Efficient Measurement of Service Similarity

Joshua Church, Department of Computer Science, Volgenau School of Engineering, George Mason University, Fairfax, VA, USA
Amihai Motro, Department of Computer Science, Volgenau School of Engineering, George Mason University, Fairfax, VA, USA

ABSTRACT

The authors define a formal model for information services that incorporates the concept of service similarity. The model places services in metric spaces, and allows for services that have arbitrarily complex inputs and output domains. The authors then address the challenge of service substitution: finding the services most similar to a given service among a group, possibly large, of candidate services. To solve this nearest neighbor problem efficiently the authors embed the space of services into a vector space and search for the nearest neighbors in the target space. The authors report on an extensive experiment that validates both their formalization of similarity and their methods for finding service substitutions.

Keywords: Information Services, Service Similarity, Service Substitution, Side-Effect-Free Service, Stateless Service

INTRODUCTION

The increasing deployment of the service-oriented programming paradigm, in which programs are composed by weaving together distributed, platform-independent software components, raises the problem of service substitution: How to best replace one software component with another. The substitution may be motivated by a variety of reasons: An existing component might have failed, or there could be an alternative service that is of higher quality or lower cost. As the service-oriented paradigm gains popularity, the number of available services increases substantially, making the searches for substitutions more complicated and costly, and raising the need for methods and tools that facilitate such searches (Sillito, Murphy et al. 2008).

Finding the service (or group of services) that are most similar to a given service can be stated as a k-nearest neighbors problem. As such, it requires establishing a formal notion of service similarity. In this paper we describe a formal model for information services that incorporates the concept of service similarity, and we utilize this model for addressing the challenge of finding service substitutions.

DOI: 10.4018/jwsr.2013040102
Our model has several distinguishing aspects. First, the model considers services as *mappings* of inputs to outputs. A simple example is a service that returns the current temperature for a given US Zip code. These services are stateless and free of side-effects. Our view of services is thus quite similar to that advocated by the Representative State Transfer (REST) architectural style (Zhao and Doshi 2009). Services designed according to RESTful principles make resources available to clients through Web pages: Clients navigate to a URL, provide the necessary input values, invoke the service and receive a response. Since REST describes functionality with simple URLs rather than operations enveloped in complex XML standards, RESTful service implementations are becoming a popular alternative to SOAP/WSDL technologies. Unfortunately, formal models and techniques that prove so useful for developing services with SOAP/WSDL standards are absent for this type of service. So, as more data is made available via RESTful services, a formal model for information services is expected to be useful for RESTful services.

An important feature of the model is that the input and output of a service could be *arbitrarily complex*. For example, a service that returns the value of a stock portfolio given a particular date and a set of stock symbols and their corresponding quantities — has input which is a pair comprising a single value (date) and an arbitrarily-sized set of pairs (stock symbol and quantity). Another feature of the model is that the *semantics* of services need not be represented externally — by means of natural language descriptions, sets of keywords, ontologies, and so on. Instead, when necessary, semantics are inferred from observable behavior.

Our model defines a notion of service *similarity* that is based solely on observable behavior. Similarity is measured with distance *metrics*. Each basic domain (such as real numbers or character strings) is associated with a simple metric, and these metrics are then combined to create complex metrics that could be associated with domains of arbitrary complexity. Given two services with identically structured inputs and outputs, their similarity is derived from the similarity of their outputs for identical inputs. Additionally, the model is extended to deal with service exceptions: instances in which services might not provide valid outputs as expected.

A naïve solution to the problem of finding substitutions is to compare the observable behavior of the given service to each of the candidate services. To avoid long computations at the time when the substitutions are needed, similarity measurements should be prepared ahead of time. The cost, however, could be high, since metrics could be complex, and the number of services could be high. To overcome this challenge we describe a method that *embeds* the given set of services into a vector space, where the complexity of measurement is reduced significantly.

Two aspects of this work require experimental validation. The first is the *quality* of the discoveries. It must be demonstrated that our metrics indeed reflect the similarity inherent among services. That is, that services that behave similarly are indeed measured to have strong similarity. Our experiments show the precision of discoveries to be between 88% and 97%. The second aspect is the *performance* of our nearest neighbor methods. Our experiments show that the metric space embedding method reduces search cost by as much as 98%. This paper expands on an earlier version (Church and Motro 2013).

Our model is described in two sections: the third section defines services and the section after defines service similarity. Our approach to the challenge of service substitutions is described in the fifth section. The following section details and analyzes the experiments that validate our methods. The next section extends the work by relaxing key assumptions in the model, thereby allowing for a wider range of substitutions. Finally, the last section summarizes the results and sketches future research directions. We begin in this next section with a brief survey of related work.
Related Content

A Method for Predicting Wikipedia Editors' Editing Interest: Based on a Factor Graph Model
Haisu Zhang, Sheng Zhang, Zhaolin Wu, Liwei Huang and Yutao Ma (2016).
*International Journal of Web Services Research* (pp. 1-25).
[www.igi-global.com/article/a-method-for-predicting-wikipedia-editors-editing-interest/161800?camid=4v1a](www.igi-global.com/article/a-method-for-predicting-wikipedia-editors-editing-interest/161800?camid=4v1a)

Development of Distance Measures for Process Mining, Discovery and Integration
Joonsoo Bae, Ling Liu, James Caverlee, Liang-Jie Zhang and Hyerim Bae (2007).
*International Journal of Web Services Research* (pp. 1-17).
[www.igi-global.com/article/development-distance-measures-process-mining/3107?camid=4v1a](www.igi-global.com/article/development-distance-measures-process-mining/3107?camid=4v1a)
QoS-Aware and Federated Enhancement for UDDI
[www.igi-global.com/article/qos-aware-federated-enhancement-uddi/3041?camid=4v1a](www.igi-global.com/article/qos-aware-federated-enhancement-uddi/3041?camid=4v1a)

NAM: A Network Adaptable Middleware to Enhance Response Time of Web Services
[www.igi-global.com/chapter/nam-network-adaptable-middleware-enhance/31209?camid=4v1a](www.igi-global.com/chapter/nam-network-adaptable-middleware-enhance/31209?camid=4v1a)