

# A Case Study on Semantic Web Search using Ontology Modeling

Monica Shekhar<sup>1</sup> and Saravanaguru RA. K.<sup>2</sup>  
School of Computing Science and Engineering, VIT University  
Vellore, Tamil Nadu - 632014, India  
monicashekhar03@gmail.com, sarophd@gmail.com

## ABSTRACT

The Semantic Web works on the existing Web which presents the meaning of information as well-defined vocabularies understood by the people. Semantic Search, at the same time, works on improving the accuracy of a search by understanding the intent of the search and providing contextually relevant results. The paper describes a semantic approach towards web search through a PHP application. The goal was to parse through a user's browsing history and return semantically relevant web pages for the search query provided. The browser used for this purpose was Mozilla Firefox. The user's history was stored in a MySQL database, which, in turn, was accessed using PHP. The ontology, created from the browsing history, was then parsed for the entered search query and the corresponding results were returned to the user providing a semantically organized and relevant output.

KEYWORDS: protégé, python, Semantic Web, Semantic Search, ontology, RDF, RDFS, OWL

## I. INTRODUCTION

Over the past years, the ever-increasing growth in the World Wide Web has brought into light the insufficiency of the currently existing techniques used for searching information on the web [2]. For a given query entered by the user, search for a target web page in most search engines is based on keyword-based searches and popularity based ranking. Although the results might be good enough, not all the search results turn out to be relevant to the given query. What is lacking in these search engines is a semantic structure, relationships between the information available over the web, making it difficult for the machine to understand the information asked for by the user and resulting in the loss of critical information while searching [3]. Therefore, semantic knowledge plays an important role.

The Semantic Web is, necessarily, a vision for the future of the World Wide Web where the available information is given a meaning, providing logical connections of terms and making it easier of the machine to integrate data and process the information available on the Web [8]. Semantics is the study of meaning. For the search query entered by the user, a Semantic Web Search ensures contextually relevant results by understanding intent and meaning of the query provided.

With the need for Semantic Web, W3C defined the first Resource Description Framework (RDF) specification for semantic interoperability in 1997. RDF required triple-based representations for Universal Resource Identifiers (URIs). Expression of structured vocabularies was then introduced in RDF Schema (RDFS). Web Ontology Language (OWL) provided greater expressivity in the objects and relations of the RDFS [4]. These ontologies provided a strong semantic structure to the data.

In the paper published by Doms A. and Schroeder M. in 2002, the first semantic search engine for biomedical texts using the Gene Ontology was published [5]. The Gene Ontology is a hierarchically structured vocabulary for molecular biology. Sara Cohen Jonathan Mamou et al, in 2003, presented a semantic search engine for XML called XSearch [6]. It provided semantically related document fragments in accordance to the user's query. Later in 2004, Li Ding Tim Finin et al introduced a prototype Semantic Web search engine in a research project named Swoogle [7]. It is a search engine for the Semantic Web on the Web. Swoogle is a crawler-based indexing and retrieval system that searches for Semantic Web documents, instance data, terms, ontologies, etc. published on the Web.

These semantic search engines are designed to search for information in the World Wide Web. The objective of this research is to provide the user with semantically relevant results for the entered search query based on the browsing history of the user. The browser taken under study for this purpose was Mozilla Firefox. The browsing history of the user from the web browser was accessed and stored in MySQL database. The browsing domain considered in the research was "Apple". For all the URLs visited by the user under the considered domain, an ontology was created. For the query entered by the user, the created ontology was parsed and the visited URLs were displayed providing a semantically relevant list of URLs.

## II. SEMANTIC WEB LANGUAGE

The semantic web languages used in this application are RDF, RDFS and OWL [8]-[10]. The Resource Description Language is a general-purpose triple-based language used for representing information in the Web. The triples in RDF are represented as subject-predicate-object. RDF Schema is a semantic extension over RDF, providing vocabulary descriptions over the triples-based RDF. Web Ontology Language is used to make information available to be processed by applications, where the meaning of each term and their inter-relationships are explicitly represented. The representation of the terms and the relationship between these terms is called an ontology. Due to the greater expressiveness of OWL, this language has the ability to represent machine interpretable content on the Web [8].

The various constructs in the OWL language, mostly used in the ontology created in the application, are the RDF Schema features (`rdf:subClassOf`, `rdf:Property`, etc.) for defining the classes, subclasses, properties, sub-properties, etc., (In)Equality (`differentFrom`, `distinctMembers`, etc.) for specifying the inequalities between the various individuals, Property Restrictions (`Restriction`, `onProperty`, `allValuesFrom`, `someValuesFrom`, etc.) for defining restrictions on the individuals and Annotation Properties (`rdfs:label`, `rdfs:comment`, `AnnotationProperty`, etc.) for specifying the details for each individual [9].

## III. ARCHITECTURE

The research had been carried out in three modules:

- Web Content Extraction (Module 1)
- Semantic Knowledge Base (Module 2)
- Reasoner (Module 3)

The architecture of the system, in accordance to the modules, is shown below.

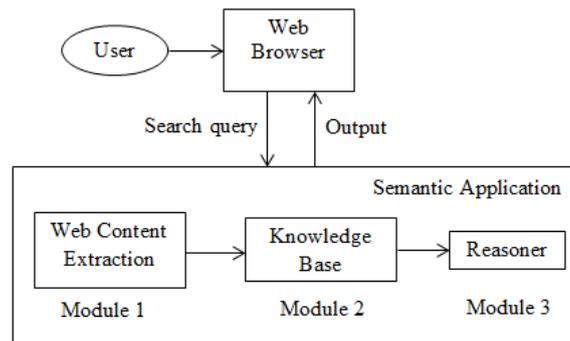


Figure 1 Architecture of the system

Here, the user enters the search query into the semantic application. The semantic application in itself consists of the three modules, wherein, the browsing history of the user was contained in module 1, the ontology created in accordance to the browsing history was contained in module 2 and module 3 parsed through the created ontology to return the result. This result was then displayed to user through the semantic application on the browser.

## IV. METHODOLOGY

### 1. Web Content Extraction

The coding for this module was done using python. Also, the browser under consideration was Mozilla Firefox. Mozilla Firefox stores all its data into a SQLite database. The following figure shows the SQLite Manager that was used to access the browsing history from the Mozilla Firefox web browser.

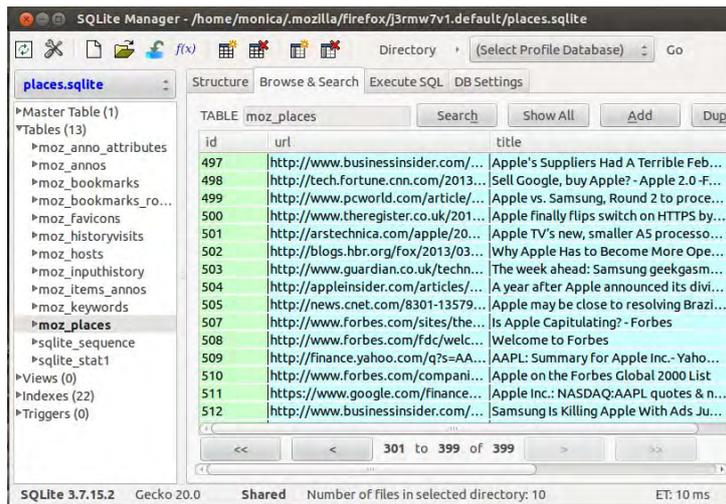


Figure 2 The sqlite manager displaying the database used to store the user's browsing history

From the *moz.places* database, the user's browsing history was accessed and stored into a MySQL database using python, the python packages used being *sqlite3* and *MySQLdb*. This is displayed in the figure as shown below.

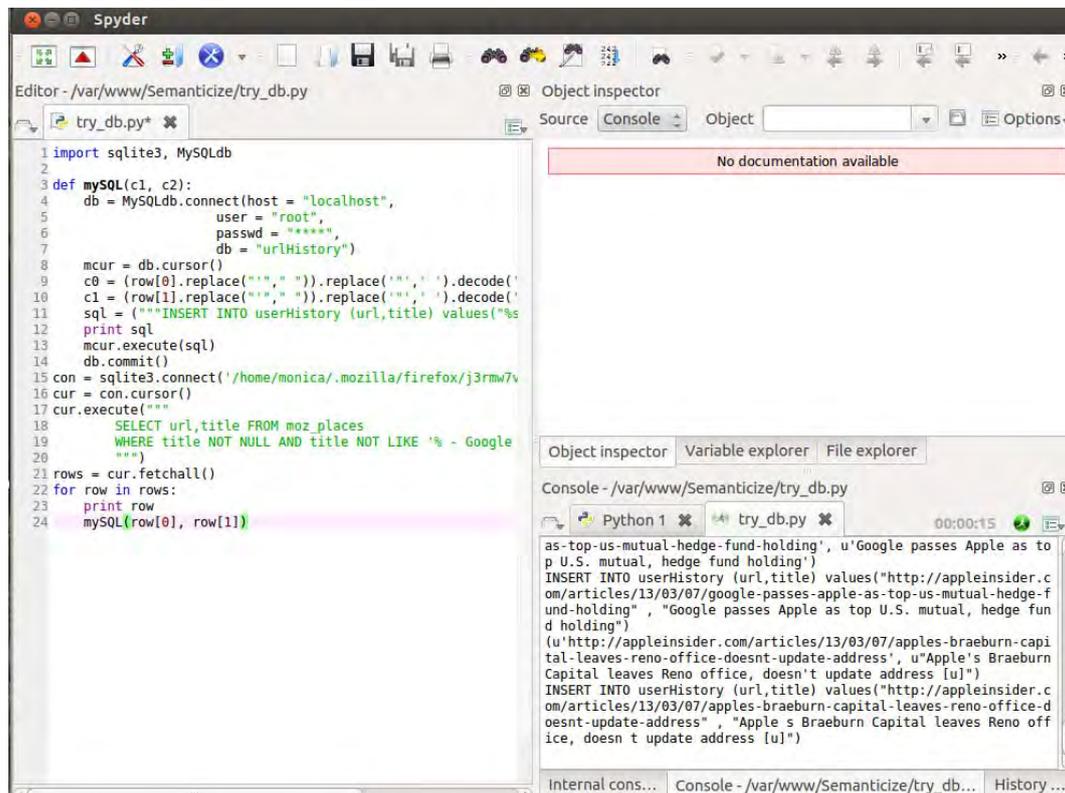


Figure 3 The python code that runs in the background, storing the table values from sqlite3 to MySQL

The above python code selects values from the *sqlite3* database and stores it into a *MySQL* database, to be accessed in the *PHP* application. This python code was made to run in the background by adding a scheduler (the *sched* package) for the code to execute every 15mins. The web pages visited, in the meantime, would get updated into the *MySQL* table in intervals of 15mins.

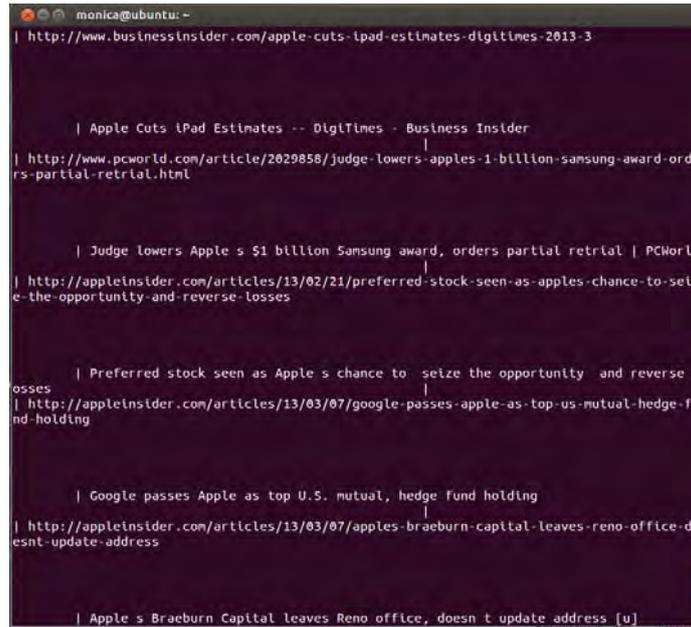


Figure 4 MySQL table displaying the URLs and the title as accessed from the sqlite database

Other python packages like BeautifulSoup4 and NLTK were used to extract the details of the visited URLs, like the title, description, etc. from the Web and were in turn stored into the MySQL database.

## 2. Semantic Knowledge Base

The semantic knowledge base creation was done using the *Protégé-OWL editor*. Since the knowledge base domain, considered here, was “Apple”, the ontology so created consisted of subclasses *AppleInc* (for the company) and *AppleFruit* (for the fruit). The various URLs obtained from Module 1 were added into the ontology as individuals for the corresponding classes, specifying the various schema features, OWL property restrictions, annotation properties, etc. [8][9].

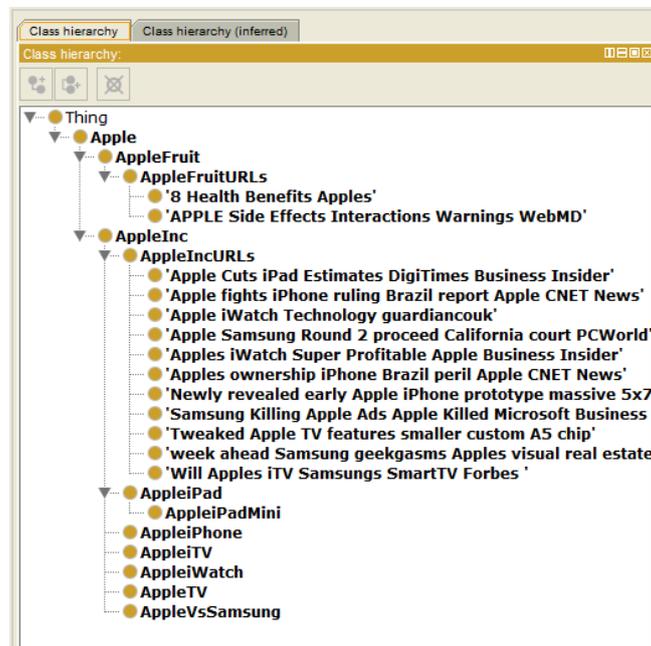


Figure 5 The classes and subclasses in the ontology

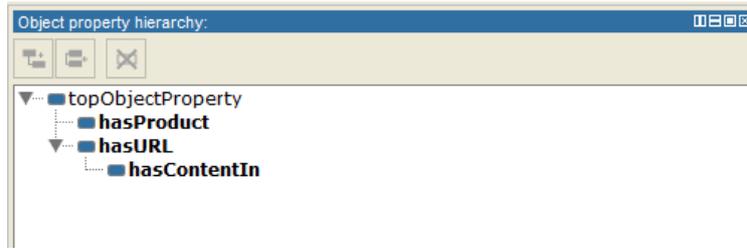


Figure 6 The object properties and sub-properties of the ontology

The property restriction allValuesFrom was used to differentiate between the URLs of the two Apple subclasses, where the subclasses AppleInc and AppleFruit were defined to be disjoint with each other. The various restrictions applied on the classes in the ontology is shown in the figure below.

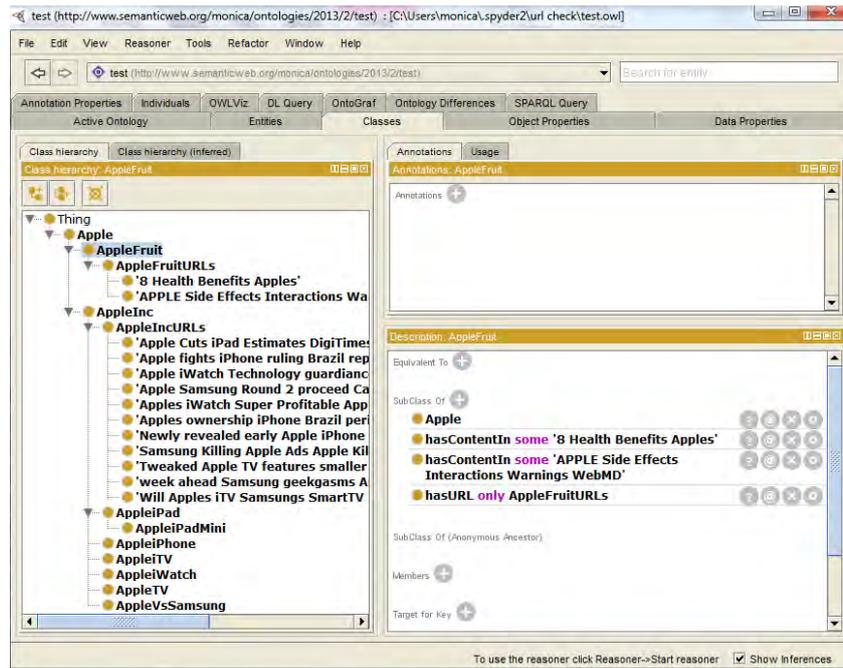


Figure 7 The restrictions applied on the classes in the ontology

### 3. Reasoner

The ontology was queried using DL query, which displays the corresponding classes, subclasses, individuals, etc. pertaining to the query. This query retrieved information from the ontology as created in the previous module.

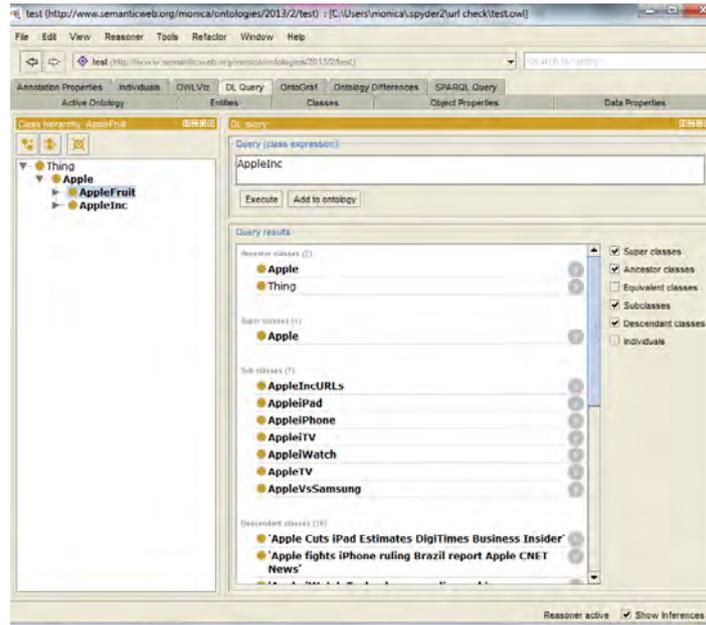


Figure 8 Querying the created semantic knowledge base using DL query

This ontology, saved as an OWL file, was accessed, parsed and queried in the PHP application, where the query entered by the user was passed as the query to the file. The result so obtained was displayed by the application, providing a semantically relevant list of URLs for the entered search query from the user’s list of visited URLs. The semantic application was provided with user login, thereby keeping the application and the database more user-specific.

**V. RESULT AND FURTHER WORK**

The application, hence, aids the search of the user by providing a list of useful and relevant web pages earlier visited by him, in accordance to the query, thereby providing him with results that would be relevant and helpful. The following table shows some sample input and output values for the application.

INPUT	OUTPUT
<b>Apple iPad</b>	<a href="http://news.cnet.com/8301-13579_3-57569172-37/apple-loses-iphone-trademark-in-brazil-report/">http://news.cnet.com/8301-13579_3-57569172-37/apple-loses-iphone-trademark-in-brazil-report/</a> - Apple loses iPhone trademark Brazil report Apple CNET News
	<a href="http://appleinsider.com/articles/13/03/11/newly-revealed-early-apple-iphone-prototype-had-massive-5x7-inch-display">http://appleinsider.com/articles/13/03/11/newly-revealed-early-apple-iphone-prototype-had-massive-5x7-inch-display</a> - Newly revealed early Apple iPhone prototype massive 5x7inch display
<b>Apple fruit</b>	<a href="http://www.nutrition-and-you.com/apple-fruit.html">http://www.nutrition-and-you.com/apple-fruit.html</a> - Apple fruit nutrition facts health benefits
	<a href="http://www.huffingtonpost.com/2012/09/06/health-benefits-apples_n_1855590.html">http://www.huffingtonpost.com/2012/09/06/health-benefits-apples_n_1855590.html</a> - 8 Health Benefits Apples

Table 1 Sample input and output values for the application

The major area of further work in the system would be to make the application work for web browsers other than Mozilla Firefox. Accessing databases of Google Chrome, Safari, Internet Explorer, etc. and storing them in the MySQL database along with the existing database would make the application work for the other web browsers as well.

Also, to make the application more accessible to the user, further work can be done on the system by deploying it as a plugin to be installed into the web browser, providing semantically organized results to the user simultaneously with the current searches being made.

## VI. CONCLUSION

With the increasing amount of data being stored on the Web every day, Semantic Search would ensure the provision of useful and relevant information to the user. This application works on making the procedure more user-friendly by providing contextually relevant visited pages to the user along with the searches available to the user otherwise through the various search engines. Understanding the user's query and providing just the content required is the objective of the application.

## REFERENCES

- [1] Tim Berners-Lee, James Hendler and Ora Lassila, "The Semantic Web," *Scientific American*, Feature Article, May 2001.
- [2] Max J. Egenhofer, "Toward the Semantic Geospatial Web," in *Proc. GIS'02*, 2002.
- [3] G. Madhu, Dr. A. Govardhan and Dr. T. V. Rajinikanth,, "Intelligent Semantic Web Search Engines: A Brief Survey," in *Proc. ICIET'10*, 2010.
- [4] Nigel Shadbolt, Wendy Hall and Tim Berners-Lee, "The Semantic Web Revisited," *IEEE Intelligent Systems*, vol. 21, pp. 96-101, Jan.-Feb. 2006.
- [5] Doms A. and Schroeder M., "GoPubMed: exploring PubMed with Gene Ontology," *Nucleic Acids Res.*, 33(Web Server issue):W783-6, Jul. 2005.
- [6] Sara Cohen, Jonathan Mamou, Yaron Kanza and Yehoshua Sagiv, "XSEarch: A Semantic Search Engine for XML," in *Proc. VLDB'03*, 2003.
- [7] Li Ding, Tim Finin, Anupam Joshi, Rong Pan, R. Scott Cost, Yun Peng, Pavan Reddivari, Vishal Doshi and Joel Sachs, "Swoogle: A Search and Metadata Engine for the Semantic Web," in *Proc. CIKM'04*, 2004.
- [8] OWL Web Ontology Language Overview: <http://www.w3.org/TR/owl-features/>
- [9] OWL Web Ontology Language Reference: <http://www.w3.org/TR/owl-ref/>
- [10] RDF Vocabulary Description Language 1.0: RDF Schema: <http://www.w3.org/TR/rdf-schema/>