Semantic-driven Management and Search for Resources in Research Community

Junsheng Zhang Zhixian Yan
Key Lab of Intelligent Information Processing
Institute of Computing Technology, Chinese Academy of Sciences
{junsheng,yanzx}@kg.ict.ac.cn

Abstract

In recent years, semantic search has become one hot motivation of the semantic web. In this paper, we propose a semantic-based resource management and search architecture and its implementation in research community, including most of the research relevant resources. We have built two authoring tools respectively for concepts and instances. With the authoring tools, semantic search portal is provided, which has three different interfaces, respectively for retrieval of concept , instance and semantic relationship. Compared with the traditional search methods, semantic search methodology not only gives us more interesting and satisfying search results, whose precision is higher than those of traditional ones only based on keywords, but also displays the results in a more structural and semantic way.

1 Introduction

With the Web information explosion, people are exhausted to find right thing at right time. Traditional search engines such as Google have resolved this problem in some extent, but sometimes they can not meet web users real search requirements for lack of semantics. The essence of traditional search engines is keyword based retrieval with statistic methodology. Although statistic is one of the most efficient tools, it also has some limitations for lack of semantics understanding.To diminish the gap between what the users really require and what the search engine can provide, we need a new kind of search engine, that is, semantic search.

Some semantic search research work have been done. Ref. [9] aims at improving the search of documents on Semantic Web, and divides the search into two kinds: navigation search and research search. Ref. [1] integrates heterogeneous data into a common schema which the semantic search is based on. Some researchers focus on the semantic ranking of entities and relationships. Some mechanism to rank the complex semantic relationships between retrieved entities is provided in ref. [6], while resources on the Semantic Web based on semantic links are ranked in ref. [5]. Most of the semantic links are explicated as semantic association in semantic link network [3, 4]. Given a set of keywords, ref. [2] seeks to find important related entities by using a spread activation mechanism with the domain knowledge provided by experts or knowledge engineers.

In this paper, we establish a kind of semantic-driven resource management and search model, architecture and its implementation in research community. Searching with the developed system, query results will be returned in a more structural way. Besides, the relations among different concepts and instances are listed explicitly, so the related resource can be acquired conveniently.

2 General Architecture

2.1 Semantic Scenario of Research Community

There are two layers in a general scenario of research community. The upper layer is the concepts of research community and relationships between them. In addition, there are also linkages from concepts to literal descriptions. The lower layer is the instances corresponding to concepts in the upper layer. Each instance has a linkage to the concept which it belongs to, and there are also linkages between instances, which are inherited from their belonged concepts. In our architecture, the upper layer is call "Knowledge Schema", and the lower layer is called "Instance Base".

2.2 Semantic-driven Search Architecture

As shown in Fig. 1 The architecture for semantic-driven resources management and search has four layers. They are as the following From the top to bottom.

- Graph User Interface: Users give query to the system and get the query results. To retrieve satisfied

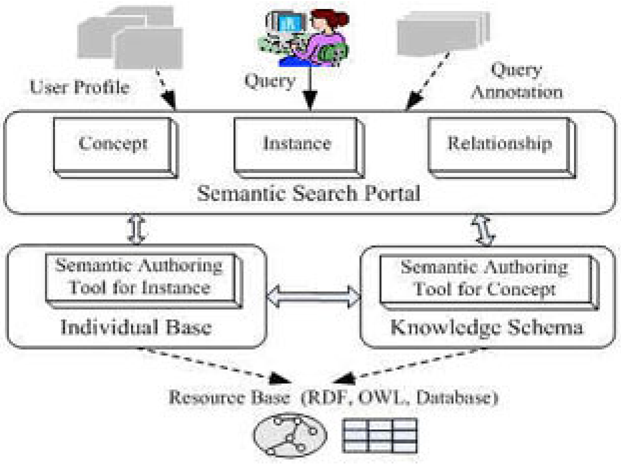

Figure 1. architecture for semantic-driven resource management and search

information, it is needed to excavate hidden semantic meanings among the keywords and user search requirements. One direct and effective strategy for excavation of semantic meanings is *Query Annotation*. Another powerful strategy for excavation of semantic meanings is *Semantic User Profile*, which records user's background, interest and search preference etc. The *profile* is considerably effective to achieve more individualized and adaptive search results. While its construction is complex and changeable, we will not pay more attention to this strategy in this paper, and more detailed work are in ref. [8, 10–13].

- Semantic Search Portal: The former two components are used for resources management. Nevertheless, only management is far from meeting the user's demand, and it is just for information modeling. Resource management is not the goal, but a means to achieve more effective retrieval. The semantic search portal provides three search interfaces respectively for concept, instance and relationship. Section 3.2 will discuss the portal in more details.

- Semantic Authoring Tools: For there are two layers in research community scenario, two relevant components are also needed. Based upon them, semantic search portal for search is given. Knowledge Schema is responsible for modeling domain knowledge. Instances relevant to the concepts are modeled and stored in Instance Base, which is responsible for modeling instances of certain domain. To make human-machine interaction easier, we have built two authoring tools for concepts construction and instances construction respectively.

- Resource Repository: and the resource can be stored in database, or semantic web files, such as RDF, OWL etc.

3 Implementation

3.1 Semantic Authoring Tools

In our architecture, the semantic search is based on the semantic-driven resource management. To manage resource in semantic way, we have built two semantic authoring tools. One is for domain concepts and the other is for domain instances.

3.1.1 Authoring Tool for Concept

The authoring tool provides functions to define domain concepts and the relationship between those concepts. The former is also called classes in ontology technology and can be seen as nodes; the latter is also called relations and can be seen as edges. To describe the concepts explicitly, the tool also needs to define properties related to the concepts. Properties can be divided into two kinds, one is data-type properties, and the other is object properties. The former is to define basis information for concepts, such as concept name, memo, etc. The latter is for defining associated information, for example person "Z. Yan" has a research direction "knowledge grid", which would be an instance or object of concept Research Area. Particularly, the relationships between concepts can be regarded as the object properties.

One thing we haven't discussed above is the storage strategy for the semantic information. In Section 2.2 discussed several resource bases, such as database and RDF/OWL files. Database model is mature, and has many approachable API for implementation, not only in construction and deployment, but also in retrieval. But its semantic representation ability is low and limited; RDF/OWL model with more serviceable semantic meanings can store and search concepts and instances by means of semantics way, while its retrieval efficiency is faraway to meet demand.Consequently, in above functions of authoring tool for concept, we have implemented two type storage strategies, one is by database for instance detail information, and the other is by RDF for concept information.

3.1.2 Authoring Tool for Instance

Semantic authoring tool for concept discussed above is the basis of authoring tool for instance. As the result of the former tool, each concept has its own integrated schema and strage model. With these schemas, we can define and manage related instances.

Coming to the instances storage, there are also two strategies: database storage or semantic web storage such as RDF. Using database, the related data tables are constructed dynamically according to the former concepts, so each concept has a table, and sometimes we also need tables to store the relationships. For example, we have *paper* table to store papers, and *paper2area* table to store relationships between papers and research areas; while using RDF files directly, each concept has a RDF schema, and the defined instance ought to be coincident with the schema strictly. The information to be conserved about concept is less than that of instance, while the concept has more semantic relationships to be stored and retrieved.

During the system implementation, we choose a fairly mature semantic model RDF as the crucial repository for concept, which has a successful Java API named Jena (http://jena.sourceforge.net/) provided by HP Labs Semantic Web research group, notwithstanding we also have put it into practice by means of database. When turn to instance storage, considering its abundance, we just choose database for its powerful storage and effective retrieval efficiency.

3.2 Semantic Search Portal

With the two authoring tools mentioned above, we have constructed and stored domain concepts and relevant instances as a whole integrated resource base. Next, we propose semantic portal for those resources retrieval, which can be ascribed to abundant semantic information, including concepts, instances and relationships, built in the two semantic authoring tools.

Different from traditional search engine, the semantic portal gives the search results in semantic mechanism, in which, the query is not just non-sense keyword but with some useful semantic meanings. Meanwhile the retrieved resources are not just in non-sense results list, but organized by semantic structures.

3.2.1 Semantic Search Interface for Concept

This portal provides the interface for searching domain concepts. In the authoring tool for domain concept, we have defined knowledge schema of domain ontology as a whole structure, which includes domain concepts and relationships between those concepts. Given a concept, the portal will return the concept's schema with the data-type properties and object properties.

3.2.2 Semantic Search Interface for Instance

In order to achieve more semantic strategies, we use front-end semantic methodologies and adopt semantic annotation for query, that is, when submitting a query with keywords, user can also denote it some concept. With the concept, information will be retrieved more precisely.

In research community, domain concepts are limited, and in our prototype system, there are about ten concepts, such as Person, Researcher, Student, Paper etc. These concepts are fairly stable and the relationships are moderately clear, which enable people pay more attention to the interested instances. Different from traditional search engine, the portal provides the retrieved resources not in a list with little structure information, but in a more semantic list, in which each object not only has structure depiction but also has relevant resources according to the semantic association.

During the query process, users can select an item from the list to specify information types that he intends to search. The search results are displayed in a structured manner. Each object is described with semantic information to guide user's navigation, such as the type of object, introduction, link-address, related objects, etc. Users can click the related topics to get more focused information.

3.2.3 Semantic Search Interface for Relationship

Relationships between resources are also interesting, and can be classified into three categories: relationship of concept and concept, instance and instance, and concept and instance.

- Relationship between concept and concept

  Given two concepts, it is intresting to discover the semantic association between them. Taking "researcher" and "research team" as example, we may discover their semantic relationship "MemberOf"; although relationship is not defined, but "Researcher" is subclass of "Person" and "Person" has relationship "MemberOf" with "Research Team", so we can conclude that the relationship "MemberOf" exists. Therefore, relationships can be inferred, according to sub-class properties, and in the future, by absorbing the semantic links from Semantic Link Network [4, 7], we may get much more inference between those concepts. The graph below illustrates a simple reasoning role for inferring semantic relationship as used before.

- Relationship between concept and instance

  Given one concept and one instance, we can discover the semantic association between them. For example, when given instance "Z.Yan" and concept "Person", we may attain semantic relationship "Z.Yan" is an Instance of "Person" retrieved from our implementation system. Although we haven't defined the instance "Z.Yan" in the authoring tool for concept "Person", it has been defined as an instance of "Student". As defined, "Student" is a subclass of "Person". Consequently, according to the following reasoning rule, we may safely draw a conclusion that relationship "Z.Yan is an Instance of person" exists. More powerful and efficient reasoning rules can be obtained or adopted from semantic link network [3].

- Relationship between instance and instance

  The relationships between instances are also interesting and abundant. Given two instances, we can discover the semantic association between them. For example, given two students "Z.Yan" and "N.Zhang", we can find their common semantic properties, such as they are both instance of concept "Student", they are both supervised by researcher "Zhuge". Retrieving relationship between instances is comparatively complex. With the number increase of instances, relationships may grow at exponential rate. As a result, in our current implementation, semantic relationship retrieval between instances is limited, and we have not proposed an effective strategy for ranking those semantic relationships yet.

3.3 System Data Storage

According to semantic model, the implementation has been divided into three parts: *Concept Management*, *Instance Management*, and *Semantic Search Portal*. Considering the system efficiency and semantics, Database and RDF are both used in *Concept Management*.

During database implementation , we adopt J2EE light architecture, *Struts* (http://struts.apache.org/) and *Hibernate* (http://www.hibernate.org/). *Struts* and *Hibernate* enable the development conveniently. When using RDF as the repository, *Jena* is adopted, which is a Java framework for building semantic web application developed by HP Lab Semantic Web Group. The graph exhibition for semantic structure of research community during the implementation adopts *JGraph* (http://www.jgraph.com/), which is excellent in drawing graph.

4 Conclusion and Future Work

Semantics can impact the search deeply for its widespread availability. Based on semantic mechanism, this paper proposes a kind of semantic-driven resource management and search model, and provides an implementation for the resources in research community. The paper presents two semantic authoring tools for domain resources management, , upon which, semantic search portal is constructed, which has three sub-interfaces. In turn, the relationship can be further classified into three sub-types.

Our ongoing work is taking this semantic-driven resources management and search architecture into the application of Dunhuang culture research, which is a prominent archaeological field about ancient Chinese religion and art. In the future, we will go further on semantic search for relationship, especially ranking for semantic relationships retrieved.

References

[1] A. Calsavara, G. Schmidt, Semantic Search Engines, ISSADS 2004, Springer, 2004, pp.145-157.

[2] C. Rocha, D. Schwabe, Poggi de Aragao, M. A Hybrid Approach for Searching in the Semantic Web, WWW2004, New York City, May 17-22, 2004.

[3] H. Zhuge, Active E-Document Framework ADF: Model and Tool, Information and Management, 41(1), 2003, pp.87-97.

[4] H. Zhuge, The Knowledge Grid. World Scientific Publishing Co., Singapore, 2004.

[5] H. Zhuge, L. Zheng, Ranking Semantic-linked Network, WWW2003, Budapest, May, 2003.

[6] K. Anyanwu, A. Maduko, A. Sheth, SemRank: Ranking Complex Relationship Search Results on the Semantic Web, WWW2005, Chiba, Japan, May 10-14, 2005.

[7] H. Zhuge, Autonomous Semantic Link Networking Model for the Knowledge Grid, Technical Report of Knowledge Grid Center, KGRC-2006-01, www.knowledgegrid.net/TR. Jan, 2006.

[8] M. Kobayashi , K. Takeda, Information Retrieval on the Web, ACM Computing Surveys, 32(2), June, 2000, pp.144-173.

[9] R. Guh, R. McCool, E. Miller, Semantic Search, WWW2003, Proc. of the 12th International Conference on World Wide Web, ACM Press, 2003, pp.700-709.

[10] H. Zhuge, A Knowledge Grid Model and Platform for Global Knowledge Sharing, Expert Systems with Applications, 22, 2002, pp.313-320.

[11] H. Zhuge, China's E-Science Knowledge Grid Environment, IEEE Intelligent Systems, 19(1), 2004, pp.13-17.

[12] H. Zhuge, Discovery of Knowledge Flow in Science, Communications of the ACM, 49(5), 2006, pp.101-107.

[13] H. Zhuge, Resource Space Grid: Model, Method and Platform, Concurrency and Computation: Practice and Experience, 16(14), 2004, pp.1385-1413.