Discovering Semantic Relations between Web Services Using Their Pre and Post-Conditions

Lin Lin & I. Budak Arpinar
LSDIS (Large Scale Distributed Information Systems) Lab
Department of Computer Science, University of Georgia
Athens GA 30602-7404, {lin, budak}@cs.uga.edu

Abstract

Discovering and assembling individual Web services into more complex yet new and more useful Web processes receive significant attention from the academia recently. In this paper, we explore using pre and post-conditions of Web services to enable their automatic composition. Also, we present a novel technique for discovering semantic relations between pre and post-conditions of different services using their ontological descriptions. This enables determining services with complimentary functions and generating a semantic Web of services. Our technique takes semantic similarity of pre and post-conditions into account and builds on our earlier work on discovering semantic relationships between interfaces (input and output) of Web services.

1. Introduction

Semantic Web Services (WSs) extend the semantic Web from a distributed source of information to a distributed source of services. Developing efficient automatic discovery and composition techniques is among the most important challenges for the success of semantic WSs. To contribute towards this goal, we developed an Interface-Matching Automatic (IMA) composition technique earlier [1, 2]. However, without functionality constraints, IMA technique is more appropriate for the services that always return relatively simple results based on the user-supplied inputs [4]. To overcome this problem, we propose a discovery technique based on pre and post-conditions of WSs. In this paper, we present a new technique to identify possible relations between pairs of WSs by checking semantic similarities between their pre and post-conditions.

The rest of the paper is organized as follows: Section 2 presents the technique for discovering semantic relations between pre and post-conditions of WSs. Section 3 provides the conclusion and future work.

2. Relations between Pre and Post-conditions

Given a set of WSs, the semantic relations between pre and post-conditions of these services need to be established, and then a semantic network of services with complimentary functions can be constructed according to these relations. Therefore, the subsequent step of services composition according to a specific task at hand can be viewed as a path traversal problem.

2.1. Relationships among Web Services

We define two WSs to have a relationship as either these two WSs can be somehow plugged together to perform a value added service or one can be substituted by the other. Let S_m and S_n be two WSs, and a relationship R between S_m and S_n can be identified as follows:

R1: Prerequisite Relationship ($\rightarrow$): $S_m \rightarrow S_n$
It shows that S_m has to finish before S_n starts. For example, the flight booking service has to be done before the payment service.

R2: Parallel Relationship (//): S_m // S_n
Here S_m and S_n can execute in parallel but the results of each need to be combined to enable further execution.

R3: Substitute Relationship ($\Leftrightarrow$): $S_m \Leftrightarrow S_n$
Here S_m can be substituted by S_n. The S_m and S_n seem to provide the same functionality but they have some different attributes. For example, in the case of delivery service, S_m can be an air courier delivery service while S_n is a ground delivery service.

R4: Include Relationship ($\ni$): $S_m \ni S_n$
It means that S_m provides services that include the services offered by S_n. For example, S_m can be an express delivery service that offers both ground and air delivery while S_n is a ground delivery service.

2.2. Semantic Relations between Conditions

Current OWL-S specification provides no existing standards for representing pre and post-conditions. Therefore, we model pre and post-conditions as two sets of RDF triples from an ontology. The use of RDF triples provides a way that is simple yet rich enough to express pre and post-conditions semantically.

Each set of RDF triples for a pre or post-condition is simply called a condition. Thus, a WS can be represented as pre-condition→post-condition. So S_m and S_n can be represented as $Cond_{m1} \rightarrow Cond_{m2}$ and $Cond_{n1} \rightarrow Cond_{n2}$ respectively. Several semantic relations between conditions can be identified as follows:

Exact Match: $Cond_{m2} \rightarrow Cond_{n1}$
Cond$_{m2}$ exactly matches to condition Cond$_{n1}$. S_m and S_n have the prerequisite relationship.

Plug-in (PI) Match: $Cond_{m2}\ \boxed{PI}\rightarrow Cond_{n1}$
Cond$_{m2}$ is semantically stricter than Cond$_{n1}$. In this case, S_m and S_n have the prerequisite relationship. For example, Cond$_{m2}$ can specify the availability of payment by MasterCard only, and Cond$_{n1}$ can allow the availability of payment by all major credit cards.

Plus (+) Match: $Cond_{m2}\ \boxed{+}\rightarrow Cond_{n1}$
Cond$_{m2}$ only partially satisfies Cond$_{n1}$ so that some other condition(s) are needed together with Cond$_{m2}$ to completely satisfy Cond$_{n1}$. We say that Cond$_{m2}$ can be a plus match to Cond$_{n1}$. S_m has the parallel relationship (i.e., //) with some other services since their results together enable the execution of S_n.

Complimentary (CP) Match: $Cond_{m2}\ \boxed{CP}\rightarrow Cond_{n1}$
Cond$_{m2}$ compliments Cond$_{n1}$. For example, Cond$_{m2}$ can specify that book is available to sell, and Cond$_{n1}$ can specify that book is available to buy. Buying denotes an action that is compatible to selling. In this case, S_m and S_n also have the prerequisite relationship (i.e., →).

The degree of similarity between two conditions is assessed through comparing similarity between triples of these two conditions. We have developed a Semantic Relations Discovery System prototype (see Figure 1). Currently, we are running our experiments using TAP [3], which is an experimental knowledge base (i.e., populated ontology) about people, places, products, etc.

4. Conclusions and Future Work

In this paper, we propose a new technique for discovering semantic relations between pre and post-conditions of different services using their ontological descriptions. We model pre and post-conditions as sets of RDF triples, which allow us to express conditions in a simple but flexible way. We also address the issue of relaxed matching in the sense that the pre-condition of one service can be satisfied by the post-conditions of more than one service. A prototype of proposed approach is implemented so that a semantic Web of services can be generated and displayed visually. Our future work includes conducting experiments to measure performance of our system.

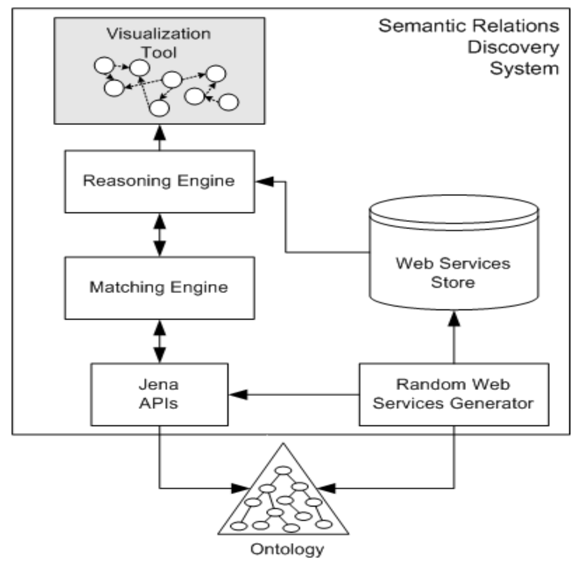

Figure 1. An overview of system architecture

References

[1] B. Arpinar, R. Zhang, B. Aleman-Meza, and A. Maduko., "Ontology-driven Web Services Composition Platform", IEEE Intl. Conf. on e-Commerce Technology, San Diego, California, July 6-9, 2004.

[2] B. Arpinar, R. Zhang, B. Aleman-Meza, and A. Maduko, "Ontology-driven Web Services Composition Platform", Journal of Information Systems and e-Business Management, to appear.

[3] TAP knowledge base, URL: http://tap.stanford.edu/tap/tapkb.html

[4] R. Zhang, B. Arpinar, and B. Aleman-Meza, "Automatic Composition of Semantic Web Services", International Conference on Web Services, Las Vegas, NV, June 2003.