kuno, and Machiraju V. Web Services: Concepts, Architectures and Applications. *Berlin: Springer,* 2004.

[2] S. Dustdar, and W. Schreiner. A survey on web services composition. *Intl J. Web and Grid Services*. 2005 1 (1): 1-30.

[3] KD. Hoffman. Marketing + MIS = E-Services. *Comm. of the ACM*, 2003 46(6): 53-55.

[4] U. Küster, B. König-Ries, M. Stern, and M. Klein. DIANE: An integrated approach to automated service discovery, matchmaking and composition. *In: Intl Conf. on World Wide Web (WWW2007)*, 2007, pp.1033-1041.

[5] OB. Kwon. Meta web service: building web-based open decision support system based on web services. *Expert Systems with Applications* 2003, 24: 375–389.

[6] R. Ladner, et al. Soft computing techniques for web service brokering, *Soft Computing*. 2008, 12: 1089-1098.

[7] B. Limthanmaphon, and Y. Zhang. Web service composition with case-based reasoning. *In: Proc. 14th Australasian Database Conf., Adelaide, Australia*, 2003, pp. 201-208.

[8] F. Lorenzi, and F. Ricci. Case-Based Recommender Systems: A Unifying View. In: B. Mobasher and S.S. Anand (Eds.): ITWP 2003, LNAI 3169, *Berlin Heidelberg: Springer*, 2005: pp. 89–113.

[9] NC. Narendra, and B. Orriens. Requirements-driven modeling of the web service execution and adaptation lifecycle. *In: S. Madria, et al (Eds): ICDCIT, LNCS* 4317. 2006: pp. 314-324.

[10] MP. Papazoglou, and D. Georgakopoulos. Service-oriented computing, Comm. of The ACM, 2003, 46(10): 25-28.

[11] RT. Rust, and PK. Kannan. E-service: A new paradigm for business in the electronic environment. *Comm. of the ACM*, 2003 46(6): 37-42.

[12] MP. Singh, and MN. Huhns. Service-oriented computing: Semantics, Processes, and Agents. *Chichester: Wiley & Sons*, 2005.

[13] Z. Sun, and G. Finnie G. A unified logical model for CBR-based e-commerce systems. *Intl J. Intell. Syst.* 2005, 20(1): 29-46.

[14] Z. Sun, and SK. Lau. Customer Experience Management in E-services. In: J. Lu, D. Ruan, and G. Zhang (eds) E-Service Intelligence: Methodologies, *Technologies and applications, Springer*, 2007: pp. 365-388.

[15] Z. Sun, and G. Finnie. Intelligent Techniques in E-Commerce: A Case-based Reasoning Perspective. *Berlin Heidelberg: Springer,* 2004.

[16] Z. Sun, and G. Finnie. A unified logical model for CBR-based e-commerce systems. *Intl J. Intell. Syst.* 2005, 20(1): 29-26.

[17] X. Tang, C. Jiang, Z. Ding, and C. Wang. A Petri net-based semantic web services automatic composition method. *J. Software (in Chinese)*, 2007, 18(12): 2991-3000.

[18] H. Wang, and YQ. Zhang, R. Sunderraman. Extensible soft semantic web services agent. *Soft Compting 2006*, 10: 1021-1029.

[19] M. Wang, WK. Cheung, J. Liu, X. Xie, and Z. Lou. E-service/process composition through multi-agent constraint management. In: BPM 2006, Vienna, LNCS 4102, *Springer, 2006*, pp. 274-289.

[20] Y. Yao, F. Yang, and S. Su. Flexible decision making in web services negotiation. *In: J. Euzenat and J. Domingue (Eds.): AIMSA 2006,* LNAI 4183, 2006, pp.108-117.

[21] LJ. Zhang, and M. Jeckle. The next big thing: Web services composition. *In: M. Jeckle, and LJ. Zhang (eds): ICWS-Europe, LNCS 2853, Springer, 2003, pp. 1-10.*