Searching Linked Objects with Falcons: Approach, Implementation and Evaluation

Gong Cheng, Southeast University, China
Yuzhong Qu, Southeast University, China

ABSTRACT

Along with the rapid growth of the data Web, searching linked objects for information needs and for reusing become emergent for ordinary Web users and developers, respectively. To meet the challenge, we present Falcons Object Search, a keyword-based search engine for linked objects. To serve various keyword queries, for each object the system constructs a comprehensive virtual document including not only associated literals but also the textual descriptions of associated links and linked objects. The resulting objects are ranked by considering both their relevance to the query and their popularity. For each resulting object, a query-relevant structured snippet is provided to show the associated literals and linked objects matched with the query. Besides, Web-scale class-inclusion reasoning is performed to discover implicit typing information, and users could navigate class hierarchies for incremental class-based results filtering. The results of a task-based experiment show the promising features of the system.

Keywords: Class Hierarchy, Object Search, Query Refinement, Ranking, Reasoning, Snippet, Virtual Document

The Web is developing toward the era of the data Web. In the past two years, a large amount of RDF data has been accessible on the Web, including both native RDF data and those available via RDF wrappers. More importantly, data describing various domains such as DBpedia, GeoNames, and Flickr has been interlinked, leading to a Web of data, based on which a lot of interesting applications have been developed.

Searching is one of the most common activities in daily life. People are using text-based search engines almost every day. Thus, for ordinary Web users as well as developers, an important reason to accept the data Web is that they could find more accurate information on the data Web in a shorter time. On the hypertext Web, people who seek information have to firstly retrieve Web documents, and then extract knowledge from text by themselves. On the data Web, knowledge has been structurally formalized so that it is possible to answer the question behind a query in a more straightforward way. For example, to find relations between two people, we usually combine

DOI: 10.4018/jswis.2009081903
their names into a single keyword query and submit it to a text-based search engine. Then, we locate the two names in the text of each resulting webpage and find out possible relations by reading contexts. Evidently, such process is rather time-consuming. Comparatively, linked data exactly describes relations between objects. (In this article, an object means a URI identifying an individual-level entity, that is, neither a class nor a property.) Therefore, it is possible for a Semantic Web search engine to automatically find and present accurate relations between two people.

Nowadays, several Semantic Web search engines have been released. Most of them focus on RDF document search (d’Aquin, Baldassarre, GridinoC, Sabou, Angeletou, & Motta, 2007; Oren, Delbru, Catasta, Cyganiak, Stenzhorn, & Tummarello, 2008) or ontology search (Ding, Pan, Finin, Joshi, Peng, & Kolari, 2005). An RDF document serializes an RDF graph, and an ontology, as a schema on the Semantic Web, usually defines some classes and properties. Both RDF document search and ontology search are useful to application developers but can hardly attract and serve ordinary Web users. In contrast, object search is more relevant to the information needs of ordinary Web users. However, very few Semantic Web search engines, for example, SWSE (Harth, Hogan, Delbru, Umbrich, O’Riain, & Decker, 2007), focus on Web-scale object search. They usually match keyword queries only with literals associated with objects, but have not exploited the textual descriptions of associated links and linked objects. Besides, the generation of query-relevant structured snippets has not been addressed in object search.

To efficiently meet the challenge, we have significantly improved the object search function of our Falcons system (http://iws.seu.edu.cn/services/falcons/), which is basically a keyword-based Semantic Web search engine. The system constructs for each object a comprehensive virtual document consisting of textual descriptions extracted from its concise RDF description. Then an inverted index is built from terms in virtual documents to objects, serving keyword-based search. When a keyword query arrives, based on the inverted index, the system matches the terms in the query with the virtual documents of objects to form the result set. The resulting objects are ranked by considering both their relevance to the query and their popularity. For each resulting object, the system generates a query-relevant structured snippet to show the associated literals and linked objects matched with the keyword query. The comprehensiveness of virtual documents and the structural nature of snippets make the system go beyond searching for a particular object. For example, users can search for objects with a certain property, or can submit keyword queries describing two or more objects to seek their relations. Besides, the typing information of objects, expanded by class-inclusion reasoning, is used to provide class-based query refinement. A mechanism for recommending subclasses is implemented to allow navigating class hierarchies for incremental results filtering.

This article substantially extends our previous conference paper (Cheng & Qu, 2008) in the following aspects: the virtual document of an object includes not only the textual descriptions of its linked objects but also the names of the links, so that more diverse keyword queries are supported; query-relevant structured snippets are provided to help users quickly determine the relevance of the resulting objects and in some cases directly obtain knowledge; a task-based evaluation is performed to compare the system with traditional Web search engines and other Semantic Web search engines.

RELATED WORK

TAP (Guha, McCool, & Miller, 2003) is one of the earliest keyword-based Semantic Web search systems. It matches the terms in a keyword query with the labels of objects, and then selects and returns an object as well as its surrounding subgraph. The selection is based on popularity, user profile, and search context. With TAP, users can retrieve the information of only a single object at a time. Besides, us-
Related Content

A Context-Based Approach for Supporting Knowledge Work with Semantic Portals
[www.igi-global.com/article/context-based-approach-supporting-knowledge/2811?camid=4v1a](www.igi-global.com/article/context-based-approach-supporting-knowledge/2811?camid=4v1a)

A Broader View on Context Models towards Supporting Business Process Agility
[www.igi-global.com/chapter/broader-view-context-models-towards/60069?camid=4v1a](www.igi-global.com/chapter/broader-view-context-models-towards/60069?camid=4v1a)
Dealing with Structure Heterogeneity in Semantic Collaborative Information Systems

A Context-Based Approach for Supporting Knowledge Work with Semantic Portals