DBPedia Based Meta Search Engine

Kim Soon Gan[1#], Kim On Chin[1], Patricia Anthony[2], Vooi Keong Boo[1]

1 Center of Excellence in Semantic Agent, Faculty of Computing and Informatics, Universiti Malaysia Sabah, Jalan UMS, 88400 Kota Kinabalu, Sabah, MALAYSIA,
2 Faculty of Environment, Society and Design, Lincoln University, Christchurch, NEW ZEALAND
# Corresponding author. E-Mail: g_k_s967@yahoo.com; Tel: +6088-320000; Fax:+6088-320390.

ABSTRACT

Retrieving information on the web has become more challenging due to the overwhelming growth of online data. Consequently, even a good ranking algorithm is not able to deliver a precise search result. Traditional search engines work based on keyword search rather than the meaning of the keyword being searched. The main problem with keyword search is that it cannot solve the synonym and polysemy problems. Semantic search engines address this problem by focusing on the meaning of the query rather than considering it as a mere search phrase (keywords). In this work, we propose MEOW, a semantic meta search engine that utilises the power of a traditional search engine and enriches the search results by trying to understand the meaning of the search query. The proposed meta search engine uses DBPedia as its knowledge base and Google to generate the search results. The search results classify which semantic concept for a given query is used for different context. The main idea of this work is to transform non-semantic search results into semantic results by extending the search query to get more quality results based on the meaning of the search query.

KEYWORDS: Semantic; Meta Search Engine; DBPedia; Information Retrieval; Clustering

Received 24 May 2017 **Revised** 25 September 2017 **Accepted** 28 September 2017 **Online** 16 October 2017

INTRODUCTION

The World Wide Web (WWW) was developed at CERN under the leadership of Tim Berner-Lee in 1990 (Berners-Lee, 2000). The idea behind the inception of WWW is to build a system that is composed of computers distributed widely such that information can be accessed anywhere and anytime as long as there is an Internet connection. Information made available on the Web is commonly known as a web page and is identified by a Universal Resource Locator (URL). Each URL refers to a type of resource on the web so that a client program, such as a browser, can accurately fetch the information required by a user on the web. One of the features of the Web is that a web page can provide hyperlinked information that points to another page based on its URL. This design enables each page to act as an individual information resource. For example, if new information becomes available, a new web resource (URL) can be created by adding a hyperlink to this new web page, which in turn makes this new web page immediately accessible.

The Web focuses on how information can be shared on a greater scale. In December 1995, there were a total of 16 million web users. As of March 31, 2017, the number of web users was 3,731,973,423. The numbers of websites have also grown from 23,500 in Dec 1995 to 3 billion in May 2017. This growth simply means that there are massive data available on the Web. While such data growth allows us to search for information, this growth also means that looking for information is not a trivial process. There are two issues that arise from the growth of the web. Firstly, the size of the web and secondly, the rate at which it is growing. The enormous size of information and its exponential growth has increased the complexity of the information retrieval process. Often, we may not find what we want as there is too much information. This is the reason why search engines have been developed to ease the process of searching for the right information (Brin & Page, 1998).

Search engines scan the content of the web on behalf of human beings and store the visited pages to an indexed table. When users try to search for certain information, they can be directly referred to the indexed table instead of surfing from one page to another, a process that can consume a considerable amount of time. The basic operations of a search engine include crawling, indexing, and query matching. During the crawling process, the crawler starts with a selected page on the web, records the content of the web page, and finds new links from the visited page and repeats the crawling process (Spetka, 1994). Depending on the speed and the effectiveness of the crawler, it is possible to record a large part of the web content without the involvement of humans. The collected information is then indexed using techniques such as forward index and inverted index (Salton & McGill, 1983). The main idea of this process is to extract the significant features in each parsed page so that these features can constitute a key to retrieve the document during the search process. Finally, the query matching involves matching the user's search query with the index created to retrieve the documents that contain the same feature as the search query (Baeza-Yates & Ribeiro-Neto, 1999).

Tim Berners-Lee discovered the limitation of the Web's effectiveness after he invented the web. Many researches were conducted in order to build a more effective web. The web has evolved from Web 1.0 to Web 2.0 and what is now Web 3.0 (Strickland, 2007). In Web 1.0, the web merely acted as an information portal with the aim of connecting information and getting it on the web. However, there are several drawbacks in Web 1.0 such as the scalability problem and the lack of interaction between the users and the web page. The basic feature of Web 1.0 is that a web page is just a plain document and the web server acts as a remote application that is controllable through the web. The basic role of a web browser in Web 1.0 is to retrieve information from a web server and display it to the user. The advancement of technology has enabled a more dynamic interaction between human beings and web pages. This has led to Web 2.0 where interaction and collaboration are more prominent. In Web 2.0, apart from being able to send static content to a web browser, the web server is also able to send programming logics that tune it into a software agent. Users can also access various web services through the web page. Some examples of Web 2.0 are social networking sites and blogs where users can edit content on the web directly through a web page.

In Web 2.0, most search engines treat search query merely as "strings" wherein the search results are generated based on pattern matching. These search engines do not have the ability to understand the meaning of the query and this is the main reason why they are not able to deliver accurate search results. One possible solution is to build web pages that can be processed and understood by both computers and human beings. This is the basic idea of the semantic web, which allows the web to be processed by the computer in order to generate more meaningful search results for the users (Bernes-Lee *et al.*, 2001; Antonious & Harmelen, 2008). Semantic Web or Web 3.0 makes this possible by providing a mechanism where content of the web is not only readable by a human being, but by a computer too. Web 3.0 allows a computer program to achieve a higher level of efficiency in retrieving the "right" information. It can be considered as linking of data on the web in which a computer program can explore and process the data based on the interconnections between data. The benefit of semantic web is that a computer program can help analyse the web page more efficiently than Web 2.0. The development of semantic web focuses on the representation of information on the web so that it can provide the exact information on the web to any software agent accessing it. For example, if a web page wishes to deliver information that "certain shop is selling an apple product", web content like "shop X is selling apple" is not clear enough, since users may interpret it as a shop that sells "apple" as a fruit.

In Web 2.0, a browser is able to process HTML tags and display them on the screen using the appropriate interface for users to view. However, it does not understand the meaning of the web page content. Semantic Web solves this problem by providing a set of design principles and a variety of enabling technologies (Berners- Lee *et al*., 2001). Some of these important 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. These technologies will be described in detail in Section 2 of this paper.

Semantic search engines are capable of searching the web content based on the meaning of the search query. The fundamental principle of a pure semantic search engine is that the search query or the web content being crawled should not be treated as just a string, but as concept. This principle is simple yet difficult to be achieved without a good model to conceptualise both the information from the user and the web content. Semantic web standardises the way information is presented on the web such that this information can be easily mapped to the user's request. As long as the search engine is able to understand the user's search query and the web content, it can be considered a semantic search engine. Semantic search engines are now being implemented in Web 2.0 to make searching of information more efficient and accurate.

A semantic search engine must be able to interpret the meaning of the user's search query as well as to index the web content based on related topics. Traditional search engines are not able to understand language in the same way as human beings do. The difficulty in a search engine's understanding of human language results in most of the complex queries remaining unanswered. There are a number of semantics search engines for Web 2.0 that have been developed such as Hakia, Kngine, Sensebot, Omnity, and Google. Google moved from being a keyword based search engine to a semantic based search engine in the summer of 2013 with the introduction of its Hummingbird algorithm. This research was conducted prior to the introduction of Google Hummingbird. The details of these semantic search engines will be discussed in Section 3.

The purpose of this paper is to describe MEOW, a meta semantic search engine for Web 2.0. Meta semantic search engine is this context means that a traditional search engine is used, but the results were pre-processed to generate more meaningful search results. This paper also investigates the suitability of applying partial semantic search by utilising DBPedia and RDF to process the search query and search results obtained from Google. DBPedia is a crowd source community effort that extracts unstructured information from Wikipedia in a manner that allows this information to be queried like a database. DBPedia extracts Wikipedia based on its structure using an extractor. The extracted data is organised based on selected DBPedia datasets. These datasets are used as the knowledge base (KB) to enrich the search results obtained from a traditional search engine (Google). We used DBPedia in this work as it contains a large amount of readily available concepts (resources). This paper is an extension of a previously published paper with more detailed explanation (Boo & Anthony, 2011).

The remainder of the paper is organised in the following manner. Section 2 provides a brief overview of semantic web technologies and their roles in the semantic web. Section 3 describes various works that have been conducted in relation to search engines, and in particular, semantic search engines on the web. In Section 4, we discuss DBPedia and some of its useful datasets. In the following section, we describe our meta search engine design. Section 6 describes some of the features of MEOW. Finally, the conclusion and future works are discussed in Section 7.

BACKGROUND THEORY THE SEMANTIC WEB

The Semantic Web is seen as the extension of the Web that provides a standardised way of expressing the relationship between web pages such that the content of the web pages can be understood by computers (Bernes-Lee *et al.*, 2001). W3C has established standardised layers of technologies and standards for the semantic web as shown in Figure 1. The core of the semantic web technologies consists of XML, RDF, RDFS, and OWL (Koivunen & Miller, 2001).

The XML (Extensible Markup Language) has been widely adopted in Web 2.0. The popularity of XML is due to its simplicity and flexibility in allowing the user to define the markup elements where the main intention is to aid information systems in sharing structured data. It is for this reason that XML has been widely used to encode documents on the Internet and serialise data for comparison with text based serialisation languages. This simple and flexible text format is derived from SGML (Standard Generalized Markup Language). It is now a fee-free open standard recommended by W3C. XML, in combination with other standards, makes it possible to define the content of a document separately from its formatting while making it easier to reuse that content in other applications or for other presentation environments. More importantly, XML provides a basic syntax that can be used to share information between different kinds of computers, different applications, and different organisations without the need to pass through many layers of conversion.

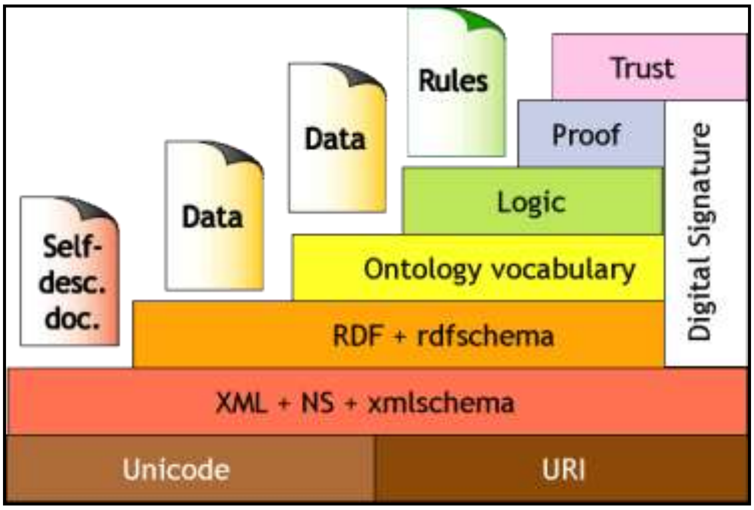
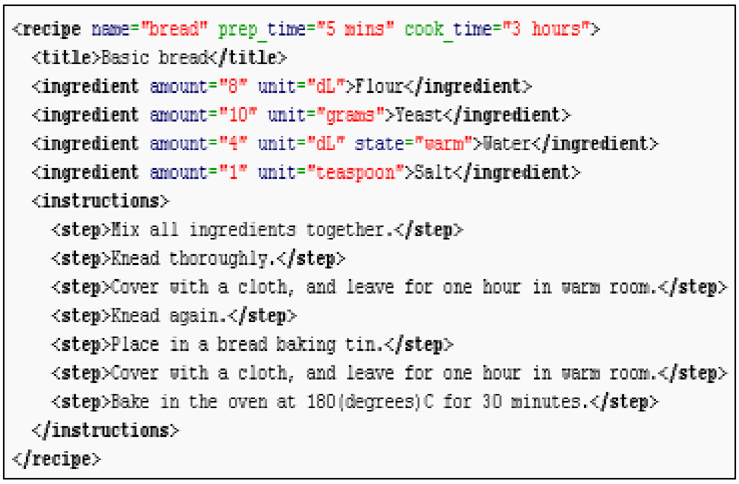


Figure 1. The Semantic Web Layer (Koivunen & Miller, 2001)



```
<recipe name="bread" prep_time="5 mins" cook_time="3 hours">
  <title>Basic bread</title>
  <ingredient amount="8" unit="dL">Flour</ingredient>
  <ingredient amount="10" unit="grams">Yeast</ingredient>
  <ingredient amount="4" unit="dL" state="warm">Water</ingredient>
  <ingredient amount="1" unit="teaspoon">Salt</ingredient>
  <instructions>
    <step>Mix all ingredients together.</step>
    <step>Knead thoroughly.</step>
    <step>Cover with a cloth, and leave for one hour in warm room.</step>
    <step>Knead again.</step>
    <step>Place in a bread baking tin.</step>
    <step>Cover with a cloth, and leave for one hour in warm room.</step>
    <step>Bake in the oven at 180(degrees)C for 30 minutes.</step>
  </instructions>
</recipe>
```

Figure 2. *XML* Encoding

In the semantic web's context, XML allows the user to structure the data in a sharable order and can be used across different applications and platform (Davies *et al.*, 2006). Besides, XML allows users to create their own tags and hidden labels where scripts or programs can then make use of these tags for a variety of purposes. In other words, XML allows users to add arbitrary structure to their documents by having shareable information that can be interpreted by different applications or platforms. Figure 2 shows an example of XML encoding.

Resource Description Framework (RDF) is a language for representing information about resources on the Web such that information can be processed by software rather than humans (Manola & Miller, 2004; Klyne & Carroll, 2004). RDF provides a common framework to represent information that is exchangeable between applications. RDF is based on two concepts:

- Anything in the world is unique and can be identified by an identifier called Uniform Resource Identifier (URI).
- A resource can be described in combination with simple properties and property values.

URI is used to identify a concept or resource uniquely. This feature is important in realising the semantic web. For example, two pages discussing the same topic may have different approaches in presenting the topic. There is no way to tell whether a particular subject discussed on both web pages refers to the same thing even though both pages may refer and use exactly the same words. Humans tend to associate a concept with certain words, but the mapping between concept and a word is not a one to one relationship. In some cases, a single word can refer up to ten distinct concepts, because new information is continuously added to the web. To elaborate this issue, assume that we refer to a person by his or her first name. Referring to a person using the first name may not be an issue in a small community (of ten to twenty people) as each individual in that community will most likely have a unique first name. However, on a global scale, a first name like "William" may belong to many people and it would be difficult to identify which "William" is being addressed. A simple solution to this problem is to add a unique feature to the first name in a way that the ambiguity can be resolved. This can be done using a combination of first name and last name. However, there is still a problem as we can still find two persons with the same first name and last name. Therefore, to uniquely identify a person, some form of identification numbers are introduced such as social security number, identification card or the driver's license number.

This simple concept can be applied to the web. URI is used to represent information on the web to prevent ambiguity since each URI is unique. For example, a page that talks about "cat" as an animal can refer to URI such as *http://animal-spec.org/Cat*, while a page that talks about "cat programming language" can refer to URI like *http://computer-spec.org/Cat*. This enables a software agent to identify information on the web on behalf of humans. A user who is interested in "cat" as an animal can instruct his agent to search for this information using this unique URI and only web pages that talk about concepts associated with *http://animal-spec.org/Cat* will be returned.

An RDF statement is composed of three parts, which are the subject, the predicate, and the object. Any description about a resource can be decomposed into a simplest form of sentence that only contains the subject, the predicate, and the object. In general, a subject refers to a URI, while a predicate defines the property of the subject. An object can be a URI or just literals. For example, given a sentence:

John has a house located in Kota Kinabalu

This sentence can be summarised using these two short statements:

John [subject]	*has a* [predicate]	*house X* [object]
house X [subject]	*located in* [predicate]	*Kota Kinabalu* [object]

Even though both statements are now in a form controlled grammar, there are still some ambiguities that exist in the literals. Using URI, the two statements can be rewritten as:

ex:John	*ex:own-property*	*ex:Property001*
ex:John	*ex:Name*	*"John"@en*
ex:Property001	*ex:Type*	*ex:House*
ex:Property001	*ex:locatedIn*	*ex:KK*
ex:KK	*ex:Name*	*"Kota Kinabalu"@en*

These statements are referred to as triples in which *ex:John* is used to represent *http://example.org/John*. Based on these statements, not only can a URI be used to form a triple, but a literal value can also be used. Literal value is normally used to represent human readable information such as name, address, or comments on a concept that is not to be represented using a URI. Based on these triples, a graph consisting of nodes and arcs can be built as shown in Figure 3. In short, with RDF, we can represent the resource, property or value for a given statement. From the language's perspective, URI is the word in the language, while the set of triples are the grammar of the language that can be processed by the computer. Hence, a group of statements can form a paragraph to describe the information in detail.

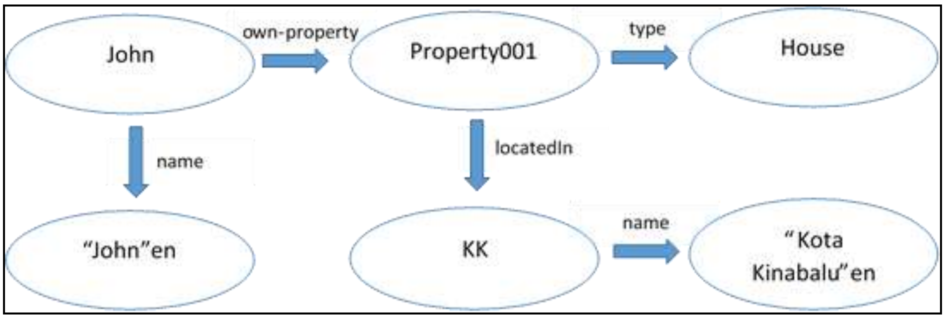


Figure 3: A graph that describes *John has a house in Kota Kinabalu*

RDF is a general purpose language used to describe information on the web. It defines how information should be presented on the web page such that a computer can parse the information accurately. However, it still lacks the semantic required for a computer to process the information. RDF's vocabulary description language, RDF schema (RDFS) is a semantic extension of RDF (Brickley & Guha, 2004). It provides a way to describe groups of related resources as well as the relationship between these resources. Two namespaces are commonly used for vocabulary in RDFS namely *rdfs,* which refers to *http://www.w3.org/2000/01/rdf-schema#,* and *rdf,* which refers to *http://www.w3.org/1999/02/22-rdf-syntax-ns#*. An *rdfs:label* can be identified using *http://www.w3.org/2000/01/rdf-schema#label*. The use of namespaces makes it easier to read and write a URI.

One of the relationships that can be represented using RDFS is a tree hierarchy concept. Under RDFS, resources can be grouped into classes, where the members of a class are called instances and each of the instances can have its own description described using RDF. At the same time, an instance of a class should have all the properties defined in a class. Vocabulary states that a certain resource is an instance of a certain class *rdf:type*. For example, consider a triple:

Ex:John	*rdf:type*	*ex:Person*

This triple means a resource "John" is an instance of class Person under the prefix "ex". In other words, it means "John is a Person". Furthermore, the vocabulary used to represent a class is *rdfs:Class*. Hence, one can state that a Person is a class by the following triple:

Ex:Person	*rdf:type*	*rdfs:Class*

A class is used to group resources that share similar properties; therefore, there is no limit on how many classes that can be defined as well as the number of classes that an instance can belong to. The hierarchy of classes can be defined using *rdfs:subClassOf* property. Classes can also be inherited such that if class A is a subclass of class B and C is an instance of class A, then C is an instance of class B as well. All resources on the web should be instances of the resource class (*rdfs:Resource*). Other relationships that can be represented with RDFS are the behaviour of property. Given a sentence, "Mary drinks Milo", it can be assumed that "Milo" is a drink. RDFS defines two vocabularies, *rdfs:domain* and *rdfs:range,* which can induce this logic. The first one, *rdfs:domain* defines the class of the subject while *dfs:range* states the range of values for a property. For example, consider the following triples:

ex:eat	*rdfs:domain*	*ex:Person*
ex:eat	*rdfs:range*	*ex:Food*
ex:ano	*ex:eat*	*ex:X*

In this example, a computer can conclude that "ano" is a Person, while "X" is a kind of food. These kinds of vocabularies allow a computer to perform some form of reasoning based on the content of the web. There are many vocabularies defined in RDFS, such as the *rdfs:label,* which is used to express a concept with human readable symbol *rdfs:subPropertyOf,* which has a function similar to *rdfs:subClassOf*. In summary, the introduction of RDFS is to add more value to the data on the web. To illustrate this further, consider the previous statement "John has a house located in Kota Kinabalu". Even though this statement can be translated to an RDF format, some descriptions are still missing like "who is John" and "where is Kota Kinabalu". Using RDFS, if we assume that "ex" refers to *http://example.org/*, the statement can be further refined as follow:

ex:John	*ex:own-property*	*ex:Property001*
ex:John	*rdfs:label*	*"John"@en*
ex:John	*rdf:type*	*ex:Person*
ex:Property001	*rdf:type*	*ex:house*
ex:Property001	*ex:locatedIn*	*ex:KK*
ex:KK	*rdf:type*	*ex:SabahCity*
ex;KK	*rdfs:label*	*"Kota Kinabalu"@en*

After these extra triples are added, we can now deduce that "John" is a Person while "KK" is a city in "Sabah". A software agent can further explore the detail of the Person class by following its URI to get more information.

Web Ontology Language (OWL) is a specification designed to further extend the sense of semantic, which is not supported in RDFS, by providing a better vocabulary to the model knowledge to be processed by a computer (McGuinness & Harmelen, 2004). Unlike the RDFS vocabulary, which states a direct relationship between concepts, OWL provides vocabulary that enables a computer to perform broader reasoning for data on the web. OWL vocabulary can describe various types of concepts such as equality, cardinality, property restriction and characteristics of property (Rector *et al.*, 2004; Doan *et al.*, 2004).

Equality concept can be described using *owlsameAs* property. For example, assume that there are two websites. Website X provides information about "cat" (as an animal) described using *ex:Cat*, and another website Y that sells "cat"(as an animal) with their own description of cat using *ez:Cat*. If website Y wishes to emphasise that they sell "cat" (as an animal), which is the same concept described on website X, then they can provide a triple as follows:

ez:Cat *owl:sameAs* *ex:Cat*

This triple means that both concepts "cat" are the same. In this case, if a software agent wants to buy a "cat" based on the reference *ex:Cat*, then it can explore website Y as well. On the other hand, there is also a vocabulary called *owl:differentFrom,* which is the opposite of *owl:sameAs*. For example, "C language" is referred to as a "programming language"; however, if there is another meaning for "C language" and if a web page wants to emphasise that it is different from the previous concept, then it can use *owl:differentFrom* to indicate that the topic is not the same as the "C programming language" concept. The characteristics of a property can be described using vocabularies such as *owl:inverseOf* and *owlTransitiveProperty*. *owl:InverseOf* allows one to specify two properties that are the inverse of each other. For example, if property *ex:hasChild* is stated as inverse of the property *ex:hasParent*, then there exists a triple with the content:

ex:John *ex:hasChild* *ex:Mary*

It can then be deduced that "Mary" has a parent called "John". *owl:TransitiveProperty* is used to represent that a property is transitive. For example, if "Kota Kinabalu" is located in "Sabah" and "Sabah" is located in "Malaysia", logically, we can deduce that "Kota Kinabalu" is in "Malaysia" as well. OWL enables more interaction and collaboration between data on the web by linking data on the web using appropriate vocabularies. This allows for all the information on the web to be integrated as a single source of data when they are linked to each other.

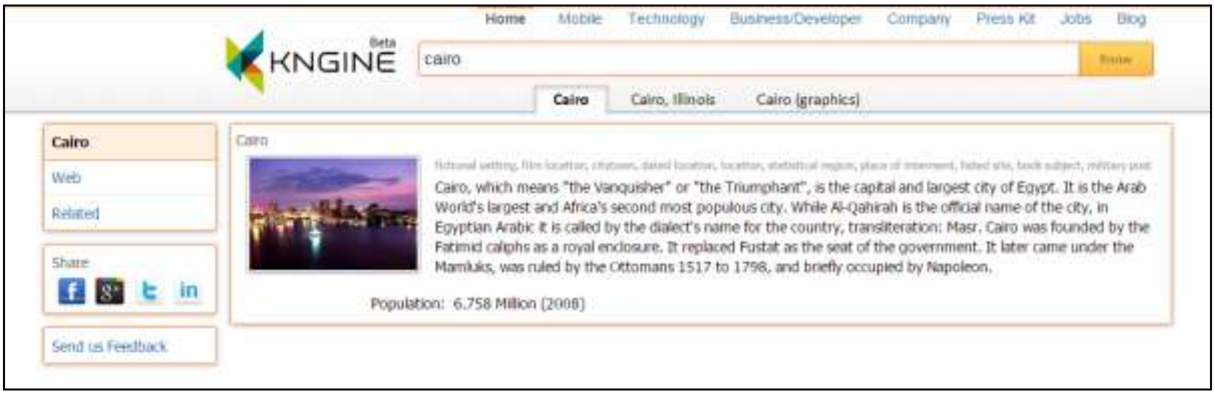
RELATED WORK

In general, semantic search refers to a search based on the meaning of a query instead of the word pattern found in a search query. Research on semantic search is not limited to the web, but to local data repository as well. For example, semantic search can be used to retrieve all the computer science papers in a database of academic papers. It should be emphasised that although semantic search is not just for the web, the web needs semantics search the most due to the format free nature of the web as well as the web size expansion that requires a high precision search schema for a search engine. Otherwise, the most complex questions will remain unanswered. In this section, we describe and discuss a few commonly used semantic search engines namely, Hakia, Kngine, SenseBot, Koru, Look4, DuckDuckGo, Omnity, and Google.

Hakia (Tumer *et al.*, 2009; Kumar & Goel, 2012; Singh & Sharan, 2013) is an ontological semantic and natural language processing based search engine. It focuses on the web content for Web 2.0. The creators of Hakia believed that a semantic search should be done just like human interaction in real life and therefore the most important feature in a search engine is the intelligence itself. It should make a computer smart enough to answer questions from humans without depending on the statistical information (user behaviour data or human power).This can solve issues where less popular topics may not be ranked properly, because of the lack of statistics on people's preference for that topic. The advantage of Hakia is in its ability to make a semantic analysis on the web content to answer user search queries and hence preventing the usage of statistical sampling. This gives better search results when searching for dynamic content such as news as these types of information lack statistical information.

Hakia processes information using its proprietary core semantic technology called QDEX (Query Detection and Extraction). The idea behind QDEX is that instead of indexing all the words into a table for web resource like the traditional search engines, it parses a sentence as a whole. Each sentence is assumed to answer a certain question. QDEX decomposes sentences into meaningful knowledge sequences. For example, the sentence "the cat is on the mat" is the answer for a question "where is the cat?" or "what is on the mat?". Therefore, if someone asks a query "where is the cat?", then supposedly, the search system can supply "the cat is on a mat" as one of the candidate answers. Hakia applies a fuzzy logic algorithm to parse the entire sentence. This process is called breeding where it generates all possible questions that a page can potentially answer and indexes the page based on the question list. When a search query comes in, it is assumed that it is asking for a certain question. Based on the query (question), Hakia will just pick the result from the question list generated previously. Another advantage of Hakia is the scalability of the indexing system. An inverted index will keep growing as the number of web pages crawled increases, which is proportional to the size of the web. QDEX on the other hand, only grows when there is a new knowledge sequence generated. Hakia has terminated its public service since 2014 and is only providing service to private/enterprise audiences.

Kngine is a search engine that merges the characteristics of both a semantic search engine and a question answer engine. The goal is to organise human beings' Systematic Knowledge and Experiences and make them accessible to everyone (Sudeepthi et al., 2012). Kngine tries to understand documents on the web and user search queries rather than just depending on the inverted indexing method to provide meaningful search results. For this purpose, Kngine collects structured information from various sources to build a knowledge base that helps in understanding the web content and the user query. The search approach implemented in Kngine includes multiple meanings word perception and question answering technology. It provides dynamic search results based on different types of search query such as providing semantic information about user query that is recognised as a concept or a direct answer about a user question in the query. For example, a word "leopard" can refer to an animal, but at the same time, it can also refer to "Apple operating system". By detecting the existence of the word with multiple meanings, Kngine can provide extra information about each meaning to be referred based on the structured information collected. Figure 4 shows Kngine's search results with the query "Cairo". As can be seen, three meanings are detected; "Cairo" as a capital city of Egypt, "Cairo" as a city in Illinois, and "Cairo" as a software library.

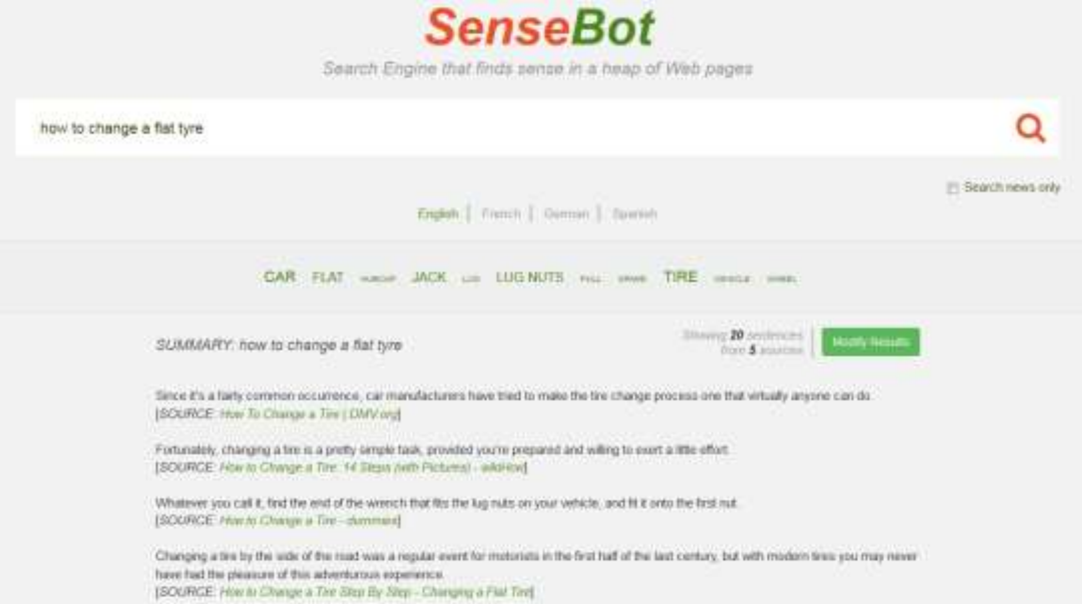


Cairo, which means "the Vanquisher" or "the Triumphant", is the capital and largest city of Egypt. It is the Arab World's largest and Africa's second most populous city. While Al-Qahirah is the official name of the city, in Egyptian Arabic it is called by the dialect's name for the country, transliteration: Masr. Cairo was founded by the Fatimid caliphs as a royal enclosure. It replaced Fustat as the seat of the government. It later came under the Mamluks, was ruled by the Ottomans 1517 to 1798, and briefly occupied by Napoleon.

Population: 6.758 Million (2008)

Figure 4: Kngine Search results for "Cairo"

SenseBot is a semantic search engine that works based on page summarisation, and as implied by its name, tries to find the sense contained in the web pages (Radhakrishnan, 2007). SenseBot returns search results as a summary of the topic pertaining to the user's search query instead of a list of links to other web pages. When a user submits a search query to SenseBot, it will first search for a list of pages that match the query. However, before the search results are returned to the user, SenseBot will perform analysis on the list of the results first by using text-mining technology, which identifies key concepts contained in the web pages semantically. Then a multi-document summarisation is performed to produce a summary of topics that are related to the user's query, which will be returned to the user. A sentiment analysis service is also provided by SenseBot, which can display information on which users are responding to a certain topic. Figure 5 shows the search results obtained for a query on "how to change a flat tyre". The result page shows some keywords that are related to the user's query on top of the page. SenseBot identified "car", "wheel", "jack", and "tire" as some of the important keywords in this query (these keywords appear larger than the rest). The summary displays a list of entries that actually explains "how to change a flat tyre". SenseBot chose to present the results in this form to move away from the conventional search engines that tend to return so many links. However, not all the links can provide useful information and to overcome this issue, SenseBot will drop a page when the content is considered not relevant to the user's query (even when that page is ranked highly during the summarisation process). The search results provided are sufficient to provide an overview of the topic desired by the user and in turn, users click on less links to further explore the topic.


SenseBot
Search Engine that finds sense in a heap of Web pages
how to change a flat tyre
□ Search news only
English | French | German | Spanish
CAR FLAT HUBCAP JACK LUG LUG NUTS PULL SPARE TIRE VEHICLE WHEEL
SUMMARY: how to change a flat tyre
Showing 20 sentences from 5 sources
Modify Results

Since it's a fairly common occurrence, car manufacturers have tried to make the tire change process one that virtually anyone can do.
[SOURCE: How To Change a Tire | DMV.org]

Fortunately, changing a tire is a pretty simple task, provided you're prepared and willing to exert a little effort.
[SOURCE: How to Change a Tire: 14 Steps (with Pictures) - wikiHow]

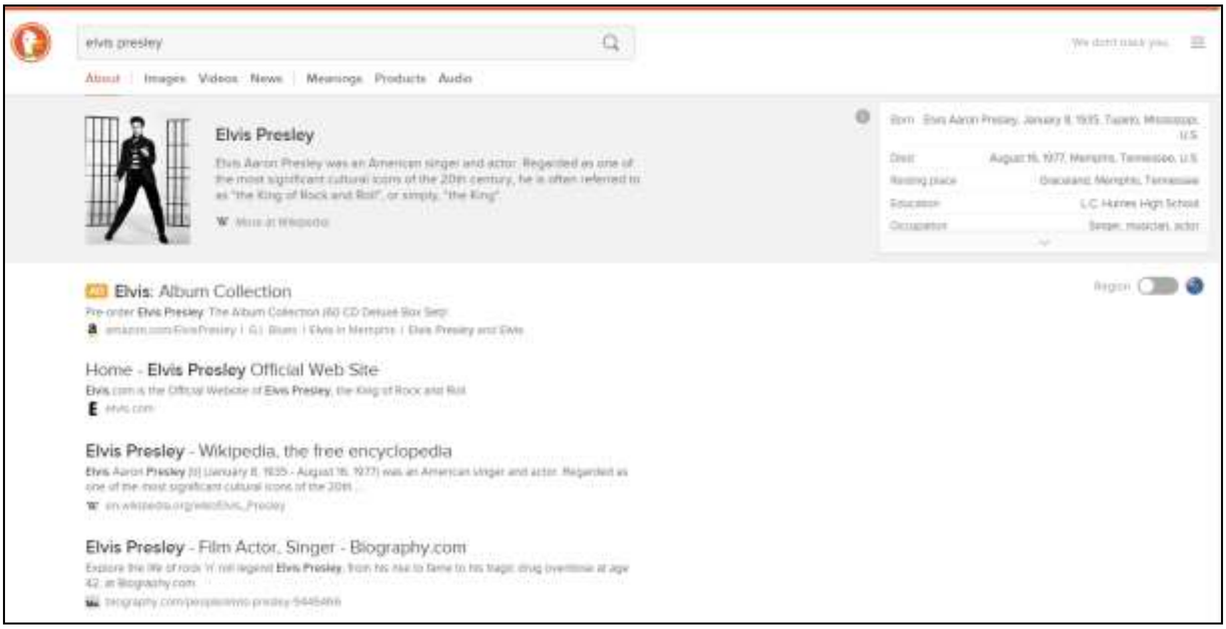
Whatever you call it, find the end of the wrench that fits the lug nuts on your vehicle, and fit it onto the first nut.
[SOURCE: How to Change a Tire - dummies]

Changing a tire by the side of the road was a regular event for motorists in the first half of the last century, but with modern tires you may never have had the pleasure of this adventurous experience.
[SOURCE: How to Change a Tire Step By Step - Changing a Flat Tire]

Figure 5: SenseBot search result for "how to change a flat tyre"

Another work based on semantic web search engine is Look4 (Avetisyan & Avetisyan, 2010). It is a meta search engine that aims to enhance the result from a traditional search engine by using a network of concepts that are built from the WordNet ontology (Miller, 1990). The categorisation of the traditional search is based on statistics and hence only covers common topics while less popular topics are most probably ignored, which decreases the coverage of all possible meanings. The main idea behind Look4 is to understand what kind of information a user is looking for, with the help of ontology and natural language processing. For example, if a user searches for a "horse", the system will list down possible meanings of "horse" for the user to select from. This type of communication enables Look4 to ensure the actual topic required by the user before sending the information to search engines like Yahoo or Google. Look4 was able to achieve an increase of approximately 60% in precision compared to Google, Yahoo, and Bing for a set of 50 selected keywords.

DuckDuckGo is a semantic search engine for general purpose search (Sudeepthi *et al.*, 2012; Hands, 2012). It provides a scaled down search, spam-free results, instant answers, and protection to user privacy. The search results are presented as snippets according to title, metadata, description and URL in an uncluttered presentation with favicon associated with the particular web. The instant answer appears on the top of the search results' page in a grey box section. The instant answer section provides a longer snippet from the search topic, because this section provides a collection of answers from the best group of resources. For example, if the search query is on "calculations" or "math", the results might come from WolframAlpha. If the query is a definition, then the results might come from Wikipedia, whereas if the query is related to map information, the results might come from Google Map. DuckDuckGo utilises a third party web crawler and indexing to maintain repository information. It also provides site searchers with three different filters, which are the filetype, inbody, and intitle. DuckDuckGo offers privacy protection by not collecting or sharing personal information of the users during the query process. Figure 6 shows the search results obtained using the query "Elvis Presley".



Elvis Presley

Elvis Aaron Presley was an American singer and actor. Regarded as one of the most significant cultural icons of the 20th century, he is often referred to as "the King of Rock and Roll", or simply, "the King".

W More at Wikipedia

Born	Elvis Aaron Presley, January 8, 1935, Tupelo, Mississippi, U.S.
Died	August 16, 1977, Memphis, Tennessee, U.S.
Resting place	Graceland, Memphis, Tennessee
Education	L.C. Humes High School
Occupation	Singer, musician, actor

AD **Elvis**: Album Collection

Pre-order **Elvis Presley**: The Album Collection (60 CD Deluxe Box Set)!

amazon.com/ElvisPresley | G.I. Blues | Elvis in Memphis | Elvis Presley and Elvis

Home - **Elvis Presley** Official Web Site

Elvis.com is the Official Website of **Elvis Presley**, the King of Rock and Roll

elvis.com

Elvis Presley - Wikipedia, the free encyclopedia

Elvis Aaron **Presley** (/ (January 8, 1935 - August 16, 1977) was an American singer and actor. Regarded as one of the most significant cultural icons of the 20th ...

en.wikipedia.org/wiki/Elvis_Presley

Elvis Presley - Film Actor, Singer - Biography.com

Explore the life of rock 'n' roll legend **Elvis Presley**, from his rise to fame to his tragic drug overdose at age 42, at Biography.com.

biography.com/people/elvis-presley-9446466

Figure 6: DuckDuckGo Search results for "Elvis Presley"

Koru is a semantic search engine that harnesses Wikipedia to provide domain-independent knowledge based retrieval (Milne, Witten, & Nichols, 2007). The research focuses on two objectives, which are to produce thesauri through an automated process as well as to apply thesauri in the search process. It involves both creation and utilisation of a knowledge base that enhances searching. The set of thesauri is generated by extracting significant relationships within Wikipedia articles. An example of measurement considered in extracting article relationships is the link structure. It is based on the assumption that if many articles link to a single article in Wikipedia, then the relationships between all the linked articles may be weak. However, if only a few articles are linked to a unique article, then the weight of those links will be higher in measuring the relatedness between articles. The main functionality of Koru is that it can expand user's search query based on the information extracted from Wikipedia. As a result, the query results may contain topics that are not found in the user query. For example, query for "air carrier" will return results that may contain the terms "airline" as both concepts are similar. Unfortunately, Koru is limited to the use of the knowledge base that contains similar word relationships to retrieve related documents. The research is also focused on using local documents only and hence it is not known how Koru can be scaled up for the web without losing accuracy and precision of the search results.

Google has also moved to semantic search with the launch of Google Hummingbird algorithm in September 2013. The main idea behind this algorithm is for Google to provide more precise and accurate search results based on the meaning of the search query (context). Omnity is the latest semantic search engine that has been made public in January 2016. Omnity is seen as the next-generation semantic search and discovery tool to discover hidden and high value interconnections between different fields of knowledge.

DBPedia

DBPedia provides information written in RDF format and it uses standards such as RDFS and OWL to represent the relationships between its resources. Any software agent that complies with the semantic web standard can perform the basic inference process on DBPedia's content. DBPedia.org is a community that tries to extract structured information from Wikipedia so that its information can be queried just like a database (Auer *et al.*, 2007; Morsey *et al.*, 2012). The project was started in 2007 with a total of 1.95 million concepts. The number of concepts grew to 2.6 million in 2009, and by January 2016, DBPedia dataset describes about 4.58 million things and over 1 billion facts. This means that there is a large amount of potential knowledge source that can be extracted and utilised whereby more accurate and precise search results can be delivered to the user. The extraction framework for DBPedia consists of a component called extractor where it processes different types of content in Wikipedia. This extraction process is similar to the crawling process that takes place in a traditional search engine. However, each type of structured information is handled by a different extractor that knows the pattern of the desired information. Types of content that can be processed include labels, abstracts, interlanguage links, images, redirects, disambiguation, external links, pagelinks, categories, geo-coordinates, and infobox.

Figure 7 shows an overview of the processes involved in extracting structured information from Wikipedia. The extracted data is stored as N-triple dumps, which is then loaded to Virtuoso Triple Store to be queried. The data dump is available freely and can be downloaded by the general public.

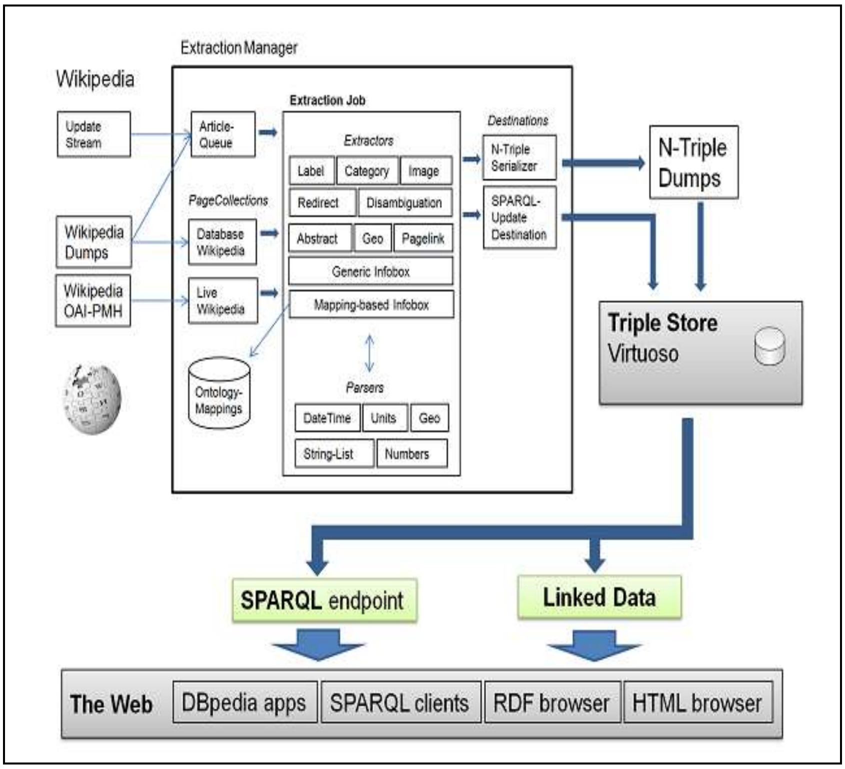


Figure 7: Extracting structured information from DBPedia (Auer *et al.*, 2007)

META SEARCH ENGINE FRAMEWORK

In this work, the proposed meta semantic search technique tries to understand the meaning of the query and expands it to include relevant concepts. This will allow for a wider search area. Five datasets in DBPedia were used to enrich the search results namely, the Article to Category Dataset, the Disambiguation dataset, the Labelling dataset, the Page link dataset, and the Redirect dataset.

Article to Category dataset

This dataset shows the relationship between concept and the classes that it belongs to and is encoded as *skos:Concept*, which describes the conceptual or intellectual structure of a knowledge organisation system defined by W3C. Based on this information, it is possible to figure out which domain the search query should belong to. For example, given a term such as “artificial intelligence”, this dataset can provide related concepts such as “cybernetics” and “formal sciences”. Using these classes, it is possible to extend it again by searching for the sibling concepts of the user’s query.

Disambiguation dataset

This dataset enables the extraction of the possible meanings of a query. For example, the term “gamma” can refer up to ten concepts like “gamma wave”, “gamma correction”, and “gamma distribution”.

Labelling dataset

This dataset shows how a certain concept can be referred to in other languages. This allows concept expressed in other languages to be included in the search query expansion.

Page link dataset

A Wikipedia page can be linked to another Wikipedia page to allow users to browse another topic from an existing page. The link structure in Wikipedia is considered a valuable resource for evaluating concepts. We can perform analysis on the link structure (similar to a PageRank analysis) and identify the core topic. Moreover, if a concept is linked to another concept, it is possible that the two concepts are related. As an example, the concept "artificial intelligence" is linked to "knowledge representation" and "logic programming" pages. We can use this information to enrich and expand the user's query.

Redirect dataset

The redirect dataset is the opposite of the disambiguation dataset. While disambiguation dataset provides multiple concepts for a single word, redirect identifies words that can refer to the same concept. For example, "AI" and "artificial intelligence" refer to the same concept as both terms are bounded under the redirect dataset. This dataset makes it possible to guess the meaning of a user's query so that other related information like category and page link can be queried.

The user's search query will be parsed by a query analyser. This query is parsed based on the five DBPedia datasets and all the relevant concepts will be used to generate extended queries. These extended queries are then sent to Google search engine. In this work we have chosen Google as the traditional search engine. However, other traditional search engines could be used for this purpose. The search results are parsed and ranked according to different clusters before they are presented to the user. Figure 8 shows the overview of MEOW. MEOW has four main modules, which are query parser, query reconstruction, result parser, and ranking.

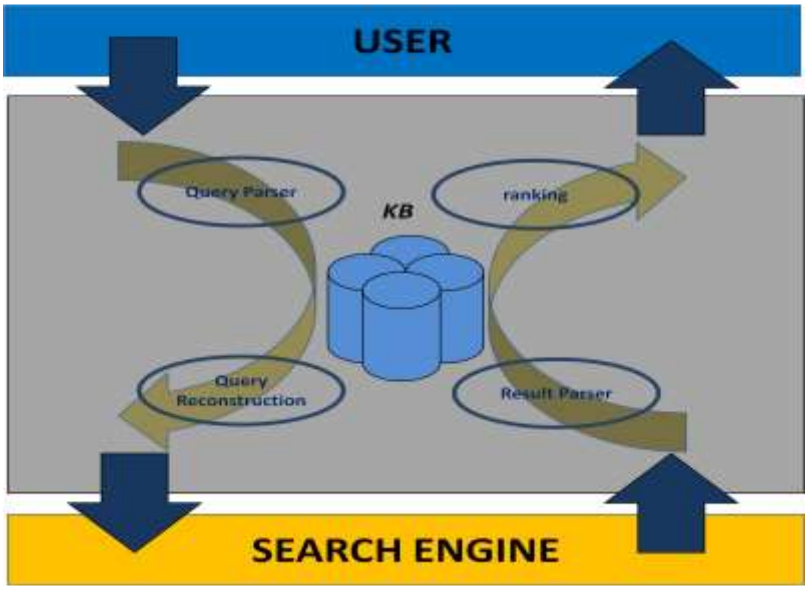


Figure 8: MEOW System Overview

The Query Parser

The query parser identifies the meaning of the user's search query using DBPedia dataset. The query is checked against the redirect dataset to discover possible meanings of the search query. It also explores the page link dataset to find all the related concepts for the search query. A search query of "AI" will also consider the concept "artificial intelligence". As a result, all the related concepts for "artificial intelligence" will also be considered. The disambiguation and the category dataset are also used to guess the possible meaning of the search query. The final output of this query parsers are all the possible concepts for the user's search query.

Query Reconstruction

This module sends all the related concepts generated by the query parser to Google. These concepts are sent to Google to get initial results. Based on these results, if the candidate returned from the page link dataset can be mapped into this result, it is assumed that this candidate is closely related to the user's query. For example, the page "cat" is linked to both "cat behaviour" and "ancient Egypt" in Wikipedia. However, since "Egypt" is not in the search results returned by Google, it is unlikely that "Egypt" is related to "cat". Therefore, in this case, "Egypt" is excluded. On the other hand, if "cat behaviour" is listed in the search results, then it should be considered in the query reconstruction process. Using this information, extended queries are submitted to Google.

The Result Parser

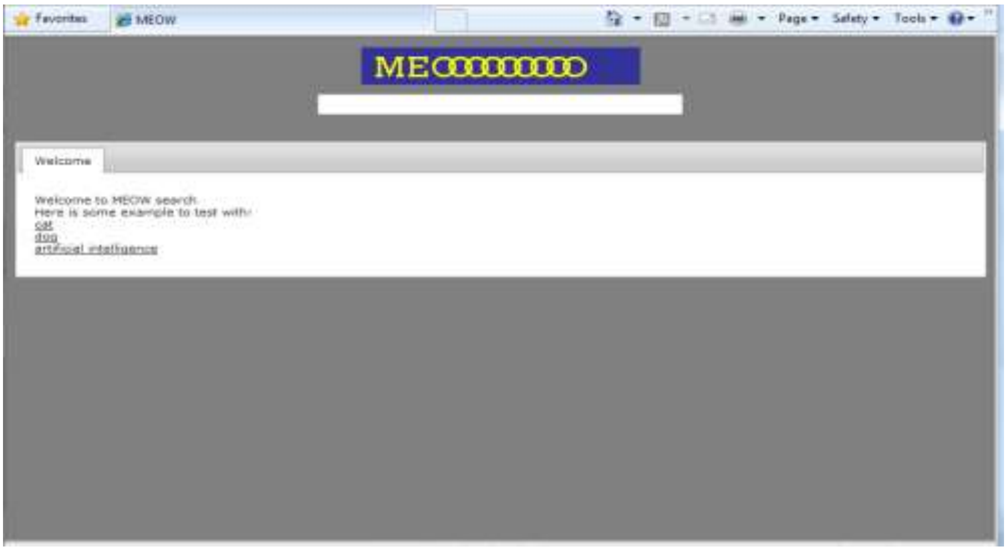
The result parser processes Google's search results based on the extended queries in clusters. These clusters are then merged into a single page of search results and the ordering of each entry is performed by the ranking module.

The Ranking Module

The results for each cluster are ranked based on the cluster's title and the significance of the concept in a DBPedia link structure. The significance of the concept is the weight given to the concept based on PageRank analysis. PageRank is computed for each concept in DBPedia based on the page link dataset. However, in this process, the weight of each concept is the inverse value of PageRank. A page that is referred by many other pages from Wikipedia will obviously obtain a higher PageRank value, because a specific topic will not likely be linked to many pages. If a concept is linked by almost every page in Wikipedia, then most likely, the concept is very common and as such should not be considered in the ranking process and therefore should have a lower PageRank value. Using the inverse value of PageRank ensures that all unique concepts are ranked higher in the final results. The final search results are presented to the user based on article category and labelling dataset.

MEOW IN ACTIONS

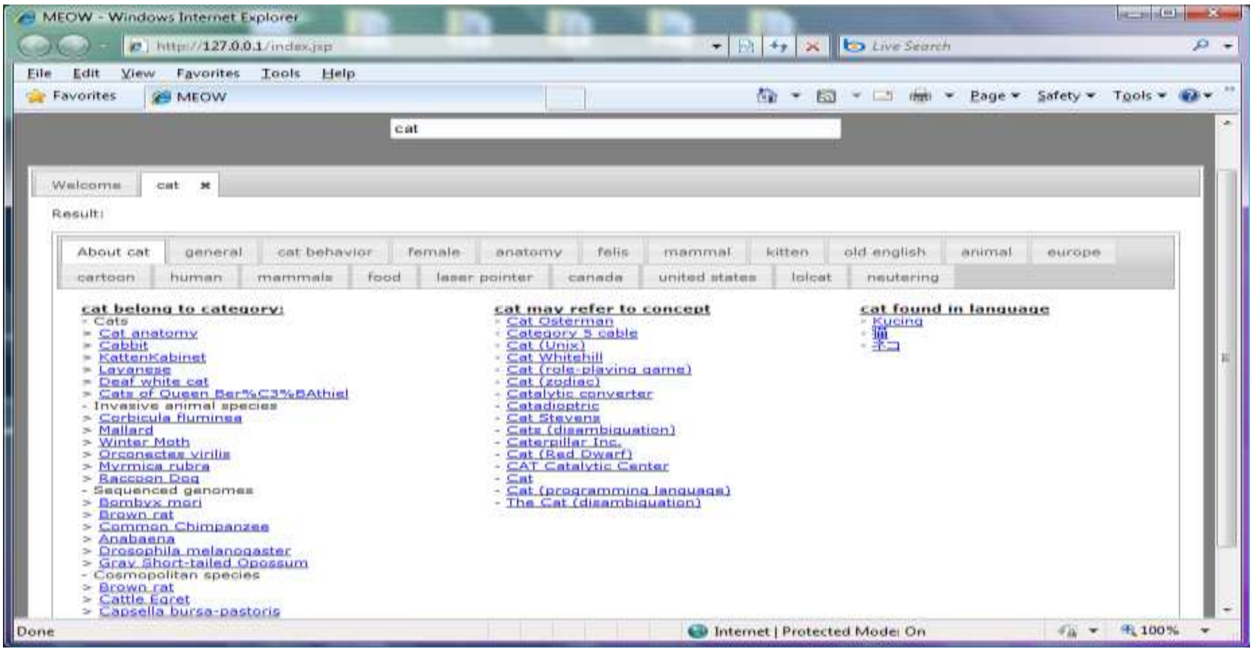
MEOW service was hosted using "tomcat" and the service is a combination of Java and JSP. The service acts as a bridge between the user (client) and Google. Figure 9 shows MEOW's search interface. The user can enter his/her query in the text field provided at the top of the page; the results are presented in the tab section below the text field.



Welcome to MEOW search.
Here is some example to test with:
cat
dog
artificial intelligence

Figure 9: MEOW's search interface

We adopted the clustering concept used in the Yippy search engine to group search results into topics that may be relevant to the search query. A tab is created for each search query and the results are also clustered into tabs. This tab functionality is provided to allow users to refer to the previous search without having to reload the content again. Figure 10 shows a search for "cat" for the "About cat" where it offers a list of suggestions related to the "cat" concept that the user can choose from. It can also be observed that the different tabs contain different concepts related to "cat".

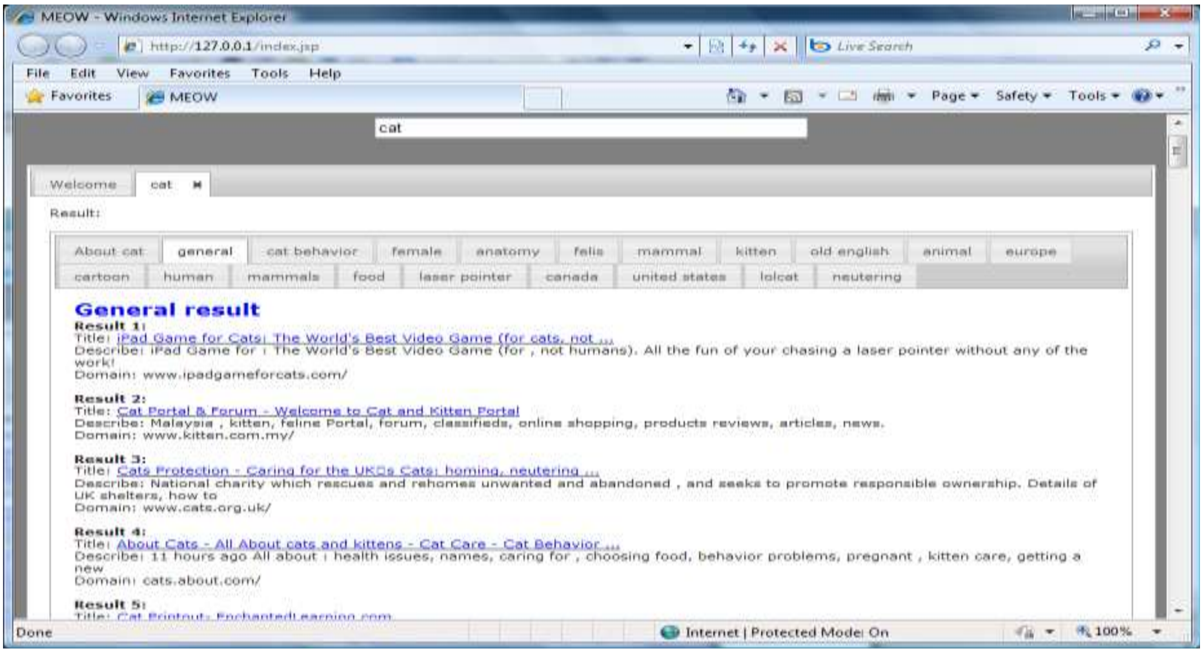


cat belong to category:
- Cats
> Cat anatomy
> Cabbit
> KattenKabinet
> Lavanese
> Deaf white cat
> Cats of Queen Ber%C3%BAthiel
- Invasive animal species
> Corbicula fluminea
> Mallard
> Winter Moth
> Orconectes virilis
> Myrmica rubra
> Raccoon Dog
- Sequenced genomes
> Bombyx mori
> Brown rat
> Common Chimpanzee
> Anabaena
> Drosophila melanogaster
> Gray Short-tailed Opossum
- Cosmopolitan species
> Brown rat
> Cattle Egret
> Capsella bursa-pastoris

cat may refer to concept
- Cat Osterman
- Category 5 cable
- Cat (Unix)
- Cat Whitehill
- Cat (role-playing game)
- Cat (zodiac)
- Catalytic converter
- Catadioptric
- Cat Stevens
- Cats (disambiguation)
- Caterpillar Inc.
- Cat (Red Dwarf)
- CAT Catalytic Center
- Cat
- Cat (programming language)
- The Cat (disambiguation)

cat found in language
- Kucing
- 猫
- ネコ

Figure 10: Search results for "cat"



General result

Result 1:
Title: iPad Game for Cats: The World's Best Video Game (for cats, not ...
Describe: iPad Game for : The World's Best Video Game (for , not humans). All the fun of your chasing a laser pointer without any of the work!
Domain: www.ipadgameforcats.com/

Result 2:
Title: Cat Portal & Forum - Welcome to Cat and Kitten Portal
Describe: Malaysia , kitten, feline Portal, forum, classifieds, online shopping, products reviews, articles, news.
Domain: www.kitten.com.my/

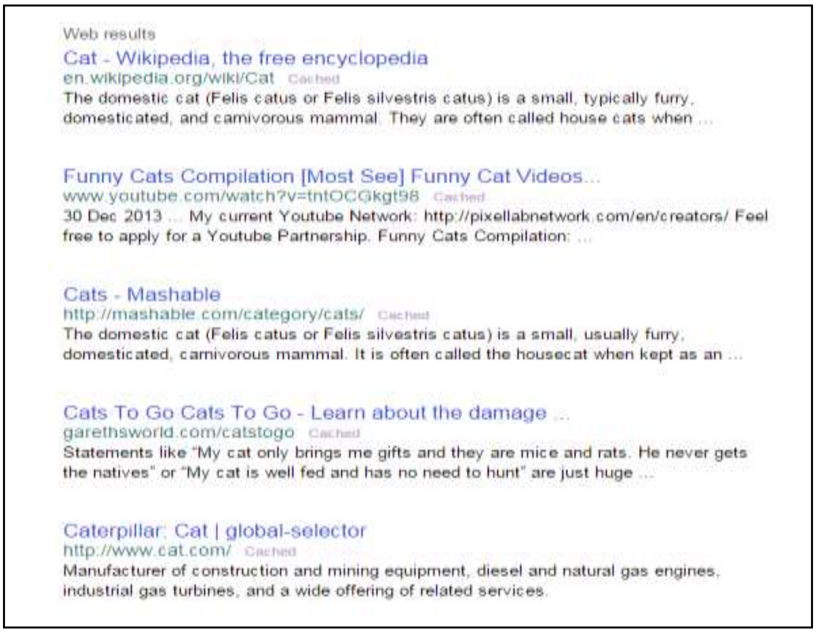
Result 3:
Title: Cats Protection - Caring for the UK□s Cats: homing, neutering ...
Describe: National charity which rescues and rehomes unwanted and abandoned , and seeks to promote responsible ownership. Details of UK shelters, how to
Domain: www.cats.org.uk/

Result 4:
Title: About Cats - All About cats and kittens - Cat Care - Cat Behavior ...
Describe: 11 hours ago All about : health issues, names, caring for , choosing food, behavior problems, pregnant , kitten care, getting a new
Domain: cats.about.com/

Result 5:
Title: Cat Printout- EnchantedLearning.com

Figure 11: The general results for MEOW search

Figure 11 shows the general results for "cat" that have been ranked using our ranking algorithms. This result can be compared to the search results obtained from Yahoo as shown in Figure 12. We have decided to show Yahoo's search results as opposed to Google as we are using Google as the search engine. It can be seen that the results obtained are different and that MEOW offered more options around how the concept should be explored. It can also be seen that Yahoo's top search is the "cat" Wikipedia page while MEOW's top search result is "iPad Game for Cats", followed by "cat Portal & Forum". In addition, MEOW also has additional tabs that user can use to explore various topics relevant to the concept such as "cat behaviour", "kitten", and "animal". Figure 13 shows the screen shot for the "cat behaviour" cluster. Figure 14 shows another query with "gamma correction" using MEOW.



Web results

Cat - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Cat Cached
The domestic cat (Felis catus or Felis silvestris catus) is a small, typically furry, domesticated, and carnivorous mammal. They are often called house cats when ...

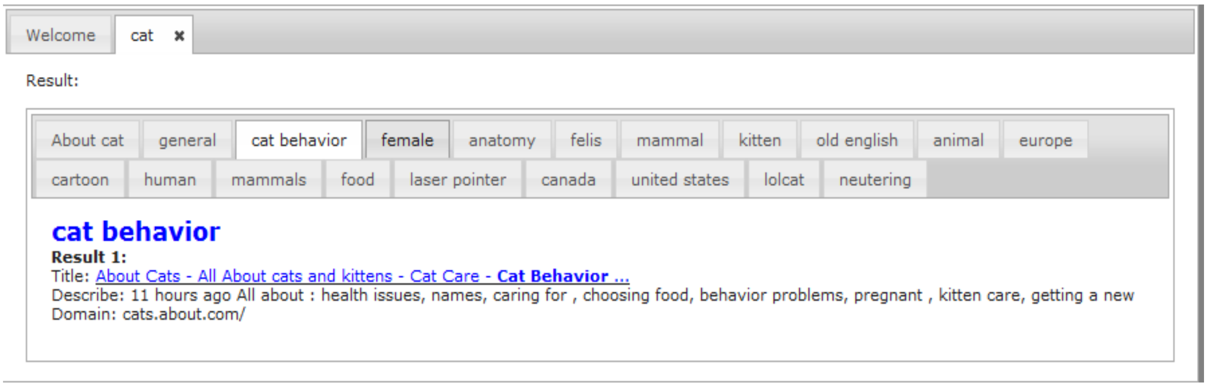
Funny Cats Compilation [Most See] Funny Cat Videos ...
www.youtube.com/watch?v=tntOCGkgt98 Cached
30 Dec 2013 ... My current Youtube Network: http://pixellabnetwork.com/en/creators/ Feel free to apply for a Youtube Partnership. Funny Cats Compilation: ...

Cats - Mashable
http://mashable.com/category/cats/ Cached
The domestic cat (Felis catus or Felis silvestris catus) is a small, usually furry, domesticated, carnivorous mammal. It is often called the housecat when kept as an ...

Cats To Go Cats To Go - Learn about the damage ...
garethsworld.com/catstogo Cached
Statements like "My cat only brings me gifts and they are mice and rats. He never gets the natives" or "My cat is well fed and has no need to hunt" are just huge ...

Caterpillar: Cat | global-selector
http://www.cat.com/ Cached
Manufacturer of construction and mining equipment, diesel and natural gas engines, industrial gas turbines, and a wide offering of related services.

Figure 12: Yahoo's search result for "cat"

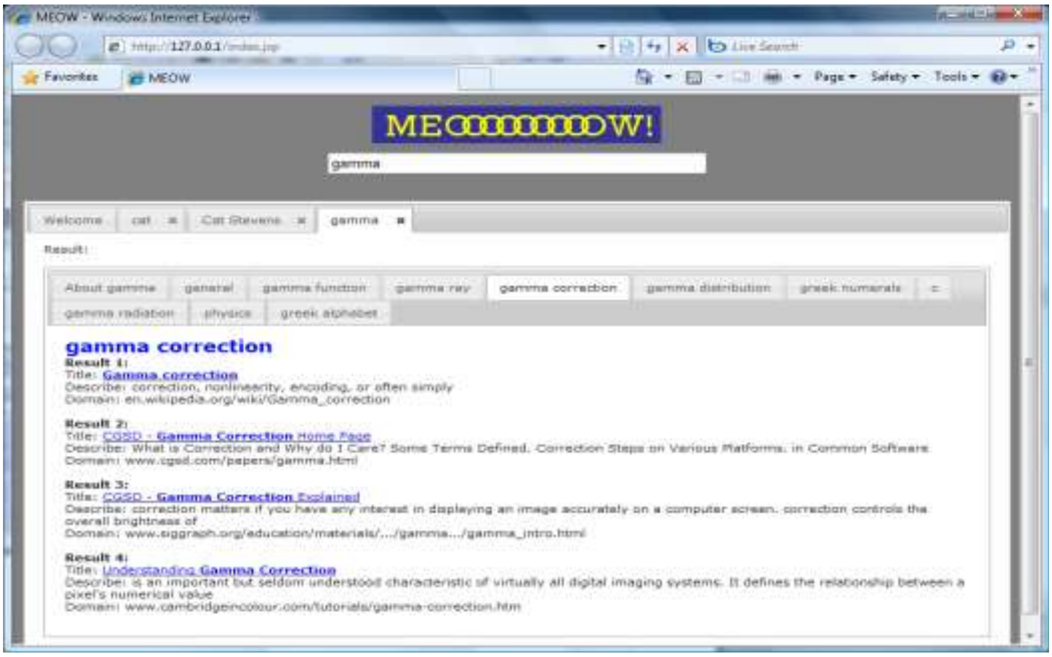


Result:

cat behavior

Result 1:
Title: About Cats - All About cats and kittens - Cat Care - **Cat Behavior ...**
Describe: 11 hours ago All about : health issues, names, caring for , choosing food, behavior problems, pregnant , kitten care, getting a new
Domain: cats.about.com/

Figure 13: The results for "cat behaviour" cluster



Result:

gamma correction

Result 1:
Title: **Gamma correction**
Describe: correction, nonlinearity, encoding, or often simply
Domain: en.wikipedia.org/wiki/Gamma_correction

Result 2:
Title: CGSD - **Gamma Correction** Home Page
Describe: What is Correction and Why do I Care? Some Terms Defined. Correction Steps on Various Platforms. in Common Software
Domain: www.cgsd.com/papers/gamma.html

Result 3:
Title: CGSD - **Gamma Correction** Explained
Describe: correction matters if you have any interest in displaying an image accurately on a computer screen. correction controls the overall brightness of
Domain: www.siggraph.org/education/materials/..../gamma.../gamma_intro.html

Result 4:
Title: Understanding **Gamma Correction**
Describe: is an important but seldom understood characteristic of virtually all digital imaging systems. It defines the relationship between a pixel's numerical value
Domain: www.cambridgeincolour.com/tutorials/gamma-correction.htm

Figure 14: The "gamma correction" cluster for "gamma" query

MEOW also provides suggestions for what a user can browse. For example, a query with the search string "Obama" will suggest links for "Blanche Bruce", "Edward Brooke", and "Roland Burris". These names were suggested, because they fall under the category of "African American United States Senators".

CONCLUSION

Although a pure semantic web may take a long time to be implemented, it is possible to build a web search engine that partially performs semantic search. This work investigates the possibility of implementing a semantic search on Web 2.0 content by utilising DBPedia as the knowledge base to enrich non-semantic search results. DBPedia as a free source of information describes millions of concepts making it possible to identify the meaning of a search query. A self-maintained knowledge base is not considered in this work, because it will make the development of the search engine more complicated. Moreover, DBPedia is already a well-known resource that are referred to by many web sites in linked data and therefore using DBPedia will make the interaction with other resources in linked data easier in the future.

In this work, the DBPedia datasets that contributed to most of the enrichment are the article category, disambiguation, and the page link datasets. Article category dataset provides sibling concept for a query wherein a search for "mercury" may also include "Venus", since these two concepts are defined under the category "planet in the solar system". Disambiguation dataset is mainly used for resolving the meaning of the query as there is more than one definition found in DBPedia. Finally, the page link dataset is used to identify relationships between concepts.

The advantages of MEOW search engine is that it can act as a combination of concept browser as well as a web information service. If the re-ranking is ignored, then MEOW offers Google search results with the flexibility of browsing the information based on Wikipedia content. Presently, MEOW is only able to handle a single word query and more work needs to be done to accommodate complex queries. MEOW can be further enhanced by making use of the other datasets such as the mapping dataset for location based search.

ACKNOWLEDGEMENTS

This project is funded by the Ministry of Higher Education, Malaysia under FRG0303-TK-1-2012.

REFERENCES

[1] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J. & Ives, Z. (2007). DBpedia: A Nucleus for A Web of Open Data. *Proceedings of the 6th International Semantic Web Conference*. 11-15 November, 2007. Busan, Korea. pp. 722-735.

[2] Antonious, G. & Harmelen, F. V (2008). *A Semantic Web Primer*. Cambridge.

[3] Avetisyan, A. & Avetisyan, V. (2010). LOOK4: Enhancement of Web Search Results with Universal Words and WordNet. *Proceedings of the 5th International Conference on Global WordNet*. 31 Jan - 4 Feb, 2010. pp. 1-5.

[4] Baeza-Yates, R. & Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Harlow.

[5] Berners-Lee, T. (2000). *Weaving the Web - the Original Design and Ultimate Destiny of the World Wide Web*. New York.

[6] Berners-Lee, T. Hendler, J. & Lassila, O. (2001). The Semantic Web. *Scientific American*. May 2001, 29-37.

[7] Brickley, D. & Guha, R. V. (2004). RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation. Retrieved Feb 12, 2016 from https://www.w3.org/TR/2004/REC-rdf-schema-20040210/

[8] Boo, V. K. & Anthony, P. (2011). Meta Search Engine Powered by DBpedia. In *International Conference on Semantic Technology and Information Retrieval (STAIR)*. 28 – 29 June, 2010. pp. 89-93

[9] Brin, S. & Page, L. (1998). The Anatomy of A Large-Scale Hypertextual Web Search Engine. *Computer Networks*, **30**(1–7). pp. 107-117

[10] Davies, J. Studer, R. & Warren, P. (2006). *Semantic Web Technologies: Trends and Research in Ontology-based Systems*. West Sussex: Wiley.

[11] Doan, B. L. Bourda, Y. & Bennacer, N. (2004). Using Owl to Describe Pedagogical Resources. *Proceedings of the IEEE International Conference on Advanced Learning Technologies*. Joensuu, Finland. pp. 916 – 917.

[12] Hands, A. (2012). Duckduckgo http://www.duckduckgo.com or http://www.ddg.gg. *Technical Services Quarterly*, **29**(4). pp.345-347.

[13] Koivunen, M. R. & Miller, E. (2001). W3C Semantic Web Activity. *Proceedings of the Semantic Web Kick-off Seminar*. 2 Nov 2001. Helsinki, Finland. pp. 27-41.

[14] Kumar, R. & Goel, R. (2012). A Detailed Study on Semantic Search Performance of Keyword and Meta Search Engines. *International Journal of Computer Science and Information Technologies*, **3**(2). pp. 3655-3658.

[15] Klyne, G. & Carroll, J. J. (2004). Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation. Retrieved Feb 12, 2016 from https://www.w3.org/TR/2004/REC-rdf-concepts-20040210/

[16] Manola, F. & Miller, E. (2004). RDF Primer, W3C Recommendation. Retrieved Feb 12, 2016 from https://www.w3.org/TR/2004/REC-rdf-primer-20040210/

[17] McGuinness D. L. & Harmelen, F. V. (2004*)*. OWL Web Ontology Language Overview. W3C Recommendation. Retrieved Feb 12, 2016 https://www.w3.org/TR/owl-features/

[18] Miller, G. A. Beckwith, R. Fellbaum, C. Gross, D. & Miller, K. J. (1990). WordNet: An On-line Lexical Database. *International Journal of Lexicography*, **3**(4). pp. 235–312.

[19] Milne, D. N. Witten, I. H. & Nichols, D. M. (2007). A Knowledge-Based Search Engine Powered by Wikipedia. *Proceedings of the 16th ACM Conference on Information and Knowledge Management.* New York. 6 – 10 Nov, 2007. pp. 445-454.

[20] Morsey, M. Lehmann, J. Auer, S. Stadler, C. & Hellmann, S. (2012). DBpedia and The Live Extraction of Structured Data from Wikipedia. *Program: Electronic Library and Information Systems*, **46**(2). pp. 157-181.

[21] Radhakrishnan, A. (2007). Summarization, the Answer to Web Search: Interview with Dmitri Soubbotin of SenseBot. Retrieved Feb 12, 2016 from https://www.searchenginejournal.com/summarization-the-answer-to-web-search-interview-with-dmitri-soubbotin-of-sensebot/6094/

[22] Rector, A. Drummond, N. Horridge, M. Rogers, J. Knublauch, H. Stevens, R. Wang, H. & Wroe, C. (2004). Owl Pizzas: Practical Experience of Teaching OWL-DL:Common Errors & Common Patterns. *Proceedings of EKAW 2004*. 5 – 8 Oct, 2004. Whittlebury Hall, UK. pp 63-81.

[23] Salton, G. & McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. New York.

[24] Singh, J., & Sharan, A. (2013). A Comparative Study between Keyword and Semantic Based Search Engines. *Proceedings of the International Conference on Cloud, Big Data and Trust 2013*. 13 – 15 Nov, 2013. pp. 130-134.

[25] Spetka, S. (1994). The TkWWW Robot: Beyond Browsing. *Proceedings of the Second World Wide Web Conference: Mosaic and the Web*. Chicago, Illinois.

[26] Strickland, M. (2007). The Evolution of Web 3.0. Retrieved Feb 12, 2016, from http://www.slideshare.net/mstrickland/the-evolution-of-web-30

[27] Sudeepthi, G. Anuradha, G. Surendra, P. & Babu, P. (2012). A Survey on Semantic Web Search Engine. *International Journal of Computer Science Issues,* **9**(2). pp. 241 - 245.

[28] Tumer, D., Shah, M. A., & Bitirim, Y. (2009). An Empirical Evaluation on Semantic Search Performance of Keyword-Based and Semantic Search Engines: Google, Yahoo, MSN and Hakia, *Proceedings of the 4th International Conference on Internet Monitoring and Protection (ICIMP'09)*. 24 - 28 May, 2009. Venice, Italy. pp. 51-55.