Journal of Southeast University (English Edition) Vol. 23, No. 3, pp. 317 – 321 Sept. 2007 ISSN 1003—7985

Design and implementation of semantic search engine Smartch

Wen Kunmei Lu Zhengding Li Ruixuan Sun Xiaolin

(College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract: To integrate reasoning and text retrieval, the architecture of a semantic search engine which includes several kinds of queries is proposed, and the semantic search engine Smartch is designed and implemented. Based on a logical reasoning process and a graphic user-defined process, Smartch provides four kinds of search services. They are basic search, concept search, graphic user-defined query and association relationship search. The experimental results show that compared with the traditional search engine, the recall and precision of Smartch are improved. Graphic user-defined queries can accurately locate the information of user needs. Association relationship search can find complicated relationships between concepts. Smartch can perform some intelligent functions based on ontology inference.

Key words: semantic search; search engine; semantic search engine Smartch; semantic web; ontology

The technology of web search has been widely used around the world. However, the precision and recall of existent search engines are not good enough to satisfy user requirements. At the same time, they are not intelligent enough to carry out some special queries. Presently most search engines are based on keyword or full-text index. As the most important application of the semantic web[1], semantic search is being paid more and more attention to. The concept of semantic search is put forward in Ref. [2]. Semantic search integrates the technologies of the semantic web and search engine to improve the search results obtained by current search engines and evolves to the next generation of search engines built on the semantic web.

Based on the ontology function in semantic search, we sort semantic search into three types: the incremental semantic search engine based on the traditional search engine[3-6], intelligent semantic search based on ontology inference[7-10] and other semantic search[11-13].

Smartch, a new kind of semantic search engine based on domain ontology, is implemented. Our initial purpose is to improve the search result ontology inference. Existent search engines should do well with not only higher precision but also higher recall. At the same time, the traditional search engine cannot perform complicated constraint queries and relationship queries between resources. Smartch integrates semantic search and search engine to effectively and truly find user information. It is focused on resolving the following questions: basic query with incremental recall and unreduced precision based on ontology keyword parsing; concept query within domain ontology; association relationship query between resources; user-defined queries based on graphic manner.

1 Architecture of Semantic Search

A semantic search does not search the whole Internet, it only searches some special domain. A complete process of a semantic search includes the following steps:

① Knowledge base is established based on some domain ontology;

② Crawlers build the index base for the internal websites which belong to a special domain;

③ System accepts the queries asked by users;

④ Inference engine analyzes user's query, performs a reasoning operation and then returns the inference results to system;

⑤ According to the inference results, the system finds the record in the index base;

⑥ Ranking the search results and combining with the reasoning results to form the final results, the system returns the final results to the user.

Based on the model proposed in Ref. [14], we build the architecture of semantic search, shown in Fig. 1. The system is composed with three parts. They are semantic search, resource and Intranet based on some domain. The most important part is semantic search. To implement semantic search, we need to construct four components, including repository based on domain on-

Received 2007-05-18.

Foundation item: The National Natural Science Foundation of China (No. 60403027).

Biographies: Wen Kunmei (1979—), female, graduate; Lu Zhengding (corresponding author), male, professor, zdlu@hust.edu.cn.

tology, reasoning engine, interface engine and crawler. Resources contain arbitrarily many websites. We give the details of these four components as follows.

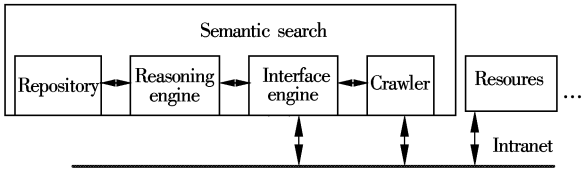

Fig. 1 Architecture of semantic search engine

1.1 Repository based on ontology

The repository stores domain ontology owl files. Owl files should be the right domain ontology. Loading ontology files and parsing RDF graph structure is time-consuming work. Database is an answer to resolve it. It is used to store ontology data. The method saves great time used for reasoning.

1.2 Crawler

The crawler's task is catching the intranet resources and building indices for them. First crawlers get the URL information of starting resource. Secondly the crawling range is set, where the range is restricted in the Intranet. Then the web pages are crawled and indices are built based on these web pages. The system needs to perform the whole crawling process timely and update the index files to ensure the web pages are updated. The crawling process is shown in Fig. 2. The crawler controller is used to maintain URL information and set the crawling range.

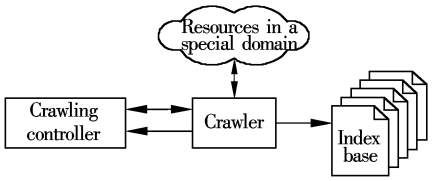

Fig. 2 Crawling process of semantic search engine

1.3 Reasoning engine

The reasoning service of semantic search is carried out by reasoning engine. It returns the inference results to the traditional search engine or directly to the user. The reasoning engine can perform concept query, instance query and association relationship query, providing powerful reasoning functions. The reasoning is implemented based on the tableau algorithm.

The semantic search engine provides four kinds of reasoning services: consistency reasoning which checks whether the ontology contains inconsistent facts; classification reasoning which calculates the consumption relationships among concepts and establishes a concept hierarchy graph; realization reasoning which finds the most direct concept of the appointed instance; and query reasoning is mainly for individual queries.

The reasoning engine obtains the user's keyword from the interface engine and treats it as a concept and then performs the concept reasoning operation, if the concept exists in the ontology knowledge base, the system returns its equivalent concept, sub-concept and super-concept, then performs the instance reasoning operation and finds all the individuals of the concept; if the concept does not exist, then the reasoning engine regards it as an instance, and does the instance reasoning operation. The reasoning results are put forward to users and also to traditional search engine. If the concept does not exist then the results are directly forwarded to the traditional search engine. The whole reference process is shown in Fig. 3.

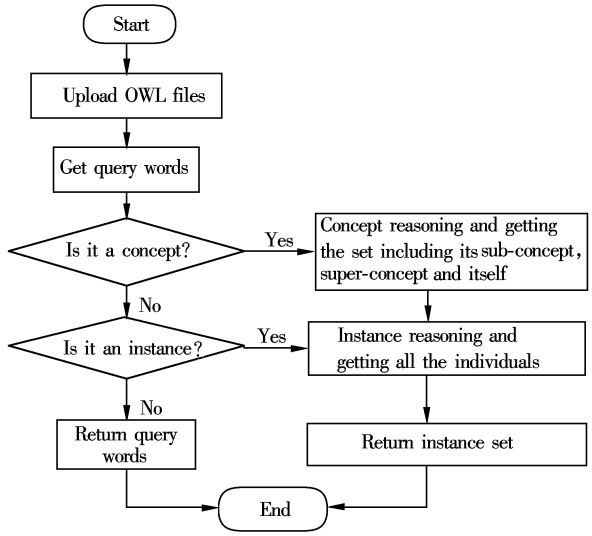

Fig. 3 Reasoning process of semantic search engine

1.4 Interface engine

The user can interface with semantic search engine using four forms: keyword query, concept query, association relationship query and user-defined query based on graphic mode.

The user submits the query. The query is firstly put forward to the reasoning engine. After the reasoning operation is completed, the inference results are forwarded to the search engine. And related web pages are found from the index base. The reasoning results are combined with related pages to generate the final results. Eventually these results are ranked according to their important values and returned to users.

The user can define the query through a graphic mode. The user-defined method is shown in Fig. 4. The first step is selecting one ontology concept. Then all the properties of the concept are shown to be extended. The user clicks the selected property to expand the query graph and restrict it. And then the user can choose the property's range concept. Again the process

can be circular until the user-defined process is over. Finally the user needs to set the query concept.

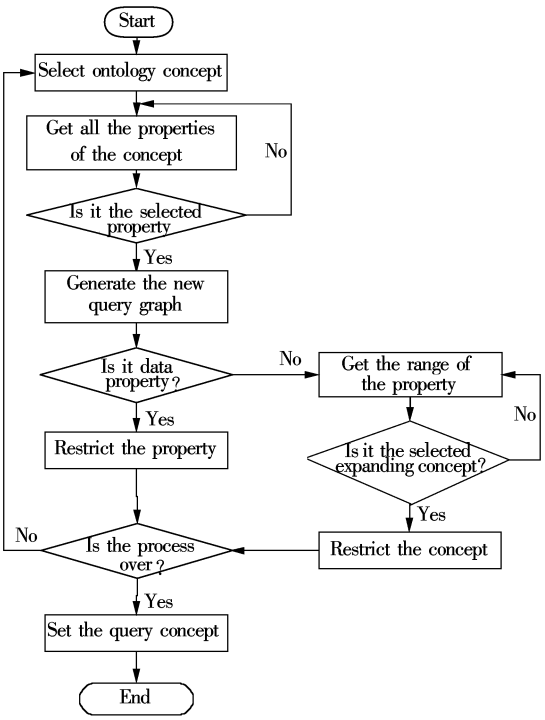

Fig. 4 User-defined process of graphic query

2 System Implementation

The implementation of semantic search is based on the integration of search and inference. We use Lucence (an open resource tool) as our Intranet search tool and Jena (another open resource tool) as our ontology parsing tool. To improve the efficiency for the operation of uploading owl files, the SQLServer is used as the tool for ontology data persistence. Based on Jena, we develop the SQLServer database interface (for the reason that Jena API only provides MySQL, Oracle interface). The ontology model can be directly read from the database. The method can save a great deal of time. We use Pellet which is an open-resource reasoning tool as our reasoning engine, the reference services can be done through calling the Pellet API.

The user-defined process of graphic query is implemented based on scalable vector graphics (SVG). On the client side SVG graphic technology and Ajax are adopted. On the server side servlet plays an important role of accepting requests from the user. Grahpviz is the tool to generate SVG format string flow.

From the user's view, the whole process of semantic search is shown in Fig. 5. There are twelve steps as follows:

① Choose keyword query;

② Choose concept query;

③ Choose user-defined query;

④ Input keywords;

⑤ Input concept;

⑥ User-defined graphic query;

⑦ Submit the query to reasoning engine;

⑧ Provide the expanded results to search engine;

⑨ Submit the reasoning results to generate final result pages;

⑩ Put index results into ranking component;

⑪ Transfer the ranked results to generate final result pages;

⑫ Return the final search results to user.

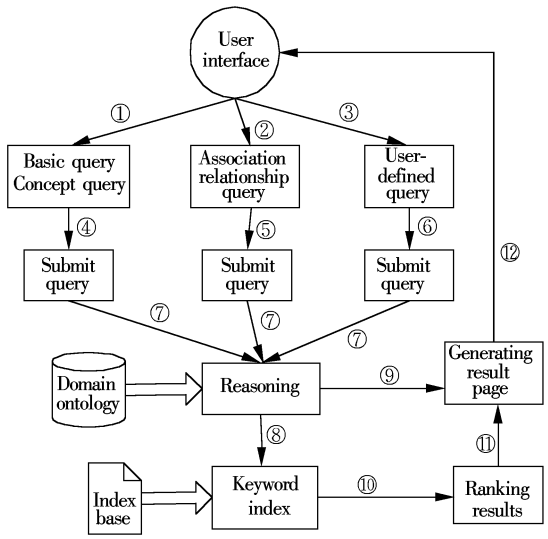

Fig. 5 Semantic search process from user's view

We build our IDC lab ontology "idc_onto.owl". It is an academic ontology. The final result pages are combined inference results with search results. We built the lab ontology mentioned above. And some instances were existent in the ontology. If the user chooses basic keyword query, such as "计算机", the reasoning engine first regards it as a concept, then through ontology inference we know that "电脑" has the same meaning as "计算机", so "计算机" and "电脑" are both returned to the search engine. If the user chooses a concept query, such as "博士生", the reasoning engine gets the equivalent concept, sub-concept and super-concept of the concept. This result is shown in the top right corner. At the same time, the reasoning engine needs to index all the individuals of the concept "博士生", and then return all the individuals to search engine. Finally the search engine finds all the results which satisfy the request. The instance of "descript" includes "唐卓", "孙小林" and "余艳玮". The search engine also regards them as keywords and searches the web pages related to these keywords. The brief introduction from the ontology base about the instance is al-

so shown at the bottom right. The detailed description is shown in Fig. 6.

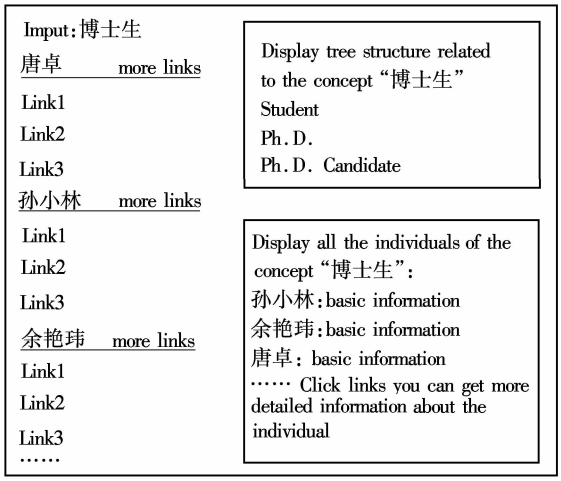

Fig. 6 Generating result pages

3 Experiment and Conclusion

Semantic search engine Smartch is implemented. We do some experiments.

First we test basic query. "论文" as a keyword is inputted. Besides the web pages containing "论文", the pages containing "paper" are also returned. The result is shown in Fig. 7.

Fig. 7 Basic keyword query result

If the user inputs the concept "教师", all the individuals of the concept "教师" existing in the ontology are returned to the user. The web pages related to these individuals are also returned to the users. The brief introduction from the ontology base about the instance is also shown at the bottom right.

If the user wants to ask the question "Who teaches the course semantic web and ontology?", we can resolve it using user-defined graphic mode. The finished defined query graph is shown in Fig. 8. First, the user chooses the concept "教师", and then displays all the properties of this concept. The user can restrict selected property "授课". "语义网与本体论" is set as the instance of the concept "课程". Finally, the user defines the concept "教师" as a query variable. The reasoning engine retrieves all the instances of the submitted concept. The final display format is the same as the concept query.

Fig. 8 User-defined graphic mode

If the user inputs two entities, it means that the user wants to find the relationship between them. In Smartch, we input "r1" and "r2", and the system returns the relationship existing between them. The interface is shown in Fig. 9.

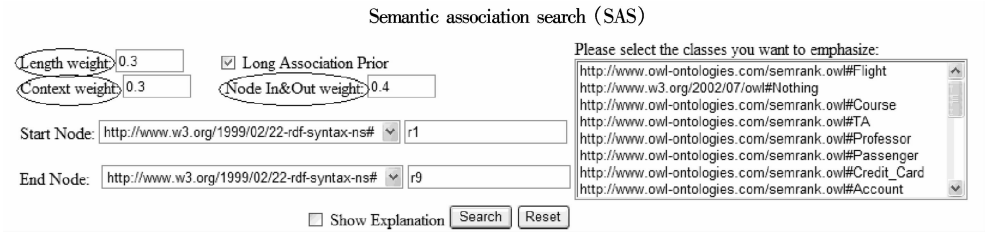

Fig. 9 Association relationship search interface

Semantic search is different from traditional search. It uses semantic search technology to improve the search results. We develop the semantic search engine Smartch. The experiment shows that Smartch can improve recall, through keyword parsing based on the ontology and it can extend keyword to its equivalent concept, sub-concept. The concept query searches all the instances of concept through inference. And the user-defined method can inerrably ensure the semantic information which is hidden in the user's query and improve the precision. Smartch can also find out the association relationship between two entities. So it can implement some intelligent functions compared to the traditional search engine.

References

[1] Berners-Lee T, Hendler J, Lassila O. The semantic web [J].

Scientific American, 2001, **284**(5): 34 – 43.

[2] Guha R, McCool R, Miller E. Semantic search [C]//*Proc of the* 12*th International World Wide Web Conference*. New York: ACM Press, 2003: 700 – 709.

[3] Guha R, McCool R. TAP: a semantic web test-bed [J]. *Journal of Web Semantics*, 2003, **1**(1): 81 – 87.

[4] Guha R, McCool R. The tap knowledge base [EB/OL]. (2005-02-02) [2007-04-20]. http://tap. stanford. edu.

[5] Guha R, McCool R. Tap: towards a web of data [EB/OL]. (2005-02-02) [2007-04-20]. http://tap. stanford. edu.

[6] Airio E, Jarvelin K, Saatsi P, et al. CIRI: an ontology-based query interface for text retrieval [C]//*Proc of the* 11*th Finnish Artificial Intelligence Conference*. Vantaa, Finland, 2004: 73 – 82.

[7] Heflin J, Hendler J. Searching the web with shoe [C]// *Proc of AAAI Workshop on AI for Web Search*. San Jose: AAAI Press, 2000: 450 – 455.

[8] Shah U, Finin T, Joshi A, et al. Information retrieval on the semantic web [C]//*Proc of the* 10*th International Conference on Information and Knowledge Management*. New York: ACM Press, 2003: 461 – 468.

[9] Mayfield J, Finin Tim. Information retrieval on the semantic web: integrating inference and retrieval [C]//*SIGIR*, *Workshop on the Semantic Web*. Toronto, 2004: 461 – 468.

[10] Aleman-Meza Boanerges, Nagarajan Meenakshi, Ramakrishnan Cartic, et al. Semantic analytics on social networks: experiences in addressing the problem of conflict of interest detection [C]//*Proc of the* 15*th International Conference on World Wide Web*. Edinburth, Scotland, 2006: 407 – 416.

[11] Popescu Ana-Maria, Etzioni Oren. Extracting product features and opinions from reviews [C]//*Proc of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Morristown, NJ, USA: Association for Computational Linguistics, 2005: 440 – 448.

[12] Cafarella Michael J, Downey Doug, Soderland Stephen, et al. KnowItAll: fast, scalable information extraction from the web [C]//*Proc of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Morristown, NJ, USA: Association for Computational Linguistics, 2005: 563 – 570.

[13] Downey D, Etzioni O, Soderland S. A probabilistic model of redundancy in information extraction [C]//*Proc of the* 19*th International Joint Conference on Artificial Intelligence*. Edinburth, Scotland, 2005: 1034 – 1041.

[14] Wen Kunmei, Lu Zhengding, Li Ruixuan, et al. A semantic search conceptual model and application in security access control [C]//*Proc of the First Asian Semantic Web Conference*. Beijing, China, 2006: 366 – 376.

语义搜索引擎 Smartch 的设计与实现

文坤梅　卢正鼎　李瑞轩　孙小林

（华中科技大学计算机科学与技术学院，武汉 430074）

摘要：为了将推理与文本检索有效融合起来，提出了一种包含多类型查询的语义搜索引擎体系结构，在此基础上设计并实现了语义搜索引擎系统 Smartch. Smartch 基于合理的推理流程和图形化定制过程，提供 4 种形式的搜索服务，分别是基本搜索、概念搜索、图形化定制搜索及关联关系搜索. 实验结果表明语义搜索引擎 Smartch 和传统搜索引擎相比，在本体推理的基础上，查全率和查准率上有一定的提高，图形化定制查询可准确定位用户需查询的概念，关联关系搜索可发现概念之间存在的复杂关系，Smartch 实现了一定程度的智能搜索.

关键词：语义搜索；搜索引擎；语义搜索引擎 Smartch；语义 web；本体

中图分类号：TP311