Comparative Study on Semantic Search Engines

Ranjna Jain
Research Scholor
YMCAUST
Faridabad, India

Neelam Duhan, PhD
Assistant Professor
YMCAUST
Faridabad, India

A.K. Sharma, PhD
Prof. & Dean
BSAITM
Faridabad, India

ABSTRACT
Current World Wide Web also recognized as Web 2.0 is an immense library of interlinked documents that are transferred by computers and presented to people. Search engine is considered the most important tool to discover any information from WWW. Inspite of having lots of development and novel research in current search engines techniques, they are still syntactic in nature and display search results on the basis of keyword matching without understanding the meaning of query, resulting in the production of list of WebPages containing a large number of irrelevant documents as an output. Semantic Web (Web 3.0), the next version of World Wide Web is being developed with the aim to reduce the problem faced in Web 2.0 by representing data in structured form and to discover such data from Semantic Web, Semantic Search Engines (SSE) are being developed in many domains. This paper provides a survey on some of the prevalent SSEs focusing on their architecture; and presents a comparative study on the basis of technique they follow for crawling, reasoning, indexing, ranking etc.

Keywords
Web 2.0, Semantic Web, Semantic Search Engines

1. INTRODUCTION
The World Wide Web (WWW) is regarded as the largest human information construct in history and in order to retrieve information from the information space, search engines are used. The web is commonly understood to have had three overlapping phases [2] of development and with these eras, role of search engines has also changed. Under web 1.0, the purpose of search engine such as World Wide Web Worm (WWWW) was purely on determining the size of the web and content relevance was ignored. Because of the limited resources, their indexing and hence searching were limited to the titles and headings found in the web pages. While web 2.0 search engines such as google was considered as a web of documents which retrieve those documents that contain keywords in that corresponding query. Due to the unstructured behaviour of information in the web page, user still had to mine his required information from the documents which were retrieved by the search engine on the basis of keywords. Therefore, it was not successful in providing the actual required information to the user.

Whereas in web 3.0 which is also named as semantic web, semantic search engines has web of data where data is represented with triple which contain <subject- predicate-object>. With this form, data is tried to be interlinked so that data consumer can discover more information. It tries to provide required information to the user directly so that user does not need to explore into the displayed document as in the case of web 2.0 search engines. The rest of the paper is organized as follows: Section 2 describes Web 2.0 search methods; section 3 introduces semantic web and its architecture; section 4 describes the existing semantic search engine's architecture; section 5 dictates the comparison study performed on the discussed semantic search engines and finally, Section 6 concludes the paper.

2. RELATED WORK
The category of Search Engines includes Google, directories like DMOZ, yahoo etc and Meta-Search Engines such as dogpile. They are very popular in spite of the fact that they do not provide exact results.

2.1 Search Engine
A Search Engine [3] is a program designed to search for information on the WWW. The search results presented in a list consist of web pages, images, information and other types of files. The architecture of a general search engine contains a front-end process and a back-end process, as shown in Figure 1.

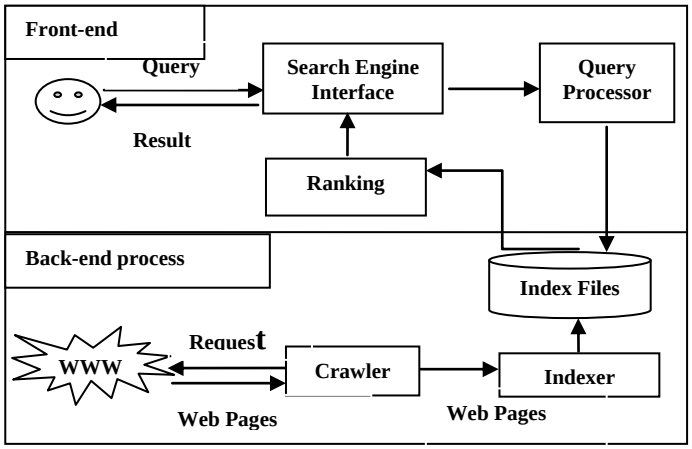

Figure 1. Architecture of General Search Engine

In the front-end side, user submits the search query to the search engine interface. The query processor then parses the search request into a form that the search engine can understand, and then the search engine executes the search operation on the index files. After ranking, the search results are returned to the user. In the back-end, the crawler module (spider or robot) fetches the web pages from the Web; the indexing subsystem parses those Web pages and stores them into the index files.

Despite of the fact that WWW contains a lot of information and knowledge, search engines usually serve only to deliver and present the content of documents describing the knowledge. Apart from this, there exist other problems that users are suffering from, which are discussed as follows:

- Current search engines are unable to provide direct answer to queries.

- Current search engines process queries based on keywords. Thus, retrieve all web pages containing those keywords without considering the fact that an accurate answer is produced on the basis of user's context.
- Current search engines are unable to gather complex information.
- Current WWW contains a lot of information and knowledge, but current search engines are unable to retrieve complex information. For instance, user fires a query "find 10 engineering college for computer stream in india and the top computer companies in their close proximity". Current search engines would not be able to yield desired results. For the results, user has to separately fire the query and manually merge the results.
- Current Search Engines are handicapped by being unable to figure out the context in which a word is being used.
- Although the search engines are very helpful in finding information on the Internet and are getting smarter with the passage of time, but they lack in finding the meanings of the terms, expressions used in the Web pages and the relationships between them. The problem comes due to the existence of words which have many meanings also known as polysemy and several words having same meaning also known as synonymy in natura1 languages. Thus, when a user gives a search query like "Flip-Flop" to find the definition of "Flip-Flop" in Computer Science domain, the most accredited search engine, Google, is unable to find the right document (no document is relevant among the top ten results returned). ' This is because Google does not know which Flip-Flop user is talking about; a kind of female shoes , or a device for Electronics for used one bit memory storage. It was possible for Google to find the right document only if it knew the relationship between the two terms given to it; "Flip-Flop" and "Electronics".

2.2 Directory

A Web Directory [4] organizes Web sites by subject, and is usually maintained by humans instead of software. The searcher looks at sites organized in a series of categories and menus. It does not display results in the form of web pages based on keywords. The results of directory are in the form of links that contains category and sub categories. The database size of directory is smaller as compared to engines' databases, it is human-sited directory and not crawled by crawlers. One of the famous directories, 'The Open Directory' has been around since 1999, and is a human-edited directory. Also known as DMOZ (Directory Mozilla), the Open Directory Project proposed to be the "largest on the Web", constructed and maintained by a vast, global community of volunteer editors. Directory tends to work best when the user want to browse a relatively broad subject. Starting with, a directory can give a good idea about the amount and type of web based information on user's desired topic.

2.3 Meta-Search Engines

A Meta-Search engine [5] performs a search by calling on more than one search engine to do the actual work. The general architecture of Meta-Search engine is shown in Figure 2 where it sends user requests to several other search engines and/or databases and aggregates the results into a single list and displays them according to their source.

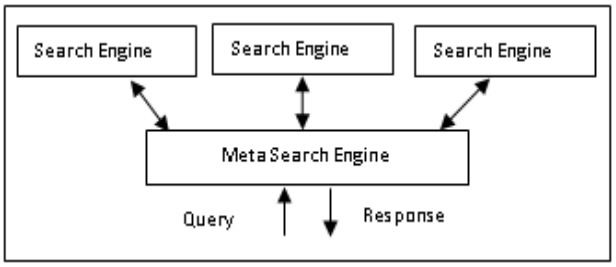

Figure 2. Architecture of a General Meta-Search Engine

Meta-Search Engines enable users to enter search criteria once and access several search engines simultaneously. Meta-Search engines operate on the premise that the Web is too large for any one search-engine to index it all and that more comprehensive search results can be obtained by combining the results from several search engines. This also may save the user from having to use multiple search engines separately. However, it is experienced by the end user that results are not relevant and thus, he keeps himself navigating within the search results for a long time.

To deal with such problem, Berners-Lee, Hendler and Lassila [6] presented a vision of a Web in which information is given well-defined meaning, better enabling computers to understand the meaning of content and help people to provide relevant information which is called Semantic Web. The next section discusses about Semantic Web.

3. SEMANTIC WEB

Current web contains millions of unstructured web documents which are accessed using search engines such as google, bing etc. but these search engines do not satisfy user's expectation because they display a list of documents which matches the terms present in the fired query. They are not concerned with the fact whether they yield the user's required information or not. It is because web documents are unstructured in nature due to which contents are analysed syntactically and thus makes difficult to generate the meaning of the content from it. Therefore, to resolve this problem, Tim-Berner-Lee, inventor of WWW and director of W3C visioned about Semantic Web.

According to Tim-Berner-Lee, Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in co-operation.

The goal of semantic Web is to represent data in structured format which would help machines to understand more information on the web which supports in richer discovery and data integration from different sources via linking hereby producing more exact results to the user as compared to current web search engines.

3.1 Architecture of Semantic Web

As part of the development of the Semantic Web vision, Tim Berners-Lee proposed a *layered architecture* for the Semantic Web [7] which is shown in Figure 3.The first layer consists of documents written in Unicode and their associated Uniform Resource Identifiers (URI) and URIRef. URI is a general form of identifier which allow user to create and represent a resource uniquely. It is not necessary that a URI of a resource will redirect to some other location as done by URL. URLref is another type of a string that represents aURI, and represents

the resource identified by that URI. It is a URI, together with the optional fragment identifies at the end separated by #.An

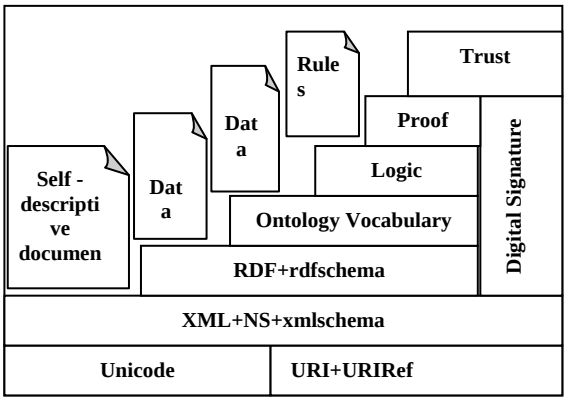

Figure 3. Architecture of Semantic Web

example of URI is http://www.w3school.org. The second layer contains XML, a general purpose markup language for documents containing structured information with XML namespace and XML schema definitions. It makes sure that there is a common syntax used in the semantic web. XML schema serves for expressing schema for a particular set of XML documents. The third layer contains the core data representation format for semantic web known as *Resource Description Framework* (RDF) and RDF Schema (RDFS). RDF represents data in the form of statement using <S*ubject-Predicate-Object*> that a system can process and understand. RDF uses URI to identify subject-Predicate-object and to process these RDF statements it uses XML. RDF defines a specific RDF/XML markup language, used to represent RDF information and for exchanging it between machines. RDF Schema defines framework to describe classes and individuals to define the vocabulary of that application.

The next layer contains OWL (Web Ontology language).The OWL is a language derived from description logics, and offers more constructs over RDFS. It is syntactically embedded into RDF, so like RDFS, it provides additional standardized vocabulary. *Logic and Proof layers* provide the ability to perform logic on semantic statements such as inferences and agents. Proofs are more difficult in that they must trawl many assertions to come to conclusions. The semantic web is based on the internet. Therefore, the levels of trust in assertions and knowledge must be determined if the source facts are to be believed. Digital signatures provide some trust elements, but referrals through the "web of trust" are also valid mechanisms. A level of trust (or distrust) will need to be factored into the agents and search engines that use the semantic web.

3.2 Why to represent data in a new format in semantic web when html is available?

Current web displays knowledge on pages using HTML which is unstructured in nature. HTML is a presentation language which displays data using tags. Web browser understands those tags and displays data accordingly but computer is not intelligent enough to understand the semantic of the content.

For e.g.

<HTML>

<TITLE>my current page</TITLE>

<BODY>

<H1>welcome to my home page

<I>this is an example of current web data presentation.

</BODY>

</HTML>

These tags are keywords which tell how to present data written in between them. The web browser reads the HTML document and uses these tags to interpret the content of the page.

Whereas semantic web is an extension of the current web in which information is represented in structured format using semantic language such as RDF, OWL, DAML, OIL etc.

For example in RDF "the sky has color blue", will be represented as the triple: a subject denoting the sky has color as predicate and object as value blue. Graphically, it will be represented as shown in Figure 4.

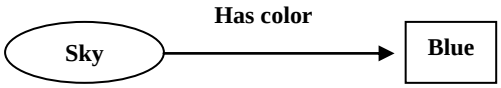

Figure 4. RDF Representation

Each and every object is identified by its URI which helps to resolve polysemy and synonymy problem that is often encountered in current web. A lot of generic ontologies are available such as Dublin core, FOAF etc which are used to represent an object.

For example, The author of Book is Ranjna Jain.

Here Subject: Book

Object: Ranjna Jain

Predicate: author

The vocabulary for the above statement is taken from Dublin core. The subject: Book can be represented by any URI such as http://www.w3.org/Book to understand the meaning.

Here, Dublin core is used to represent the predicate: author and it will be http://purl.org/dc/elements/1.1/creator and finally this statement would be expressed in RDF as

<rdf:rdf

xmlns:rdf= "http://www.w3.org/1999/01/22-rdf-syntax-ns#"

<rdf:description rdf:about= "http://www.w3.org/book">

<http://www.purl.org/dc/elements/1.1/creator> Ranjna Jain </http://www.purl.org/dc/elements/1.1/creator>

</rdf:description>

</rdf:rdf>

4. SEMANTIC SEARCH ENGINES

In order to access structured data, a number of semantic search engines has been introduced which understand the meaning of data and help in displaying more exact results as compared to current search engines. Some of the existing prevalent semantic search engines have been selected for discussion in this section with their architectures.

Swoogle" is a crawler-based indexing and retrieval system for Semantic Webdocuments using RDF and OWL. It is being developed by the University of Maryland Baltimore County (http://pear.cs.umbc.edu/swoogle/). It extracts meta-data and computes relations between documents. Discovered documents are alsoindexed by an information retrieval system

to compute the similarity among a setof documents and to compute rank as a measure of the importance of a SemanticWeb document (SWD).

4.1 Swoogle

Swoogle [9] is a crawler based indexing and retrieval system for semantic web documents (SWDs) written in RDF and OWL. The architecture of swoogle is discussed in Figure 5. Swoogle architecture can be broken into four major components: SWD discovery, metadata creation, data analysis and interface.

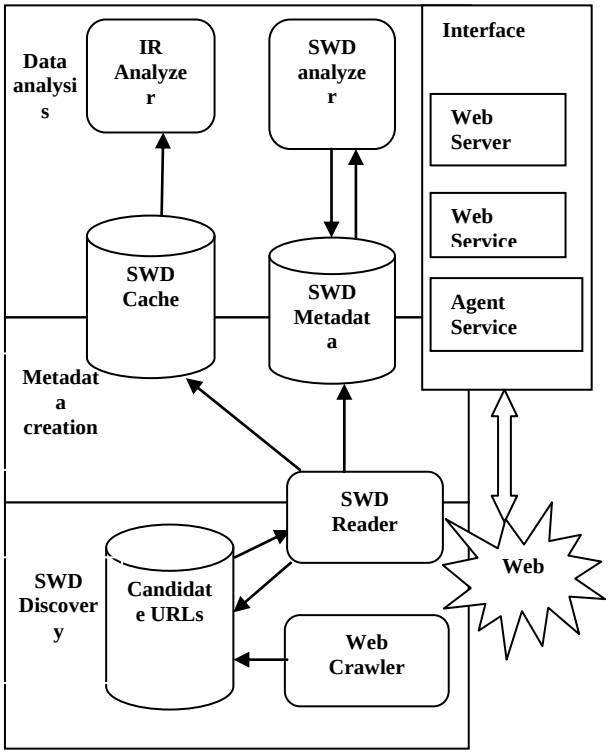

Figure 5. Architecture of Swoogle

a. SWD Discovery: At the back end, it creates a database of SWD's using hybrid approach to harvest the semantic web. It uses following mechanism to generate URLs to find SWDs on the web: (i) seed URLs and promising and trusted Sites (ii) URLs from conventional search engines using meta crawlers (iii) from swooglebot crawler that analyses SWDs and generate new URI candidates.

b. Indexing: This component indexes SWDs using its metadata and for this it captures encoding schemes namely "RDF/XML", N-triple, language such as OWL, DAML, RDFS, RDF. It records ontology properties such as label, comment, version info, relations between two SWDs via imports, extends etc.

c. Analysis: This component uses the created metadata to derive analytical reports such as classification of SWOs and SWDB, ranking SWDs using rational surfer model.

d. Services: This interface component focuses on providing data services such as search services that search ontologies at the term level.But swoogle has some limitations such as; it is not a general purpose search application and is restricted to retrieving ontologies files with embedded RDF content on the internet. Apart from this, it has poor indexing of documents and has long response time corresponding to fired query.

4.2 Falcon

It is a keyword based semantic search engine [10] which generates all the ranked RDF documents that include the terms in the fired query. For example user wants to know about BSAITM, then corresponding to this query, it tries to generates those RDF documents that contains this kind of information and in the form of snippet that exact information is shown so that user does not need to crawl unnecessarily to other pages. It displays required information on the snippet itself; therefore user does not need to explore the pages.

The Architecture of Falcon is described in Figure 6 and components are described below:

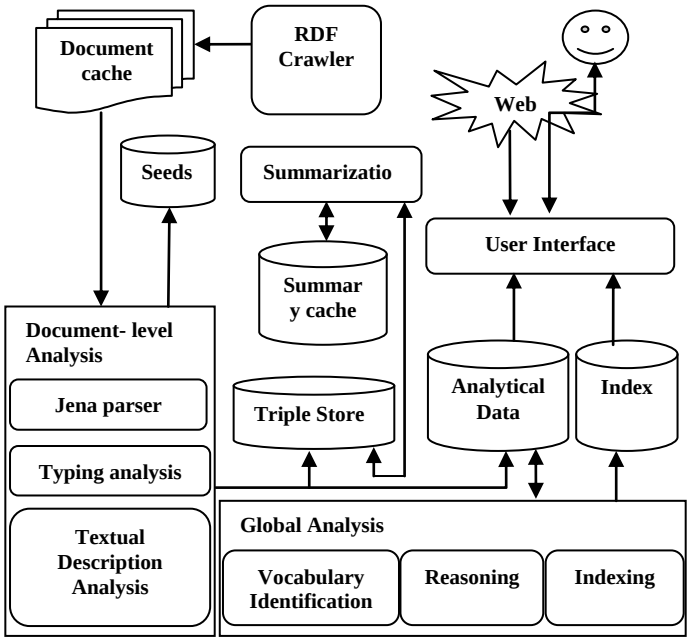

Figure 6. Architecture of Falcon Search Engine

a. RDF Crawler: An RDF crawler is setup to crawl RDF documents. It creates queries by enumerating general keywords which are sent to Google and swoogle to generate RDF documents. The crawler is also customized to download RDF documents from Dbpedia, Hannover, DBLP Bibliography, ping the semanticweb.com

b. Document level analysis: It contains jena parser, which parses the cache documents collected by RDF crawler. During this process new generated URIs are queued in the seed to explore more RDF documents. Falcon index URIs by including its local name, its associated literal values and description about its neighboring semantic web objects in RDF graph and corresponding to this, maintains a virtual document.

c. Global Analysis: Before indexing, vocabulary identification and then reasoning using class inclusion relation is done and then indexing is performed.

d. Summarization: A query dependent snippet of knowledge is provided to facilitate the end user to gather its information from the snippet itself.

e. User Interface: when a user gives a query to falcon, it serves a list of objects as well as types such as location, organization etc. with this, user can specify a type and focus on a particular dimension of knowledge.But Falcon has some limitations such as this engine is not interested to rank these objects according to query.

4.3 Hakia

Hakia[12] is a semantic search technology based search engine that presents relevant results based on concept match rather than keyword match or popularity ranking. The Architecture of Hakia is described as below in Figure 7 and of components described below.

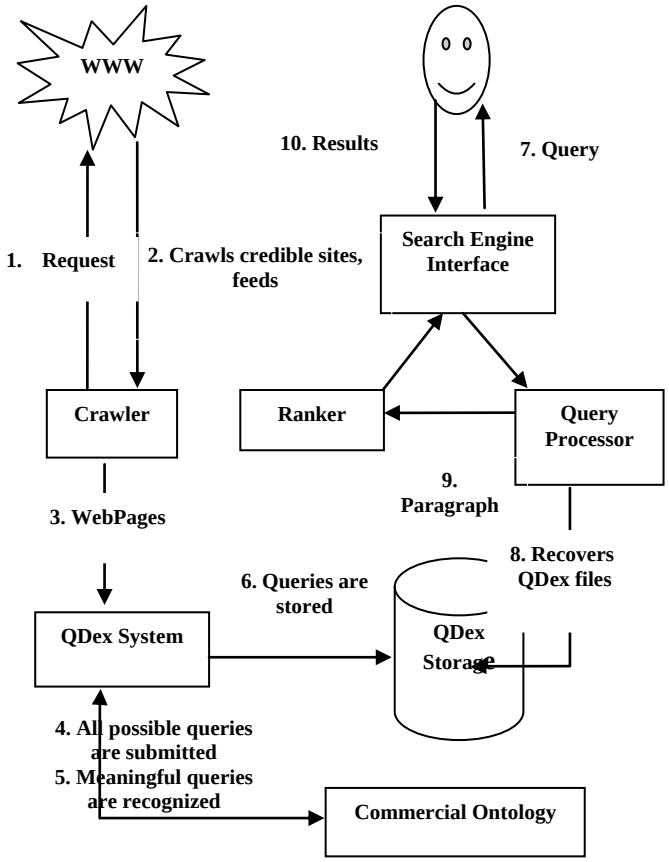

Figure 7. Architecture of Hakia Search Engine

a. Crawler: Hakia forms a collection of relevant document from credible site recommended by librarian. It also crawl dynamic content from Blogs, news, database etc.

b. QDexing: After collecting data from different segments, QDex(stands for Query Detection & Extraction) analyzes each web page and extracts all the possible queries that can be asked to that page by decomposing sentences into sequences of words resulting generating the vast number of queries.

c. Commercial Ontology: Here, all the extracted queries are further analyzed by methods such as morphological analysis, generalization, and characterization and by this queries are categorized into various senses they convey.

d. QDex Storage: It creates or maintains a file for each query which stores information about the document, paragraph from which that it was extracted. After that, each Qdex file is placed in a known destination via hash-mode operation. All this work is performed offline.

e. Query Processor: When the user fires a query from user interface, the query is sent to the query analyzer to generate the sense and context of the user using fall back algorithm and with hash mode, Qdex files destination location is retrieved correctly.

f. Ranking: A pool of relevant paragraphs are ranked by semantic analysis rank algorithm which is based on advanced sentence analysis and concept match between the query. And the best sentence for each paragraph which will be highlighted in the snippet to attract the user is retrieved.

But hakia has some limitation such as it has some issues such as URL canonicalization, privacy session ID's, virtual contents and dynamic contents.

4.4 Semantic Web Search Engine (SWSE)[13]

It is a search engine for RDF data on the web, and provides the equivalent services a search engine currently provides for HTML Web. The architecture is shown in Figure 8 and component details are given below.

a. Crawler: It starts with a set of seed URIs, retrieves the content of URIs, parses and writes content to disk and recursively extracts new URIs for crawling. Currently, it crawls RDF/XML syntax documents which are most commonly used for publishing RDF on the web.

b. Consolidation: It provides a mean of identifying equivalent entities in RDF data for e.g.; OWL defines the owl:sameas property which relates two equivalent entities; entities representing the same real world individual but identified incongruously-would enable the merging of information contribute on an entity given by heterogeneous source without the need for consistent URI naming of entities.

c. Ranking: It ranks crawled data by considering URI redirections encountered by the crawler while performing the link based analysis.

d. Reasoning: By appending instance data(i.e. assertional data) describing about an object, SWSE uses scalable authoritative OWL reasoned (SAOR) system to infer logical consequences from a set of facts or axioms described using classes and properties and system pre-compute inference to avoid the runtime expense.

e. Indexing component: It employs an inverted index for keyword lookups based on RDF literals (text), and a sparse index for lookups of structured data. With a pair of keys and pointers for every entity in the data file, every entity in this file is associated with a particular pointer to the block in the sorted data file. This block contains entity snippet containing a detailed description which is formed by

aggregating from many sources, description also include inferred data which is not necessarily been published but derived from the existing data through reasoning.

f. Query processing and User interface: It accepts user queries, retrieves top k hits and requests the snippet result data for each of the hits and display them as an output at interface.

But Semantic Web Search engine has some limitations such as poor ranking of documents because the Ranking process comes before the indexing stage. Ranking technique is coming independently with data indexed in dataset.

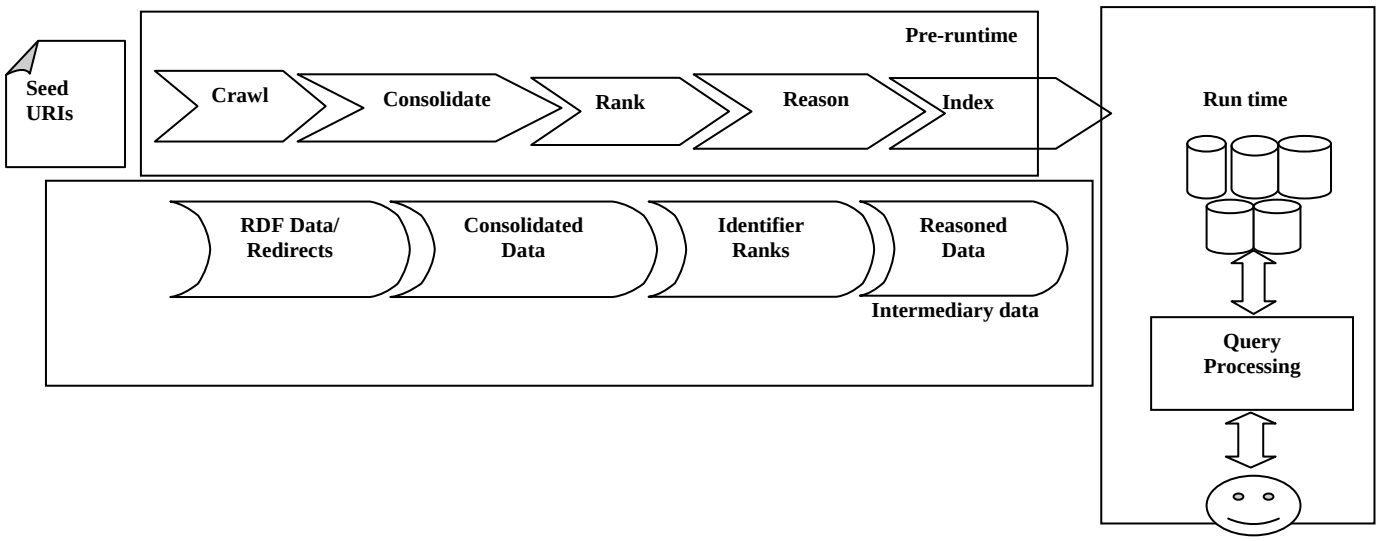

Figure 8. Architecture of Semantic Web Search Engine

5. COMPARATIVE STUDY

The comparison of discussed Search Engines is performed on various measures like the underlying technique, input parameters required, working levels, complexity, quality and relevancy of the returned pages etc. The detailed comparison study is outlined in Table 1.

Table1: Comparative study of Google, Falcon, Hakia, SWSE

	Google	Hakia	Swoogle	SWSE	Falcon
Evolution:	**Research project by Larry Page and Sergey Brin, Ph.D students at Stanford working on Stanford Digital Library Project**	**Founded in 2004 and is funded by private institutional and angel investor**	**Research project of Ebiquity Research group in the Computer Science & Electrical Engineering Department at the University of Maryland, Baltimore County(UMBC)**	**Research project of Digital Enterprise Research Institute, National University of Ireland, Galway**	**Research project of Web Software research group at the department of Computer Sc. and Technology, Nanjing Univ, P.R. China**
Main Feature	**1. Web based search engine operates over web pages 2. Several features such as Query Refinement, Choosing Keywords, everyday essentials and many more**	**Web search generates relevant results based on concept match rather than Keyword match**	**1. Search semantic web ontologies and documents 2. Searches SW terms i.e.; URIs 3. provides metadata of SWDs.**	**Keyword based search engine for object, operates over RDF data**	**Keyword based search engine for objects, concepts(class and properties), ontologies and RDF documents**

What does it Crawls?	HTML Web Pages	HTML Web Pages	RDF data	RDF data	RDF data
Reasoning Technique	Not available	Not available	Rule based systems, Bayesian reasoning	SAOR (Scalable authoritative OWL reasoner)	Class inclusion relation
Indexing Scheme	Inverted indexing	Qdex (Query Detection & Extraction)	Swangling Technique	Inverted indexing for literals sparse indexing for structured data	inverted indexing
What it Indexes?	terms	keeps a collection of queries extracted from the web page	RDF triples	RDF literals structured data (detailed description of entity aggregated from different sources)	Local name of URIs along with its associated literals description about neighbouring objects
Ranking Technique	Page Rank algorithm	Semantic rank algorithm	Rational surfer model	through link based analysis	Objects ranked based on combination of their relevance to the query and their popularity. Uses cosine similarity measures b/w the query and virtual doc. Of objects
Knowledge Snippet	Yes	Yes	Metadata	Yes	Yes
Results	document	Document	Entity	Entity	Entity

6. CONCLUSION

Web 2.0 search engines are unable to present direct answers against the user's fired query because Web 2.0 contains information which is unstructured in nature. Web 3.0 is Semantic Web using RDF format organizes information in more structured form which helps the semantic search engines such as swoogle, falcon, SWSE etc. to present results in a more direct way. But these semantic search engines deals with structured data written in RDF or OWL Format only. Web 2.0 also contains a huge library of interlinked documents that are in semi-structured (CSV and XML) and structured form (database). They can be used as data exchange format in different domains- XML covers the syntactic level but lacks reasoning. But, if files get converted in OWL files then they can be used by semantic web search engines and can expand its coverage area. This Paper has reviewed working of above discussed Semantic Search Engines and their corresponding techniques used in Crawling, Indexing, Ranking and Result formation process.

7. REFERENCES

[1] Swoogle Search Engine available at: http://swoogle.umbc.edu/

[2] Comparision between phases of world Wide Web available at: http://ahmedyassen.xomba.com/web-1.0-web-2.0-web-3.0-and-web-4.0-review.html

[3] S. Brin, L. Page, The Anatomy of a Large-Scale Hypertextual Web Search Engine, Computer Networks 30 (1-7) (1998)

[4] Web Directory available at: http://en.wikipedia.org/wiki/Web_directory

[5] Meta-Search engine available at: http://en.wikipedia.org/wiki/Metasearch_engine

[6] Berners-Lee T., Hendler J. and Lassila O. The Semantic Web. The Scientific American, vol. 5(1) 2001. URL:http://www.scientificamerican.com/

[7] Patel-Schneider P.F. and Fensel D. Layering the Semantic Web: Problems and Directions. In Proceedings of The Semantic Web - ISWC 2002: First International Semantic Web Conference, Sardinia, Italy, vol. 2342 / 2002. springer-Verlag GmbH 2002, p. 16.

[8] Ora Lassila, ralph R. Swick, Resource Description Framework(RDF) model and syntax specification, February 1999, www.w3.org/tr/rec-rdf-syntax-199990222.

[9] Li Ding Tim Finin Anupam Joshi Rong Pan R. Scott Cost Yun Peng Pavan Reddivari Vishal Doshi Joel Sachs Swoogle: A Semantic Web Search and Metadata Engine. ACM Press 2004.ppp. 652-659

[10] Cheng, G., Ge, W., Wu, H., Qu, Y.: Searching Semantic Web Objects Based on Class Hierarchies. In: Bizer, C., Heath, T., Idehen, K., Berners-Lee, T. (eds.) LDOW 2008. CEUR-WS, vol. 369. CEUR-WS.org (2008)-researchgate.net

[11] Y.Qu, G.cheng,H.Wu, W.Ge, X,Zhang, Seeking Knowledge with Falcon, Semantic web Challenges.

[12] http://company.hakia.com/new/documents/White%20Paper_Semantic_Search_Technology.pdf

[13] Aidan Hogan, Andreas Harth , J□urgen Umbrich , Sheila Kinsella , Axel Polleres , Stefan Decker, Searching and Browsing Linked Data with SWSE: the Semantic Web Search Engine. Journal of Web Semantics, Vol 9 no.4 (2011)

[14] Services provided by falcon available at: http://iws.seu.edu.cn/services/falcons/

[15] T. Berners-Lee, Linked Data, Design issues for the World Wide Web,World WideWeb Consortium, http: /www.w3.org/DesignIssues/LinkedData.html (2006).

[16] Hai Dong, Farookh Khadeer Hussain, Elizabeth Chang,A Survey in Semantic Search Technologies, , 2008 Second IEEE International Conference on Digital Ecosystems and Technologies (IEEE DEST 2008)© 2008 IEEE.