ISSN:0975-9646

CIS: Web Knowledge on Achievable Search Engine

P.Srilakshmi[1],M.Rajasekhar[2]

[1]*M.Tech (CSE) , Department of CSE,*
INDIRA PRIYADARSHINI ENGG COLLEGE FOR WOMEN, Dist:Kurnool, A.P, India
[2]*Assistant. Professor, Department of CSE,*
AVR&SVR ENGG COLLEGE, Dist:Kurnool, A.P, India

Abstract—The World Wide Web (WWW) allows the people to share the information (data) from the large database repositories globally. The amount of information grows billions of databases. We need to search the information will specialize tools known generically search engine. There are many of search engines available today, retrieving meaningful information is difficult. However to overcome this problem in search engines to retrieve meaningful information intelligently, semantic web technologies are playing a major role. In this paper we present survey on the search engine generations and the role of search engines in intelligent web and semantic search technologies.

Index Terms—Indexing ***Information retrieval, Intelligent Search, Search Engine, Semantic web.***

I. INTRODUCTION

The Semantic Web is an extension of the current Web [1] that allows the meaning of information to be precisely described in terms of well-defined vocabularies that are understood by people and computers. On the Semantic Web information is described using a new W3C standard called the Resource Description Framework (RDF). Semantic Web Search is a search engine for the Semantic Web. Current Web sites can be used by both people and computers to precisely locate and gather information published on the Semantic Web. Ontology [2] is one of the most important concepts used in the semantic web infrastructure, and RDF(S) (Resource Description Framework/Schema) and OWL (Web Ontology Languages) are two W3C recommended data representation models which are used to represent ontologies. The Semantic Web will support more efficient discovery, automation, integration and reuse of data and provide support for interoperability problem which can not be resolved with current web technologies. Currently research on semantic web search engines are in the beginning stage, as the traditional search engines such as *Google, Yahoo, and Bing (MSN)* and so forth still dominate the present markets of search engines.

Most of the search engines search for keywords to answer the queries from users. The search engines usually search web pages for the required information. However they filter the pages from searching unnecessary pages by using advanced algorithms. These search engines can answer topic wise queries efficiently and effectively by developing state-of art algorithms. However they are vulnerable in answering intelligent queries from the user due to the dependence of their results on information available in web pages. The main focus of these search engines is solving these queries with close to accurate results in small time using much researched algorithms. However, it shows that such search engines are vulnerable in answering intelligent queries using this approach. They either show inaccurate results with this approach or show accurate but (could be) unreliable results. To overcome this problem in search engines to retrieve relevant and meaningful information intelligently, semantic web technology deals with a great role [3]. Intelligent semantic technology gives the nearer to desired results by search engines to the user.

In this paper, we will make a preliminary survey over the existing literature regarding intelligent semantic search engines and semantic web search. By classifying the literature into few main categories, we review their characteristics respectively. In addition, the issues within the reviewed intelligent semantic search methods and engines are analyzed and concluded based on perspectives.

II. BACKGROUND

Current web is the biggest global database that lacks the existence of a semantic structure and hence it makes difficult for the machine to understand the information provided by the user. When the information was distributed in web, we have two kinds of research problems in search engine i.e.

_ How can a search engine map a query to documents where information is available but does not retrieve in intelligent and meaning full information?

_ The query results produced by search engines are distributed across different documents that may be connected with hyperlink. How search engine can recognize efficiently such a distributed results? Current web is the biggest global database that lacks the existence of a semantic structure and hence it makes difficult for the machine to understand the information provided by the user.

When the information was distributed in web, we have two kinds of research problems in search engine i.e.

_ How can a search engine map a query to documents where information is available but does not retrieve in intelligent and meaning full information?

_ The query results produced by search engines are distributed across different documents that may be connected with hyperlink. How search engine can recognize efficiently such a distributed results? Current web is the biggest global database that lacks the existence of a semantic structure and hence it makes difficult for the machine to understand the information provided by the user.

When the information was distributed in web, we have two kinds of research problems in search engine i.e.

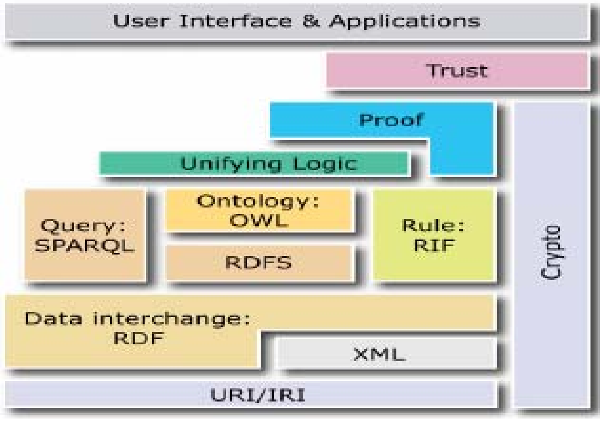
_ How can a search engine map a query to documents where information is available but does not retrieve in intelligent and meaning full information?

_ The query results produced by search engines are distributed across different documents that may be connected with hyperlink. How search engine can recognize efficiently such a distributed results? Current web is the biggest global database that lacks the existence of a semantic structure and hence it makes difficult for the machine to understand the information provided by the user.

When the information was distributed in web, we have two kinds of research problems in search engine i.e.

_ How can a search engine map a query to documents where information is available but does not retrieve in intelligent and meaning full information?

_ The query results produced by search engines are distributed across different documents that may be connected with hyperlink. How search engine can recognize efficiently such a distributed results?



Semantic Web Frame Work

2.1 Current Web & Limitations

Present World Wide Web is the longest global database that lacks the existence of a semantic structure and hence it becomes difficult for the machine to understand the information provided by the user in the form of search strings. As for results, the search engines return the ambiguous or partially ambiguous result data set; Semantic web is being to be developed to overcome the following problems for current web.

The web content lacks a proper structure regarding the representation of information.

Ambiguity of information resulting from poor interconnection of information.

Automatic information transfer is lacking.

Usability to deal with enormous number of users and content ensuring trust at all levels.

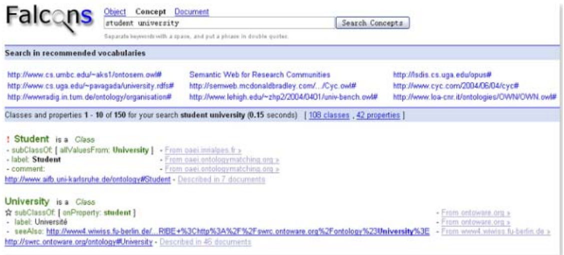
Incapability of machines to understand the provided information due to lack of a universal format.

III. Intelligent Semantic Web

3.1 Intelligent Search Engines

Currently, a couple of Intelligent search engines are designed and implemented for different working environments, and the mechanisms that realize these search engine are distinct.

Fu-Ming Hung and *Jenn-Hwa Yang* present an *intelligent search engine* with semantic technologies. This research has combine *description logic* inference system and digital library ontology to complete *intelligent search engine* [10]. According to search engine mechanism, presenting demands and a formula evaluating present related technology of that can solve and promote the efficiency of search engine, and formulating the demands of wisdom search engine. If uses *Description Logic Inference System* to integrate the digital library ontology to proceed with the inference of user requirement, and combines the content search mechanism and knowledge inference to accomplish the study of *intelligent search engine.*


Falcons Object Concept Document
student university Search Concepts
Separate keywords with a space, and put a phrase in double quotes.

Search in recommended vocabularies

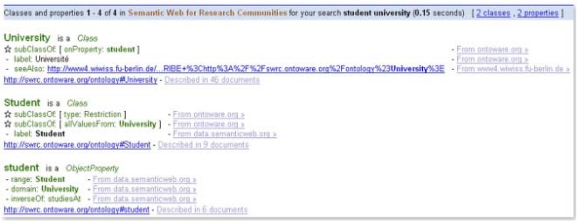
http://www.cs.umbc.edu/~aks1/ontosem.owl#
http://www.cs.uga.edu/~pavagada/university.rdfs#
http://wwwradig.in.tum.de/ontology/organisation#
Semantic Web for Research Communities
http://semweb.mcdonaldbradley.com/.../Cyc.owl#
http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#
http://lsdis.cs.uga.edu/opus#
http://www.cyc.com/2004/06/04/cyc#
http://www.loa-cnr.it/ontologies/OWN/OWN.owl#

Classes and properties 1 - 10 of 150 for your search student university (0.15 seconds) [108 classes , 42 properties]

! Student is a Class
- subClassOf: [allValuesFrom: University] - From oaei.inrialpes.fr »
- label: Student - From oaei.ontologymatching.org »
- comment: - From oaei.ontologymatching.org »
http://www.aifb.uni-karlsruhe.de/ontology#Student - Described in 7 documents

University is a Class
☆ subClassOf: [onProperty: student] - From ontoware.org »
- label: Université - From ontoware.org »
- seeAlso: http://www4.wiwiss.fu-berlin.de/...RIBE+%3Chttp%3A%2F%2Fswrc.ontoware.org%2Fontology%23University%3E - From www4.wiwiss.fu-berlin.de »
http://swrc.ontoware.org/ontology#University - Described in 46 documents

Fig. 1. First result page for the keyword query “student university.”


Classes and properties 1 - 4 of 4 in Semantic Web for Research Communities for your search student university (0.15 seconds) [2 classes , 2 properties]

University is a Class
☆ subClassOf: [onProperty: student] - From ontoware.org »
- label: Université - From ontoware.org »
- seeAlso: http://www4.wiwiss.fu-berlin.de/...RIBE+%3Chttp%3A%2F%2Fswrc.ontoware.org%2Fontology%23University%3E - From www4.wiwiss.fu-berlin.de »
http://swrc.ontoware.org/ontology#University - Described in 46 documents

Student is a Class
☆ subClassOf: [type: Restriction] - From ontoware.org »
☆ subClassOf: [allValuesFrom: University] - From ontoware.org »
- label: Student - From data.semanticweb.org »
http://swrc.ontoware.org/ontology#Student - Described in 9 documents

student is a ObjectProperty
- range: Student - From data.semanticweb.org »
- domain: University - From data.semanticweb.org »
- inverseOf: studiesAt - From data.semanticweb.org »
http://swrc.ontoware.org/ontology#student - Described in 6 documents

Fig. 2. First result page for the keyword query “student university” after the SWRC ontology is selected to filter the results.

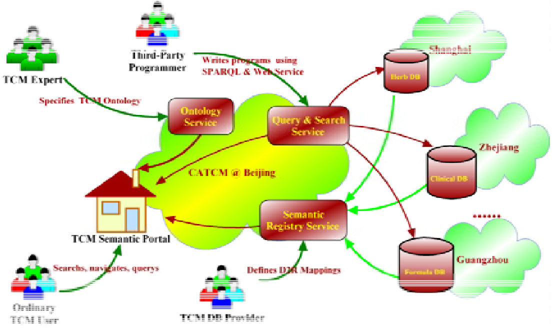


Fig. 3. Architecture of Cranes Intellection Search.

Each RDF triple in an RDF document and the document URI form a quadruple and is stored in the quadruple store implemented based on the MySQL database. The meta analysis component periodically computes several kinds of global information and updates them to the meta-data data base, e.g., which kind of entity (class/property/individual) a

URI identifies and which Intellection s an ontology contains. Then, also periodically, the indexer updates a combined inverted index, which serves the proposed mode of user interaction, i.e., keyword search with ontology restriction. This combined index consists of two inverted indexes implemented based on Apache Lucerne (lucene.apache.org). First, for each Intellection , a virtual document [4] is constructed, which consists of the terms extracted from its RDF description (cf. Section III). An inverted index, as a classic information retrieval data structure, is built from terms in virtual documents to Intellection s, to serve keyword search. Second, based on the metadata database, an inverted index is built from Knowledge to the Intellection s they contain, to serve ontology-based result filtering. Thus, for a keyword query with an ontology restriction, the Intellection s finally returned are obtained by performing the intersection operation on the two result sets separately returned by these inverted indexes. The ranking process (cf. Section IV-A) is also implemented based on Lucene. At indexing time, a popularity score is computed and attached to each Intellection.

IV. RANKING

A. Intellection Ranking

In the system, the ranking score of a Intellection is concerned with two factors, i.e., its relevance to the keyword queryq and its popularity

RankingScore(c,q)=TextSim(c, q)·Popularity(c) (1)

which will be separately discussed in the following.

1. Query Relevance: On the one hand, as described in Section III, a virtual document is constructed for each Intellection . On the other hand, a keyword query can be treated as a short document. Thus, the problem of calculating the relevance of a Intellection to a keyword query could be transformed into the problem of calculating similarity between two documents.

We use the vector space model and the term frequency weight to represent documents, i.e., each document is represented as a vector where each component corresponds to the frequency of a term in the document. In particular, the weights of the terms extracted from the local name and label of the Intellection in question are additionally multiplied by 10.0, based on our previous experience of using virtual documents in ontology matching [6]. Then, weights are further refined by the well-known inverse document frequency measure, i.e., a higher weight is assigned to a term in a virtual document if the term occurs in fewer documents in the whole data set because such a term is considered to be a more distinctive feature. Finally, the relevance of a Intellection to a keyword query, TextSim(c, q), is defined as the cosine of the angle between the vector form of the virtual document ofcand the vector form ofq.

2. Popularity:Other than query relevance, existing approaches study ontology structures to evaluate Intellection s with structural measures such as Page Rank-like algorithms [7], [8] or graph centrality [9].

However, they failed to investigate the use of Intellection s in practice. To develop a new Web application, in order to maximize the interoperabil-ity of different applications, one best practice is to reuse Intellection s that have been widely used by existing applications. Therefore, our system gives higher ranks to popular Intellection s.

For a Intellection , let Docs(c)be the set of RDF documents where cis instantiated. A Intellection cis instantiated in an RDF document d if either cis a class and contains an RDF triple whose predicate is rdf:typeand whose object isc,orcis a property anddcontains an RDF triple whose predicate isc. The popularity score ofcis calculated as follows:

Popularity(c)=log(|Docs(c)|+1)+1. (2)

In the system, popularity scores are evaluated based on a large data set collected from the real Semantic Web, which includes not only Intellection ual-level RDF documents (Knowledge) but also a lot of instance-level RDF documents. Therefore, it is possible to characterize the use of Intellection s in practice.

B. Ontology Recommendation

In the system, according to the proposed mode of user interaction, several Knowledge are recommended to be selected to filter the con-cepts returned. In Section IV-A, we have detailed the principle and method of ranking Intellection s. Now, we rank Knowledge based on the ranking of Intellection s.

For a keyword query, the Knowledge that the Intellection s returned come from are regarded as candidates for recommendation. For each ontology candidate, its ranking score is evaluated by adding up the ranking scores of those Intellection s returned and contained in this on-tology. Finally, up to nine top-ranking Knowledge are recommended. The underlying criterion is that an ontology is more likely to be recommended if the Intellection s in the ontology that are matched with the terms in the keyword query are more popular on the Semantic Web.

V. USER EVALUATION AND LESSON LEARNED

5.1 Feedbacks from CATCM

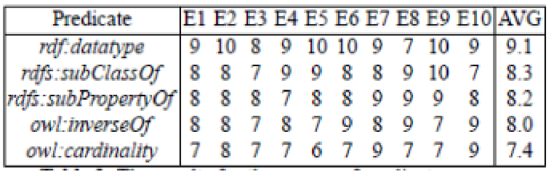
The first proof-of-concepts prototype was deployed during fall 2004. By using that prototype, we convinced CATCM partner to take the semantic web technologies to help them in managing their fast increasing TCM databases. After a thorough requirements analysis and with a careful redesign and re-engineering of the entire system, a more stable and user-friendly version was released in September 2005, and deployed at CATCM for open evaluation and real use.

Currently, the system deployed at CATCM provides access to over 70 databases including TCM herbal medicine databases, TCM compound formula databases, clinical symptom databases, traditional Chinese drug database, traditional Tibetan drug database, TCM product and enterprise databases, and so on. The TCM shared ontology includes over 70 classes, 800 data or object properties.

5.2 A Survey on the Usage of RDF/OWL Predicates

RDF/OWL has offered us a range of predicates, but not all of them are useful for relational data integration. We made a survey on the usage of RDF/OWL predicates for relational database integration, and the results are indicated in table 1.

In this survey, we invited ten developers who are familiar with both semantic web technologies and our system. They are asked with the same questions: “From a practical view, what are those most important constructs do you think for relational data integration in semantic web", and are requested to write down some explanation for the reason of their choice. We summarize their comments and the score result as follows.



Predicate	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	AVG
rdf:datatype	9	10	8	9	10	10	9	7	10	9	9.1
rdfs:subClassOf	8	8	7	9	9	8	8	9	10	7	8.3
rdfs:subPropertyOf	8	8	8	7	8	8	9	9	9	8	8.2
owl:inverseOf	8	8	7	8	7	9	8	9	7	9	8.0
owl:cardinality	7	8	7	7	6	7	9	7	7	9	7.4

Table 1. The results for the survey of predicates usage.

VI. RELATEDWORKS

6.1 Semantic Web Context

In the Semantic Web community, semantic data integration has been always a noticeable research topic. In particular, there have been a number of works dealing with how to make contents of existing or legacy database available for semantic web applications.

A typical one is D2RQ7. D2RQ is a declarative language to describe mappings between relational database schemata and OWL/RDFS ontologies, and is implemented as a Jena plugin that rewrites RDQL queries into SQL queries. The result sets of these SQL queries are transformed into RDF triples that are passed up to the higher layers of the Jena framework. RDF Gateway8 is a commercial software having similar functionalities. It connects legacy database resources to the SemanticWeb via its *SQL Data Service Interface*. The *SQL Data Service* translates a RDF based query to a SQL query and returns the results as RDF data. Our system is different from D2RQ and RDF Gateway. We take the view-based mapping approach which has sound theoretical foundation, and we have visualized mapping tool and ontology-based query and search tool which are not offered by these two systems.

For other related works, Dejing Dou and colleagues [8] propose an ontology-based framework called OntoGrate. It can automatically transform relational schema into ontological representation, and users can define the mappings at the ontological level using bridge-axioms. Francois [9] considers theoretic aspect of answering query using views for semantic web and Peter Haase and Boris Motik introduces a mapping system for OWL-DL ontology integration

6.2 Conventional Data Integration Context

Without considering the semantic web technologies, our solution can be categorized to the topic "answering query using view", which has been extensively studied in database community [2] [11]. Most previous works has been focused on the relational case [2], and XML case [12]. On the one hand, we believe it would be valuable for the semantic web community to take more consideration of the techniques that have been well studied in the database community such as answering query using view. On the other hand, we think that the semantic web research does raise a lot of new issues and challenges for database researchers. From our experiences, the challenges include: From our experiences, the challenges include: how to rank the data object just like the page rank of google? How to maintain highly evolvable and changeable schema mappings among an great number of and open-ended set of databases with no centralized control?

Moreover, a lot of works have been done in the area of ontology-based data integration [13]. Many of them took some ontological formalism such as DL to mediate heterogenous databases, and used the view-based mapping approach. In comparison with them, our implementation is the case of RDF/OWL-based relational data integration with a *semantic web vision in mind*.

CONCLUSION AND FUTURE ENHANCEMENTS

In this paper, we presented an in-use application of Traditional Chinese Medicine enhanced by a range of semantic web technologies, including RDF/OWL semantics and reasoning tools. The ultimate goal of this system is to realize the "web of structured data" vision by semantically interconnecting legacy databases, that allows a person, or a machine, to start in one database, and then move around an unending set of databases which are connected by rich semantics. To achieve this demanding goal, a set of convenient tools were developed, such as visualized semantic mapping tool, dynamic semantic query tool, and intuitive search tool with concepts ranking. Domain users from CATCM indicated that the system provided an amazing solution for the semantic heterogeneity problem troubling them for a long time.

Currently, although this project is complete, several updated functionalities are still in our consideration. To be specific, we are going to enhance the mapping tools with some heuristic rules to automate the mapping task as far as possible, just like the approach proposed by Yuan An and colleagues [5]. Otherwise, we will develop a more sophisticated mechanism to rank the data objects just like the page rank technology provided by popular search engines.

REFERENCES

[1] L. Ding, T. Finin, A. Joshi, Y. Peng, R. Pan, and P. Kolari, "Search on the semantic web,"IEEE Comput., vol. 38, no. 10, pp. 62–69, Oct. 2005.

[2] M. d'Aquin, C. Baldassarre, L. Gridinoc, M. Sabou, S. Angeletou, and E. Motta, "Watson: Supporting next generation semantic web applica-tions," inProc. IADIS Int. Conf. WWW/Internet, 2007, pp. 363–371.

[3] C. Anutariya, R. Ungrangsi, and V. Wuwongse, "SQORE: A framework for semantic query based ontology retrieval," in Proc. 12th Int. Conf. Database Syst. Adv. Appl., 2007, pp. 924–929.

[4] C. Watters, "Information retrieval and the virtual document,"J. Amer. Soc. Inf. Sci., vol. 50, no. 11, pp. 1028–1029, Aug. 1999.

[5] Nokia, P. Stickler, CBD—Concise Bounded Description. [Online]. Avail-able: http://sw.nokia.com/uriqa/CBD.html

[6] Y. Qu, W. Hu, and G. Cheng, "Constructing virtual documents for ontology matching," inProc. 15th Int. World Wide Web Conf., 2006, pp. 23–31.

[7] X. Zhang, H. Li, and Y. Qu, "Finding important vocabulary within ontol-ogy," inProc. 1st Asian Semant. Web Conf., 2006, pp. 106–112.

[8] G. Wu, J. Li, L. Feng, and K. Wang, "Identifying potentially important Intellection s and relations in an ontology," inProc. 7th Int. Semant. Web Conf., 2008, pp. 33–49.

[9] H. Alani and C. Brewster. Metrics for Ranking Ontologies. in 15th International Conference for World Wide Web. 2006. Edinburgh, UK