A Practical Agent-Based Method to Extract Semantic Information from the Web*

J. L. Arjona, R. Corchuelo, A. Ruiz, and M. Toro

Escuela Técnica Superior de Ingeniería Informática de la Universidad de Sevilla
Departamento de Lenguajes y Sistemas Informáticos
Avda. de la Reina Mercedes, s/n, Sevilla, Spain
{arjona,corchu,aruiz,mtoro}@lsi.us.es

Abstract. The semantic Web will bring meaning to the Internet, making it possible for web agents to understand the information it contains. However, current trends seem to suggest that it is not likely to be adopted in the forthcoming years. In this sense, meaningful information extraction from the web becomes a handicap for web agents. In this article, we present a framework for automatic extraction of semantically-meaningful information from the current web. Separating the extraction process from the business logic of an agent enhances modularity, adaptability, and maintainability. Our approach is novel in that it combines different technologies to extract information, surf the web and automatically adapt to some changes.

1 Introduction

In recent years, the web has consolidated as one of the most important knowledge repositories. Furthermore, the technology has evolved to a point in which sophisticated new generation web agents proliferate. They enable efficient, precise, and comprehensive retrieval and extraction of information from the vast web information repository. A major challenge for these agents has become sifting through an unwieldy amount of data to extract meaningful information. The following important factors contribute to these difficulties:

- Most web pages are available in human-readable forms, and they lack a description of the structure or the semantics associated with their data.
- The layout and the aspect of a web page may change unexpectedly. This changes may invalidate the automatic extraction methods used so far.
- The access to the page that contains the information in which we are interested may involve navigating through a series of intermediate pages, such as login or index pages.

Several authors have worked on techniques for extracting information from the web, and inductive wrappers are amongst the most popular ones [2,6,7,9,10].

* The work reported in this article was supported by the Spanish Inter-ministerial Commission on Science and Technology under grant TIC2000-1106-C02-01

A. Banks Pidduck et al. (Eds.): CAISE 2002, LNCS 2348, pp. 697–700, 2002.

They are components that use a number of extraction rules generated by means of automated learning techniques such as inductive logic programming, statistical methods, and inductive grammars. Although induction wrappers are suited, they do not associate semantics with the data they extract.

There are also some related proposals in the field of databases, e.g., TSIMMIS [5] and ARANEUS [8]. Their goal is to integrate heterogeneous information sources such as traditional databases and web pages so that the user can work on them as if they were a homogeneous information source. However, these proposals lack a systematic way to extract information from the web because extraction rules need to be implemented manually, which makes them not scalable and unable to recover from unexpected changes on the web.

In this article, we present a proposal that achieves a complete separation between the logic an agent encapsulates, the navigation path to the page that contains the information it needs, and way to extract it. Giving the necessary mechanisms to associate semantics with the extracted information and to adapt automatically to some changes on the web.

2 Our Proposal

Our proposal aims at providing agent developers with a framework in which they can have access to semantically-meaningful data that resides on heterogeneous, user-friendly web pages. It relies on using a number of agents that we call information channels or IC for short. They allow to separate the extraction of information from the logic of an agent, and offer agent developers a good degree of flexibility. In order to allow for semantic interoperability, the information they extract references a number of concepts in a given application domain that are described by means of ontologies.

2.1 The Architecture

Figure 1 sketches the architecture of our proposal. As we mentioned above, information channels are at its core because they specialise in extracting information from different sources, and are able to react to information inquiries (reactivity) from other agents (social ability). They act in the background proactively according to a predefined schedule to extract information and to maintain the extraction rules updated.

Each information channel uses several inductive wrappers to extract information so that they can detect inconsistencies amongst the data they extract. If such inconsistencies are found, they then use a voting algorithm to decide whether to use the data extracted by most wrappers or regenerate the set of extraction rules on the fly. This may happen because of an unexpected change to the structure of the web page that invalidates the extraction rules used so far.

There is also an agent broker for information extraction that acts as a trader between the agents that need information from the web and the set of available information channels. When an agent needs some information, it contacts the

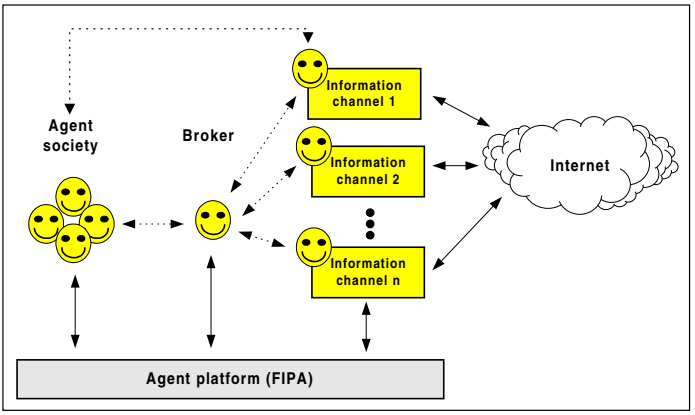

Fig. 1. Proposed architecture

broker, which redirects the request to the appropriate information channel, if possible. This way, agents need not be aware of the existence of different ICs, which can thus be adapted, created or removed from the system transparently (yellow pages).

2.2 Information Channels

. An information channel is characterised by the following features: a set of ontologies, a set of extraction rules, and a navigational document.

- *The Ontologies*. The ontologies [3] associated with an IC describe the concepts that define the semantics associated with the information we are going to extract from a high-level, abstract point of view. An ontology allows to define a common vocabulary by means of which different agents may interoperate semantically.
- *The Extraction Rules.*The extraction rules allow to define how to have access to the information in which we are interested. To generate them, we need a set of sample pages containing test data on which we use inductive techniques. To endow the sample pages with semantics, we also need to annotate them with DAML [1] tags that associate the concepts that appear in them with their ontologies. Once the sample pages have been annotated, we can generate the extraction rules using a tool called *wrapper generator*.
- *The Navigational Document*. A navigational document defines the path to intermediate pages, such as login or index pages, we need to visit in order to have access to the page that contains the information in which we are interested. Roughly speaking, a navigational document may be viewed as a state machine in which each state is a web page, and the transitions between states may be viewed as the navigation from the page associated with the initial state to the page associated with the destination state.

Once we have set up an abstract channel, we can send it messages to retrieve information about a given book by means of the broker. The content of the messages in DAML is based on an ontology that defines communication [4].

3 Conclusions and Future Work

In this article, we have sketched some ideas that show that the process of extracting information from the web can be separated from the business logic of a web agent by means of abstract channels. They rely on inductive wrappers and can analyse web pages to get information with its associated semantics automatically. Thus, we are contributing to bring together the community of agents programmers and the semantic web.

In the future, we are going to work on an implementation of our framework in which data sources can be more heterogeneous (databases, news servers, mail servers, and so on). Extraction of information from multimedia sources such as videos, images, or sound files will be also paid much attention.

References

1. DARPA (Defense Advanced Research Projects Agency). The darpa agent mark up language (daml). `http://www.daml.org`, 2000. 699
2. W. W. Cohen and L. S. Jensen. A structured wrapper induction system for extracting information from semi-structured documents. In Workshop on Adaptive Text Extraction and Mining (IJCAI-2001), 2001. 697
3. O. Corcho and A. Gómez-Pérez. A road map on ontology specification languages. In Workshop on Applications of Ontologies and Problem solving methods. 14th European Conference on Artificial Intelligence (ECAI'00), 2000. 699
4. S. Cranefield and M. Purvis. Generating ontology-specific content languages. In Proceedings of Ontologies in Agent Systems Workshop (Agents 2001), pages 29–35, 2000. 700
5. H. García-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman, and J. Widom. Integrating and accessing heterogeneous information sources in TSIMMIS. In The AAAI Symposium on Information Gathering, pages 61–64, March 1995. 698
6. C. A. Knoblock. Accurately and reliably extracting data from the web: A machine learning approach. Bulletin of the IEEE Computer Society Technical Com-mittee on Data Engineering, 2000. 697
7. N. Kushmerick. Wrapper induction: Efficiency and expressiveness. Artificial Intelligence, 118(2000):15–68, 1999. 697
8. G. Mecca, P. Merialdo, and P. Atzeni. ARANEUS in the era of XML. Data Engineering Bullettin, Special Issue on XML, September 1999. 698
9. I. Muslea, S. Minton, and C. Knoblock. Wrapper induction for semistructured, web-based information sources. In Proceedings of the Conference on Automated Learning and Discovery (CONALD), 1998. 697
10. S. Soderland. Learning information extraction rules for semi-structured and free text. Machine Learning, pages 1–44, 1999. 697