Abstract

In order to combine data from various heterogeneous sources, software agents must first understand the semantics of the sources, expressed in the source model. Currently, source modeling is manual, but as large numbers of sources come online, it is impractical to expect users to continue modeling them by hand. We describe two machine learning techniques for automatically modeling information sources: one that uses source's metadata, contained in a Web Service Definition file, and one that uses the source's content, to classify the semantics of the data it uses. We go beyond previous works and verify predictions by invoking the source with sample data of the predicted type. We provide performance results of both methods and validate our approach on several live Web sources. In addition, we describe the application of semantic modeling within the CALO project.

Keywords: data integration; data modeling; data semantics; semantic data model

Introduction

Software agents will soon assist users in the office environment by carrying out complex everyday tasks, such as planning a trip to a meeting. To complete their task successfully, these agents will need to combine information from a variety of heterogeneous sources. Take, for example, the task of arranging a trip. A user has to follow a number of steps in a sequence, including:

- get the date and location of the meeting,
- find a convenient flight on those dates to that location,
- get the weather forecast,
- find all hotels in the city of the meeting with the required amenities (e.g., high speed Internet),
- keep hotels within 3 miles of the meeting site,
- keep hotels with rates within government-allowed per diem,
- book hotel from the list with the best reviews,
- rent a car, and so on.

Some of the information sources a user interacts with in the course of carrying out this task are shown in Figure 1. In order to make use of these sources—whether they are
Web services, HTML pages or databases—an intelligent agent needs to know the schema of the source. Currently, the user must manually model the source: specify the type of data it accepts and returns, and the functionality it provides. This is what is done by information integration systems (Thakkar, Ambite, & Knoblock, 2005) that, similarly to our envisioned intelligent office assistants, provide uniform access to heterogeneous online information sources in order to answer a specific user query or be composed with other sources as a new information service. However, as new sources come online or are discovered, it is infeasible to ask the user to explicitly model them. While various technologies, most notably the Semantic Web, have been proposed to enable programmatic access to new sources, they are slow to be adopted, and at best will offer only a partial solution, because information providers will not always agree on a common schema. Rather than rely on standards to which providers may or may not conform, we instead propose to automatically model a new source: that is, *discover the semantics of its inputs and outputs, and the functionality it provides.*

We divide source modeling tasks into two subproblems: recognizing the semantics of the data it uses (*semantic labeling*), and inducing the logical definition of the source (*learning functionality*) (Carman & Knoblock, 2007). This article addresses the first problem, which is similar to the schema matching problem encountered by the database community, where the schema of one database has to be linked with the schema of another before the first database can use the data provided by the second. Unlike schema matching, where the content of both databases is available to the matching algorithm, we must first figure out how to invoke the source whose input and output parameters we are trying to label. We will show in the article that we can use the source’s metadata, if available, to classify the input parameters used by the source and assign them to predefined semantic types. If the metadata is not easily available, but instead, a user has provided a few examples of the inputs by querying the source, we show that we can classify the data types used by the source using their content only. While others have used metadata to semantically classify the source’s data types (Hess & Kushmerick, 2003), we go further and verify a classifier’s

www.igi-global.com/e-resources/library-recommendation/?id=2

Related Content

Towards Web 2.0 Applications: A Conceptual Model for Rich Internet Applications
Alessandro Bozzon, Sara Comai, Piero Fraternali and Giovanni Toffetti Carughi (2010). Handbook of Research on Web 2.0, 3.0, and X.0: Technologies, Business, and Social Applications (pp. 75-95).
www.igi-global.com/chapter/towards-web-applications/39165?camid=4v1a

Company Acquisition Relations Extraction From Web Pages
www.igi-global.com/chapter/company-acquisition-relations-extraction-from-web-pages/215059?camid=4v1a

The Berlin SPARQL Benchmark
www.igi-global.com/chapter/berlin-sparql-benchmark/55042?camid=4v1a
R-DEVICE: An Object-Oriented Knowledge Base for RDF Metadata
www.igi-global.com/article/device-object-oriented-knowledge-base/2819?camid=4v1a