Available online at http://www.journalcra.com

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

International Journal of Current Research
Vol.3, Issue, 4, pp.040-045, April, 2011

ISSN: 0975-833X

REVIEW ARTICLE

QUEST-SEMANTIC A QUERY BASED PERSONALIZATION FOR ENHANCING SEARCH ENGINE ON WEB

*** Rekha, M. and Meyyappan, T.**

Department of Computer Science and Engineering, Alagappa University Karaikudi, Tamilnadu, India

ARTICLE INFO

Article History:

Received 10th January, 2011
Received in revised form
14th February, 2011
Accepted 11th March, 2011
Published online 17th April 2011

Key words:

Semantic Web,
SPARQL Queries,
Ontological Models(OWL),
Annotation Engine,
RDF Flat Files.

ABSTRACT

In this paper, the authors propose a framework for building semantic web support for intelligent search using RDF, ontology and SPARQL queries. Existing Key Word Searching yields 60% accurate results remaining 40% are unwanted results. On the other hand, getting results using RDBMS query processing is very slow. Current keyword-based search engines can't fully capture the intrinsic richness of natural language; synonymy and polysemy. We propose a Semantic Web search technique which yields 90% accurate results. Semantic web Search employs Annotation Engine and RDF. In the proposed technique, Flat files and Sparql queries are used, which is very fast. Results produced by the proposed technique are provided.

INTRODUCTION

In today's Web, information is primarily intended to be read and processed by humans; it can't be readily comprehended and manipulated by agents. The intelligence underlying search tasks, as well as the assessment of the retrieved pages' relevance, comes mainly from human sources, with limited support from software (Uren *et al.,* 2006). Although this type of processing is still adequate for searches returning a few hundred pages, it can't scale to the volume of information available in business, where enterprises couple the vast amount of data available on the Web with company documents and databases.

****Corresponding author:*** *rekmphil@gmail.com*

Current keyword-based search engines can't fully capture the intrinsic richness of natural language; synonymy and polysemy, for example, pose difficult problems for a keyword-based search task. Enhancing search engines with lexicons such as WordNet (Miller, 1995) can help relieve these problems, but this doesn't identify and resolve more complicated types of ambiguity. Furthermore, keyword-based search engines make little provision for the formulation of very specific queries, particularly those that make use of relationships between entities. The Semantic Web is an evolution of the current Web that represents information in a machine-readable format, while maintaining the human-friendly HTML representation (Berners-Lee *et al.,* 2001). In the Semantic Web, the resource of ontology can share by machine. It means the modeling of entities and processes used to describe both the content of a Web resource and, more importantly, the logical relations between the resources (Studer *et al.,* 1998). Ontological models allow the *annotation* of Web documents (modeling the representation of information contained in them) and thus the formulation of more precise queries to retrieve documents. Annotation normally involves creating metadata items (as instances of concepts from the ontology) to represent specific entities recognized in the resources, and then linking this metadata to the resource as its description. Many research efforts have thus focused on providing automatic or semiautomatic ways to annotate Web documents in various formats mainly text, but also structured formats such as databases. This paper is organized as follows. Section 2 presents an overview of related work on QS Search Engine on Web . The origin of Semantic Web engine using annotation stage design is discussed in Section 3. Then the specific design and implementation of Annotation engine and Search engine is given in Section 4.

RELATED WORK

With the tremendous growth of information available to end users through the Web, search engines come to play ever a more critical role. Nevertheless, because of their general-purpose approach, it is always less uncommon that obtained result sets provide a burden of useless pages. The next-generation Web architecture, represented by the Semantic Web, provides the layered architecture possibly allowing overcoming this limitation. Several search engines have been proposed, which allow increasing information retrieval accuracy by exploiting a key content of Semantic Web resources, that is, relations. However, in order to rank results, most of the existing solutions need to work on the whole annotated knowledge base. In this paper, we propose a relation-based page rank algorithm to be used in conjunction with Semantic Web search engines that simply relies on information that could be extracted from user queries and on annotated resources. Relevance is measured as the probability that a retrieved resource actually contains those relations whose existence was assumed by the user at the time of query definition.

As we have experience in using well-known search engines every day, the result set returned by search engines is really too big and is mostly useless. We have to continually click the "next page" to obtain the Web pages users really want. The reason is that, when the user wants to search some information in the Web, the search engine abstracts the information to the keyword combination and then submits it. The relationship between keywords is obvious to users, while it is not for search engines. If the Web page only includes the keywords and there is no relationship between keywords in the context of the Web page, the Web page does not provide what the user wants. In this case; we say the Web page is a keywords-isolated page. However, there are many keywords-isolated pages in the result set returned by traditional search engines. In fact, because of the constraints of the current Web architecture, search engines cannot exclude these keywords-isolated pages from the result set.

Intelligent search engine is an effective tool for solving many bottleneck problems in network information retrieval. It involves acquiring, preprocessing, representing and integrating data and information available at different levels of services (e.g. HTML/XML/RDF/OWL etc) and eventually converts them into useful intelligent semantic information of each domain. This paper proposes firstly a systemic framework for building (semi-)automatically ontology learning from web pages and considers some key problems about extracting concepts and interrelationships in ontology learning. A systemic framework for building a systemic framework for building (semi)automatically ontology learning from web pages and considers some key problems about extracting concepts and interrelationships in ontology learning. Furthermore, the agents in the search engine multiagent system exhibit various autonomic features that aim at making the system more robust and scalable (Blacoe *et al.,* 2010). The QS system has been deployed in two different commercial test cases in the UK. In the first case, QS was used to examine specific Web-published documents for commercial opportunities matching the business interests of the customer company. In the second deployment, QS was used to perform knowledge-based searches over existing database sources. In evaluating the performance of the search system in both applications, we could see that by using ontological knowledge and ontology-based annotations.

ORIGIN OF THE DESIGN

We designed QS to maximize the separation between the different types of knowledge represented domain-versus task-specific knowledge, and application versus generic knowledge. The goal of this separation if to achieve reusability and easy customization of the platform's various agents, thus allowing semantics based search in various task and domain scenarios. QS includes two main components:

1) A general framework for (semi-)automatic resource annotation based on a detailed ontological model of the domain.

2) A user-friendly search interface that allows the formulation and execution of knowledge-based queries over the generated metadata.

We designed QS for application scenarios that exploit different information sources to provides searchable knowledge. The process often differs only slightly between different application scenarios and different domains. The aim of the general framework for annotation is to abstract from different scenarios all the common implementation and policy details in order to reduce and simplify application-specific code.

A. Origin of the Idea

Semantic Web is a group of methods and technologies to allow machines to understand the meaning – or "semantics" – of information on the World Wide Web The term was coined by World Wide Web Consortium (W3C) director Tim Berners-Lee He defines the *Semantic Web* as "a web of data that can be processed directly and indirectly by machines." While the term "Semantic Web" is not formally defined it is mainly used to describe the model and technologies proposed by the W3C. These technologies include the Resource Description Framework (RDF), a variety of data interchange formats (e.g. RDF/XML, N3, Turtle, N-Triples), and notations such as RDF Schema (RDFS) and the Web Ontology Language (OWL), all of which are intended to provide a formal description of concepts, terms, and relationships within a given knowledge domain. The key element is that the application in context will try to determine the meaning of the text or other data and then create connections for the user. The evolution of Semantic Web will specifically make possible scenarios that were not otherwise, such as allowing customers to share and utilize computerized applications simultaneously in order to cross reference the time frame of activities with documentation and/or data. According to the original vision, the availability of machine-readable metadata would enable automated agents and other software to access the Web more intelligently. The agents would be able to perform tasks automatically and locate related information on behalf of the user.

Many of the technologies proposed by the W3C already exist and are used in various projects. The Semantic Web as a global vision, however, has remained largely unrealized and its critics have questioned the feasibility of the approach. In order to make it easy and useful to the secured User. Traditional all are used in search engine Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

B. Practic*al feasibility*

Critics (e.g. Which Semantic Web) question the basic feasibility of a complete or even partial fulfillment of the semantic web. They may include spurious metadata into Web pages in an attempt to mislead Semantic Web engines that naively assume the metadata's veracity. Where semantic web technologies have found a greater degree of practical adoption, it has tended to be among core specialized communities and organizations for intra-company projects. The practical constraints toward adoption have appeared less challenging where domain and scope is more limited than that of the general public and the World-Wide Web.

C. Specific Semantic Web Technology

Semantic Web has developed specific Semantic Web Technologies that could be implemented free of cost that could result in huge savings in the way the Web functions. An example of this is SPARQL, a query language. It would however be erroneous to assume that Semantic Web is something that has descended from nowhere to usher in a rethinking in everything. One may be tempted to use such terms as that a revolutionary mind set would be needed to its application etc. or it represents a paradigm shift which are all not correct and would only confuse and mask the real advantages it is offering. It is neither a total replacement nor would it substitute all that has come before it, which would and continue to exist. No doubt, there could be changes, but, these changes would build and bridge the gap by leveraging the existing assets rather than replacing them. Semantic web have more capabilities are:

- Achieve Reusability and Scalability
- Easy Customization
- Robust and Accuracy of web Search

D. Everything identifiable is on the Semantic Web

People, places, and things in the physical world will have online representations identified by Uniform Resource Identifiers which will facilitate effective integration, active participation and be conceptualised in the Semantic Web. URIs are the metadata anchor points to make semantics explicit.

E. Inaccurate Queries

We have user typically domain specific knowledge. And users don't include all potential Synonyms and variations in the query, actually user have a problem but aren't sure how to phrase.

DESIGN AND IMPLEMENTATION

In order to make the design of Quest Semantic(QS) general Sysem. We have further introduced the annotation engine and annotation rules to the database concept. QS Design consists of two stages., Annotation stage and Search stage as shown in Fig. 1 and 2.

Knowledge-Independent Components

QS is intended as a generic platform, we designed its components to be customizable to the specific domain of application. Therefore, a main concern in designing the platform was to limit its customization to domain-related aspects only. The design of QS distinguishes between domain knowledge and task knowledge. *Domain knowledge* describes all relevant entities in a specific domain of knowledge, representing a state of affairs and constraining the possible states it can evolve into. *Task knowledge*, in general, uses domain knowledge to describe relevant entities with respect to the required tasks (Blacoe *et al.,* 2010).

The only decisions that QS makes at platform level relate to the formalisms adopted for representing domain and task knowledge. A domain ontology needs a formalism that allows easy expression of taxonomical and nontaxonomical relationships among agents—static knowledge. A task ontology, on the other hand, must represent dynamic operations such as sequences, selections, and iterations that are necessary to represent tasks. The Semantic Web standard for representing ontologies is the Web Ontology Language, OWL.

Although OWL is adequate for modeling domain knowledge, it isn't suitable for representing dynamic operations. For these, we supplement OWL ontologies with rules represented using the Semantic Web Rule Language, SWRL.7,8 Such an extension is necessary, for example, to express part-whole relations;9 description logic, the representation formalism underlying OWL, isn't sufficiently expressive to formalize these relations. QS represents procedural knowledge, on the other hand, by mixing declarative rules with a traditional programming language (Java). It then represents tasks using clauses—a set of conjunctive premises and a single consequence, with the consequence represented by a block of executable code.

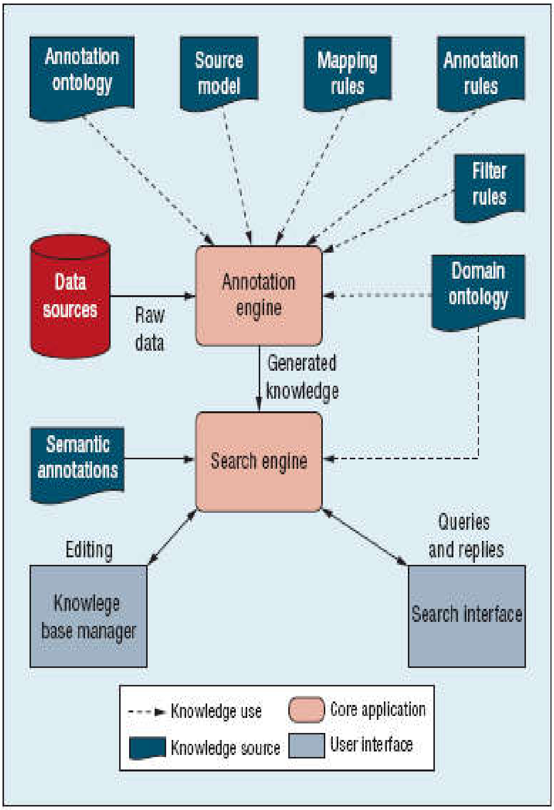

Fig. 1. Architecture: The annotation engine applies mapping and annotation rules to the database.

A) Annotation Engine

The annotation engine retrieves documents from their sources and then analyzes, annotates, and filters them on the basis of the application needs. Each of these functions is performed by a specific element that represents an implementation of one of the interfaces (harvester, analyzer, or semantic annotator). At this level of abstraction, QS separates task-specific knowledge and domain knowledge: The analyzer element poses only the task-specific knowledge available— for example, how to find relevant information on a Web page. The semantic annotator element uses domain knowledge to create the actual metadata. We obtain these independent components by leveraging the distinction between the knowledge needed for each functionality, so that changes in task or domain have an impact on only one component. Moreover, confining the task-specific knowledge to the analyzer system makes the search component completely independent of the way information is retrieved, easing the process of using multiple knowledge bases to answer users' queries.

B) Search Engine

The framework's search engine component queries the information generated by the annotation component. It accepts queries posed in SPARQL and returns a set of links to matching resources. A specialized search interface lets users develop an abstract model of a semantic query, pose it to the engine, and then review the resulting matched documents.

C) Search Interface

The search interface gives end users (people who aren't experts in Semantic Web technologies) a way to access the resources filtered and annotated by the semantic annotator component. It is also possible to add and delete entities and properties (with related values), so that a user can interact with the knowledge base to fine-tune the query, making subsequent searches more accurate.

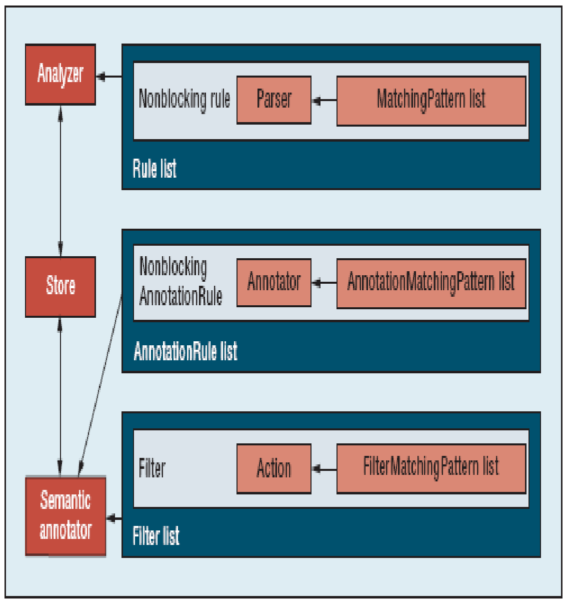

Figure 2. QS annotation components detailed architecture.

The key aim for the query interface is to give the user an intuitive and clear abstract query model that hides, as much as possible, the underlying complexity of representation and reasoning.

RESULT AND DISCUSSION

The agents in the search engine multi agent system exhibit various autonomic features that aim at making the system more robust and scalable. The QS system has been deployed in two different commercial test cases in the UK. In the first case, QS was used to examine specific Web-published documents for commercial opportunities matching the business interests of the customer company. In the second deployment, QS was used to perform knowledge-based searches over existing database sources

Conclusion

In evaluating the performance of the search system in both applications, It is observed that by using ontological knowledge and ontology-based annotations, users could perform more accurate queries while being returned up to 71 percent fewer documents than with a keyword-based search engine in the best cases eliminating more than 90 percent of the irrelevant documents. Further research is going on to refine these two deployments, and we are planning more industrial deployments in the near future

REFERENCES

Miller, G.A. 1995. Wordnet: A Lexical Database for English. *Comm. ACM*, 38(11): 39–41.

Uren *et al.,* 2006. Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art. *J. Web Semantics,* 4 (1): 14–28.

Berners-Lee, T., Hendler, J. and Lassila, O. 2001. The Semantic Web. *Scientific Am.,* 284(5): 34–43.

Studer, R., Benjamins, R. and Fensel, D. 1998. Knowledge Engineering: Principles and Methods. *J. ACM,* 25(1–2): 161–197.

Blacoe, I.W., Tamma, V. and Wooldridge, M.J. 2010. Evaluation of Scalable Multi-Agent System Architectures for Searching the Semantic Web," to be published in *Int'l J. Metadata, Semantics and Ontologies*.

Tümer, D., Shah, M. A. and Bitirim, Y. 2009. An Empirical Evaluation on Semantic Search Performance of Keyword-Based and Semantic Search Engines: Google, Yahoo, Msn and Hakia, 4thInternational Conference on Internet Monitoring and Protection (ICIMP '09) 2009.

Ramprakash *et al.,* 2008. Role of Search Engines in Intelligent Information Retrieval on Web", Proceedings of the 2nd National Conference; INDIACom-2008.

Madhu, G., Govardhan, A. and Rajinikanth, T.V. 2011. Intelligent Semantic Web Search Engines: A Brief Survey",International journal of Web & Semantic Technology (IJWesT) Vol.2, No.1, January 2011.

Inamdar1, S. A. and Shinde, G. N. 2008. An Agent Based IntelligentSearch Engine System for Web mining" Research, Reflections and Innovations in Integrating ICT in education.

Sanjib kumar, Sanjay kumar malik "TOWARDS SEMANTIC WEB BASED SEARCH ENGINES" National Conference on "Advances in Computer Networks & Information Technology (NCACNIT-09) March 24-25.

****