EVOLUTIVE CONTENT-BASED SEARCH SYSTEM

Semantic Search System based on Case-based-Reasoning and Ontology Enrichment

Manel Elloumi-Chaabene, Nesrine Ben Mustapha, Hajer Baazaoui-Zghal
Riadi Lab., ENSI Campus Universitaire de la Manouba, 2010 Tunis, Tunisie

Antonio Moreno, David Sánchez
ITAKA Research Group, Departament d'Enginyeria Informàtica i Matemàtiques
Universitat Rovira i Virgili, Av. Països Catalans, 26. 43007, Tarragona, Spain

Keywords: Information Retrieval, Semantic search, Ontology, Case-based-Reasoning.

Abstract: The exponential growth of the data available on the Web requires more efficient search tools to find relevant information. In the context of the Semantic Web, ontologies improve the exploitation of Web resources by adding a consensual knowledge. However, the automation of ontology building is a hard problem. The exploitation of the users' search feedback can aid to address that problem. In order to apply this solution, we present a semantic search system based on case-based-reasoning (CBR) and ontologies that automatically enriches the ontologies by using previous queries. In this paper we expose the contribution of CBR and ontologies in Semantic Web search. Some experiments on the system are presented, and the obtained results show an improvement on the precision of the Web search and ontology enrichment.

1 INTRODUCTION

Because of the enormous size of the textual information available in the Web, the use of traditional search engines does not always provide the most appropriate and relevant results. In fact, the problem of *Information Retrieval (IR)* consists on finding the right information corresponding to the user query. Indeed, the information available on a Web page is not always well-structured, which makes it difficult to extract knowledge. Thus, we are confronted with problems of interpretation of human language and treatment of ambiguities (polysemy, anaphora, metaphor), which are difficult to be automated. The inclusion of the semantics provided by ontologies in the IR process can help to improve the quality of the search results.

In order to avoid the necessity of manually constructing ontologies, in this work we address the *automatic construction of multi-domain ontologies* for Semantic Web search. The basic idea is to use the Web search feedback provided by users to construct ontologies from the consulted Web documents (Ben Mustapha et al., 2009a). The challenge behind this work is to find the relevant documents from which the ontology should be constructed. *Case-based reasoning (CBR)* is used to find appropriate documents related to a new query according to a case base. In this way, ontologies can be automatically enriched as new search feedback is obtained, whereas the Web search can be improved by the semantics provided by the ontologies.

This work presents an implementation of a generic approach (Ben Mustapha, 2010) which allows any search engine to develop its semantic layer by building ontologies and indexing the document base. Our contribution consists in integrating CBR with ontologies to ameliorate the semantic search system and automatic ontology learning.

In this paper, we present an overview of related works in the area of semantic search and ontology learning. The third section describes the architecture of the proposed content-based search engine. The next section illustrates the proposed approach by a use case and describes some evaluation tests. Finally, we conclude and discuss directions for future research.

Elloumi-Chaabene M., Ben Mustapha N., Baazaoui-Zghal H., Moreno A. and Sánchez D..
EVOLUTIVE CONTENT-BASED SEARCH SYSTEM - Semantic Search System based on Case-based-Reasoning and Ontology Enrichment .
DOI: 10.5220/0003067600240034
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2010), pages 24-34
ISBN: 978-989-8425-28-7

2 SEMANTIC WEB SEARCH

Semantic search has been one of the motivations of the Semantic Web (Berners-Lee, 2001) since it was envisioned as an extension of the current one, in which information is given well defined meaning, better enabling computers and people to work in cooperation. It is emerging as the next generation Web, which aims to give a semantic representation which is accessible and understandable by machines. It is based on the deployment of ontologies, the semantic annotations and the formal representation of the knowledge underlying these ontologies and annotations. The role of semantic search is to use underlying ontologies and available metadata in semantic Web portals provide better performance than keyword-based search.

2.1 Ontologies

Ontologies (Guarino, 1998) are used to represent a shared conceptualization of a knowledge domain, adding a semantic layer to computer systems. In other words, an ontology is a formal representation of the main concepts in a domain and their interrelationships. An ontology can take the simple form of a taxonomy (i.e., knowledge encoded in a minimal hierarchical structure) or a vocabulary with standardized machine interpretable terminology supplemented with natural language definitions. Ontologies are often specified in a declarative form by using semantic markup languages such as XML (Bradley, 2001), RDF (Lassila, 1999), OIL (Decker, 2000), XOL (Karp 1999) and OWL (McGuinness, 2004).

Several types of ontologies exist:

- *Generic or Common Ontologies*: they model the common sense knowledge reusable between domains. These ontologies include vocabulary related to things, events, time, space, causality, behaviour, function, etc.
- *Application Ontologies*: they model the knowledge required for specific applications. Ontology applications often specialize the vocabulary of domain ontologies and task ontologies.
- *Knowledge Representation Ontologies*: they model the primitive representations used to formalize knowledge.
- *Domain Ontologies*: ontologies focused in a specific area.
- *Modular Ontologies*: an ontology module can be considered as a reusable component of a large or more complex ontology, which is self-contained but bears definite relationships to other ontology modules (Doran, 2006).

Ontology building tools provide automatic or semi-automatic support for the construction of ontologies. Ontology Learning methods extract ontological elements (conceptual knowledge) from a corpus of documents and build ontologies from them. Ontology learning integrates many complementary techniques, including machine learning, natural language processing, and data mining. The methods that apply ontology learning methods to texts extracting ontologies by applying natural language processing techniques. The most well-known approaches are pattern-based extraction, association rules, conceptual clustering, ontology pruning and concept learning.

2.2 Classification of Semantic Search Systems

The use of ontologies and metadata is the most important aspect of the Semantic Web. Ontologies have contributed to the emergence of semantic search engines. Among these, we include the contextual search engines based on domain ontologies, which restrict the search to a well-delimited area. We can distinguish two main categories of search engines: ontology search engines and semantic search engines (Mangold, 2007). We are especially interested in semantic search engines, which can be divided in three groups:

- *Context-based Search Engines*: the final purpose of these engines is to enhance the performance of traditional search engines (especially precision and recall), by understanding the context of documents and queries. One of their most important parts is the annotator, which is responsible for generating metadata from the crawled pages. These systems are the most practical ones; in fact, they are the next generation of current search engines.
- *Evolutionary Search Engines*: These search engines are an answer to the famous and well known problem of how to automatically gather information about a specific topic. The main distinguished behaviour of these engines consists in using external metadata. They normally use an ordinary search engine and display augmented information near the original results.

- *Semantic Association Discovery Engines*: The goal of these systems is to find various semantic relations between input terms and then rank the results based on semantic distance metrics. An upper ontology like WordNet or OpenCyc can be used to evaluate this kind of search engines.

The main problems of ontology-based IR are the reformulation of the users' queries and the difficulties to build and manage ontologies. Many ontologies are available on the Web, but it is still difficult to find an ontology for every domain associated to a user's query. However, it is obvious that building ontologies for all domains, so that they can be exploited by semantic search engines, is difficult. So, using previous experiences of users can help to accomplish this mission.

2.3 Semantic Search based on CBR

Case-based Reasoning (CBR) is a methodology for problem solving by reusing previous experience (Watson, 1999) and collecting new experiences, since every new problem case, once solved, becomes a new case that may be stored and reused. The tf-idf (term frequency–inverse document frequency) weighting scheme is used in the vector space model together with cosine similarity to determine the similarity between two textual cases (Rosina, 2005).

Information retrieval based on CBR allows the management of user experiences and also some scalability to adapt and customize the system to search. It provides a strong collaboration that could simulate the learning during the search. Moreover, the incremental acquisition of knowledge on search instances can make the system scalable, thanks to the long-term learning. CBR presents the advantage of its easy exploitation, in comparison with other learning methods. However, it presents the difficulties of case modelling and representation. Many information retrieval approaches attempt to improve the recovery process (indexing texts, formulation) by using ontologies.

The combination of ontologies (as semantic background) and CBR mechanism (to enrich the ontology from search feedback) can improve the performance of Semantic Web search. However, it faces the difficulties of knowledge modelling from textual data and the representation of search cases.

3 CONTENT-BASED SEARCH ENGINE BASED ON CBR AND ONTOLOGIES

To overcome the problem of information retrieval on the Web, we have developed a system based on ontologies and case-based reasoning. In this section, we present an implementation of a generic approach (Ben Mustapha, 2010) which allows any search engine to develop its semantic layer by building ontologies and indexing a document base.
Our contribution integrates case based reasoning with ontologies to ameliorate the semantic search system and add an automatic ontology learning component.

3.1 Repository of Modular Ontologies

We have proposed in previous works the extensive use of domain ontologies (Ben Mustapha et al., 2009a). Our choice is motivated by the fact that domain ontologies are restricted to the concepts related to a specific knowledge field. This has the advantage of limiting the ambiguity of terms defined in the ontology, facilitating their detection in documents. We can represent a relationship between domain ontologies and modular ontologies as shown in Figure 1. For example, we can define two domains such as "Tourism" and "Computer Science", which can contain shared ontology modules like "Conference".

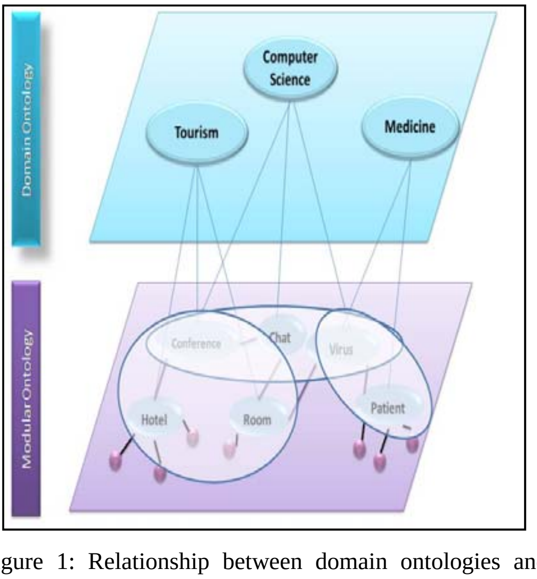


Figure 1: Relationship between domain ontologies and modular ontologies.

3.2 Architecture

In this section, we describe the architecture of the proposed system, which offers some functionalities to search Web documents for the users. It has four modules (see Figure 2). The first module is responsible for the treatment and reformulation of the users' queries, and the second one manages the case base, using ontologies. After these two processes, the system filters and classifies the documents selected by the users. Finally, the system performs an ontology enrichment process based on the analysis of the text of the selected Web documents.

The first step consists on processing the user's search query in order to match it with a topic covered in the ontology (module). The choice of the module from the ontology can help to retrieve the exact meaning of the search using WordNet.

Then, the user query is processed by the query treatment module. In the case that the search topic is new, the system provides an initial list of Web documents. If a similar case (search) is found in the case base, the system displays a list of already retrieved documents containing relevant results and adds the list from the search engine. Then, these documents are classified by means of the filtering process. Finally, the documents that are more similar to the query are displayed to the user. After this, the reformulation query module uses the case base (more precisely, the semantic signature) to provide a new query. The reformulated query allows launching a new search instance to refine the results. The final task of the system is the enrichment of the ontology from the Web documents assessed as relevant, using text-based Ontology Learning techniques.

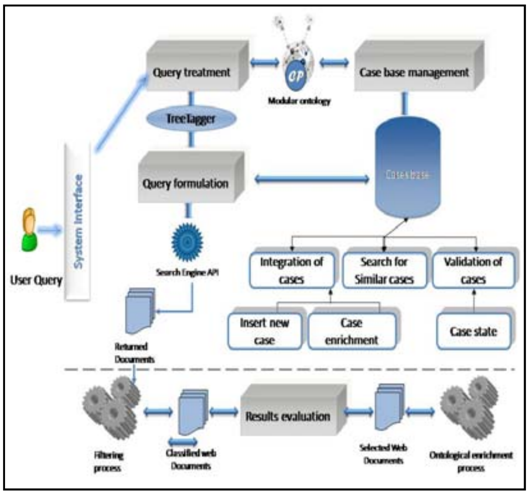


Figure 2: Global architecture of the system.

3.3 User Query Reformulation

The *Query Reformulation* step has the aim of extending the original query with additional information to obtain more accurate results. This process can help to solve the difficulties faced by the users to express their queries. Our approach is based on the enrichment of the query using the information available from past cases. The submitted query is also extended on the basis of the concepts and relations of the domain ontology. More precisely, two tasks are performed to reformulate the query:

- *Morphological Analysis*: It allows the recognition of different forms of words from a lexicon (dictionary or thesaurus). Stemming allows the transformation of conjugate or flexed forms to their canonical form or lemma. In our approach the stemming of the query is performed by *treetagger*.
- *Adding Terms from Semantic Signature*: Each query is referenced by the semantic signature of a case, whose terms can be used to enrich the query. They represent the concepts used more often for the search topic and the given query. Each term is described by a weight calculated from the relevant documents of the case.

3.4 Ontology-based CBR

The case based reasoning (CBR) is the act of solving a problem by using past experiences, stored in the case base.

Case Modelling: Typically, a case contains at least two parts: a description of a situation representing a "problem", and a "solution" used to remedy this situation. We have modelled a case with the structure shown in Figure 3.

Problem Modelling: A problem is composed of five parts (Ben Mustapha et al., 2009b). The first one represents the *search goal*. We can distinguish two types of search goals (Rose and Levinson, 2004). *The navigational goal* is a desire by the user to be taken to the home page of an institution or organization. To be considered navigational, the query must have a single authoritative Web site that the user already has in mind. For this reason, most queries consisting of names of companies, universities, or well-known organizations are considered navigational. Also for this reason, most queries for people – including celebrities – are not. *The informational goal* includes the desire to locate something in the real world, or simply to get a list of suggestions for further research. Informational queries are focused on the user goal of obtaining information about the query topic. This category includes goals for answering questions that the user has in mind.

The second part includes the *domain search*. It is referred by a topic concept from the domain ontology which aims to define the topic of the search. We have also modelled a *pivotal concept* for the chosen ontological module and a *set of queries* classified by a record. The last part is the *semantic signature* of the pivotal concept, which is a list of terms that appear frequently with the chosen concept.

Solution Modelling: The case solution contains a *module vector*, used for filtering documents of the current search and the similar cases. This vector is represented as a n-tuple (W_1C_1, W_2C_2, W_3C_3.., W_nC_n), where each tuple (W_i,C_i) represents a concept of the ontology module and its associated weight. The case solution also holds a *domain vector*, similar to the module vector, which contains the domain concepts and their weights. Finally, we have modelled a *results* part to collect the search results which are selected by the users. These results could be Web documents, ontological classes or text fragments, depending on the search goal.

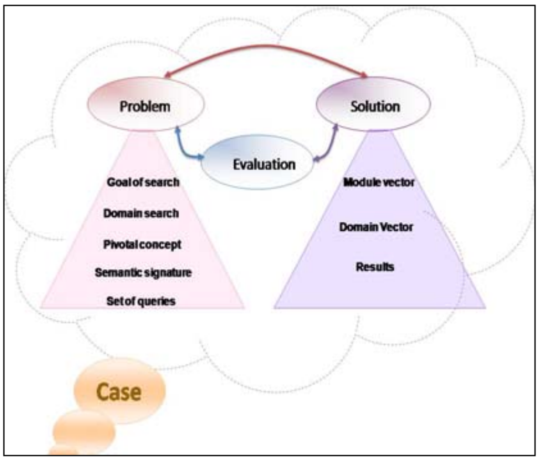


Figure 3: Case structure.

Procedures Applied to a Case. We present in the following paragraphs the different procedures that allow the enrichment of the cases (see Figure 4).

Search for Similar Cases: The submission of a query by a user is treated as a case search. The system starts by searching in the case base. Two requests are called *similar* if they belong to the same search area for the same search goal, they are both connected to the same ontological concept, and they include a similar set of terms (i.e., the terms are identical to the keywords of the original query, or synonyms of them). If a similar case exists in the database, the system executes two processes. The first is to collect all relevant documents in the case base related to similar queries. The second is to collect all documents from the Web related to the user query after treating and enriching the query with the semantic signature. After this document collecting phase, a classification process is triggered to classify all the selected documents and display them to the user. Then, these documents are evaluated by the user and an ontology learning tool analyzes them in order to improve the ontological module with new concepts and new relationships with other modules. Finally, the system updates the case to enrich it.

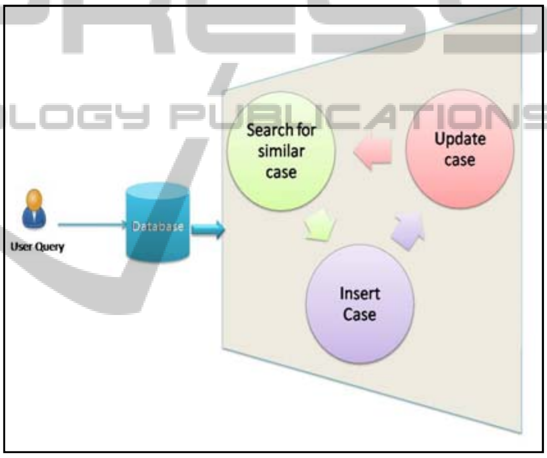


Figure 4: Procedures applied to a case.

Cases Update: Once an existing case in the base case is selected, it is updated in two steps. The first one is the addition of new relevant documents to the case. After processing the user's query, a new class of terms is added to the semantic signature. The next step is to update the module vector and the domain vector, by recalculating the weights of the terms of the semantic signature related to the case. Finally, the system saves the new adapted case into the case base.

Insertion of New Cases: This step involves the insertion of the case problem, which includes the addition of the search goal, the user-selected search field, the pivotal concept and the semantic signature (i.e., a list of terms obtained through the extraction of keywords from the user query).

After inserting the case problem, the system returns a set of URLs from the search engine. The user mission is to mark those that he/she considers relevant. The addition of the module vector and the domain vector is made as soon as the user selection is finished.

Similarity Measures: The next step is crucial for the integration and update of the module and domain vectors.

Weighting Concepts by CF-ICF Method: The CF-ICF measure is a good approximation to determine the importance of a concept in a document, particularly in a uniform size corpus. Here are some formulas to calculate CF-ICF.

CF: (Concept Frequency) is a value proportional to the occurrence and the frequency of a concept in the ontological module and so in the semantic signature.

ICF: (Inverse Concept Frequency)

$$ICF_i = Log\left(\frac{N}{n_i}\right) \quad (1)$$

N: total number of concepts in the ontology

n_i: number of documents that contain the concept c_i

The formula below (to obtain W_i) is the combination of CF-ICF:

$$CF_i = \frac{f_{ij}}{\sum_{j=1}^{N} {f_{ij}}^2} \quad (2)$$

With $$W_i = \left(1 + LogCF_i\right) * ICF_i$$

Where f_{ij} is a normalized function: to reduce the differences between the values associated with the concepts of the document. It is defined by the following formula:

$$f_{ij} = 0,5 + 0,5 \times \frac{cf_{ij}}{\max_{c_i \in D_j}(cf_{ij})} \quad (3)$$

With cf_{ij} is the number of occurrences of the concept c_i in document D_j.

Calculation of Module Vector: This involves the calculation of the weight of each term using the vector model, from the base of relevant documents related to a specific domain stored in the case base.

Calculation of Domain Vector: This metric calculates the weights of all concepts of the domain ontology using the vector model, also from the basic documents stored in the case base.

It can be noticed that CBR is used for many tasks: reformulation of new queries on the basis of previous ones, proposal of recommendations that are represented by similar queries and their results, document classification and enrichment of ontological modules.

3.5 Filtering and Classification of Web Documents

The vector model of Salton (Salton, G. and al., 1983) is used to retrieve the most relevant documents. We replace the terms by concepts to make it more suitable for our application. More precisely, we filter the domain vector to eliminate documents outside the area, which do not belong to the same module; therefore, this vector can reduce the search noise.

In this vector model each document is represented by a vector which has the following form: D_i= $(d_{1i},d_{2i},...,d_{Ni})$. We can define d_{ii} as the weight of the concept c_i in the document D_i, being N the number of concepts that are in the semantic signature. The query is represented by the vector Q= $(q_1,q_2,...,q_N)$, where q_i is the weight of the concept c_i in the semantic signature. The measure of similarity between a document and a query is calculated with the cosine formula:

$$Sim(D_j,Q) = \frac{\sum_{i=1}^{N} d_{ij} q_i}{\sqrt{\sum_{i=1}^{N} {d_{ji}}^2 \bullet \sum_{i=1}^{N} {q_i}^2}} \quad (4)$$

3.6 Ontology Enrichment from Web Documents

The final task of this system is the enrichment of the ontology fragments (associated to an old or new case) from the Web documents making up the solution of this case. Each document will be the input of the ontology learning phase of the process proposed in a previous work (Baazaoui Zghal and al., 2007) (Baazaoui-Zghal and al., 2008). Text mining techniques (syntactic patterns (Hearst, 1992) and verb based patterns) are used to discover new concepts and new relations between the concepts of the ontology fragment and the topic signature (Aguirre and al., 2004).

It is possible to extract new terms which have not a stable relation with the ontology fragment. In this case, these terms are added to the topic signature of the case (instead of being added to the ontology) in order to be used in a next iteration.

4 ILLUSTRATING SCENARIO AND USE CASE

4.1 Illustration

This section illustrates a simple scenario using the search system. We present an example in which the user wants to search for “a notebook HP”. As it is shown in the last section, our system is composed of four modules. So, the first step consists in the treatment of the query. In figure 5 we present the first iteration, including the primitive query built by the system.

The system begins by reformulating the user’s query to refine the search. On the one hand, the system searches for similar cases to send their documents (if they exist) to the user, and inserts the module and domain vectors in the case base. Then, the search engine obtains a list of Web documents. This search tries to construct a primitive ontology. On the other hand, after selecting the relevant documents, the system uses text mining techniques to enrich the primitive ontology, by extracting concepts that are relevant to the ontological module from the selected Web documents; in this case the ontological module is the “notebook”.

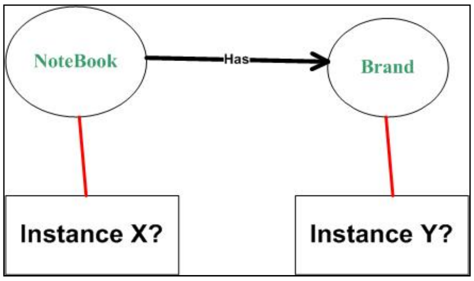


Figure 5: An example: HP Notebook search.

After the completion of this scenario, new concepts and instances are added to the module and the query of the user is reformulated and enriched with related terms and concepts as represented in Figure 6.

4.2 Use Cases Demonstration

In this section, we show two use cases representing the semantic relation between two queries *Q1* and *Q2* that appear as two independent requests submitted by two users but, according to the domain knowledge, the result of one is the answer to the other. The query Q1 is *“What is an EMS in Computer Science?”* and Q2 is *“How to make an online meeting or workshop? “*.

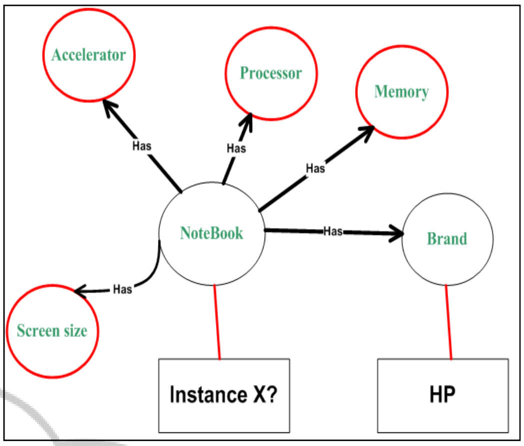


Figure 6: Module enrichment.

We suppose that the first user submits the query Q1. This first iteration can be done with any search engine, given that there is no learning at this stage. Concerning this query, we cannot find any related ontology module nor similar cases as the query keywords are not clear. A graph corresponding to the query is extracted using lexico-syntactic patterns. Then, the corresponding keywords are sent to a search engine. The user selects the relevant documents to be saved and a new case C1 is inserted in the case base (Table 1). Both the insertion of a new case and the calculation of the two vectors are done automatically. After the case insertion, the construction of the ontology module related to the concept “electronic meeting system” is done by fulfilling the following tasks:

- Extraction of sentences containing the term “EMS” from selected Web documents associated to case C1 (figure 7).
- Application of lexico-syntactic patterns and syntactic frames to discover new related concepts and relations.
- Validation of the ontology elements discovered for the set of Web documents.

The resulting ontology module is illustrated by Figure 8.

A second user submits the next query Q2. This second iteration is fulfilled by learning from similar queries. The difference between the two queries is that the first user knows exactly the concept he is searching for (“EMS”) while the second user only knows the role of this concept (without knowing the fact that an EMS is a tool to make online meetings or workshops). The use of a traditional search engine to answer the query Q2 will lead to the difficulty of finding that EMSs are used to make online meetings or workshops, as “EMS” does not

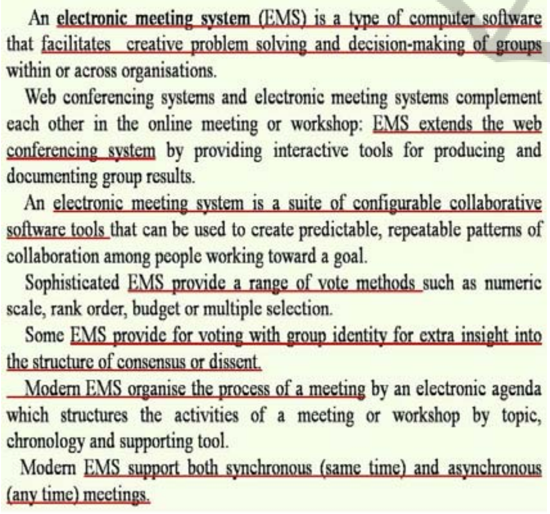
Table1: Case representation of C1.

Case Index	http://www.SemSearch.com/Computer_Science /ontology#EMS.owl

Problem	GS= navigational
	D=http://www.SemSearch.com/Computer_Science
	PC: EMS
	TopicSign= meeting, conference, online, collaborative….
	Set_Q= (EMS, (EMS +electronic meeting system))

Solution	Module Vector Domain Vector Set_response= http://en.wikipedia.org/wiki/Electronic_meeting_system

appear as a term in the second query. But, according to the proposed approach, the ontological index of cases can provide the second user with a similar case existing in the case base.



An electronic meeting system (EMS) is a type of computer software that facilitates creative problem solving and decision-making of groups within or across organisations.

Web conferencing systems and electronic meeting systems complement each other in the online meeting or workshop: EMS extends the web conferencing system by providing interactive tools for producing and documenting group results.

An electronic meeting system is a suite of configurable collaborative software tools that can be used to create predictable, repeatable patterns of collaboration among people working toward a goal.

Sophisticated EMS provide a range of vote methods such as numeric scale, rank order, budget or multiple selection.

Some EMS provide for voting with group identity for extra insight into the structure of consensus or dissent.

Modern EMS organise the process of a meeting by an electronic agenda which structures the activities of a meeting or workshop by topic, chronology and supporting tool.

Modern EMS support both synchronous (same time) and asynchronous (any time) meetings.

Figure 7: Extracted sentences containing the term "EMS" from selected Web documents associated to case C1.

By applying this approach, the system has noticed that there is a common result between the two queries. However, by using the search engine, we wouldn't be able to find the same result. In fact, the retrieval of cases similar to query Q2 provides the most ranked recommended cases (C1 and many other cases, as C2 shown in Table 2).

For this second user, we can provide a summary of what is an online meeting, using the ontology module (concepts in the neighborhood of online meeting), and give a set of recommended documents. In this context, we supposed that a previous user submitted queries related to meetings. The explication of this common result is that the ontology module indexing C1 is strongly related to the ontology module indexing C2 (figure 8).

The relation between the two ontology modules was discovered in the step of ontology enrichment of the case C1. The two ontology modules were automatically created in the first iteration by applying text mining techniques on the documents of the C1 case.

Table 2: Case representation of C2.

Case Index	http://www.SemSearch.com/Computer_Science /ontology#online_meeting.owl

Problem	GS= navigational
	D=http://www.SemSearch.com/Computer_Science
	PC: EMS
	TopicSign= meeting, online, tools,...
	Set_Q= (make online meeting, workshop making tools, interactive meeting tools,....)

Solution	Module Vector Domain Vector Set_response= http://www.online-tech-tips.com /cool-websites/free-online-meeting-software/

So, EMS is one of the available online meeting tools. There are other answers such as "Web Conferencing Roundup" (C2). The selection of one of the proposed cases narrows the process of search. This implies the use of the vector module. In this example, we suppose that the second user selects the case C1 among the recommended cases. Then, a set of queries are submitted to import other Web documents whose similarity with the Module Vector is important. Then, we apply the document filtering step by using the domain vector. We remark that the majority of removed Web documents include the term EMS, but they have not the same meaning of "electronic meeting system". In fact, there are many other senses of the expression "EMS", such as "Express Mail Service" and "European Mathematical Society".

5 SYSTEM EVALUATION

We have just shown how CBR and ontologies are used in the search. To evaluate our system we computed the precision in the first part and the influence of CBR on ontology enrichment in the second part.

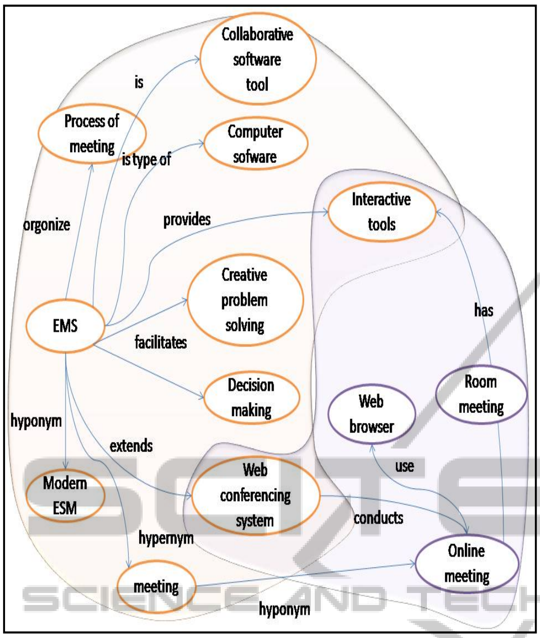


Figure 8: Relation between two ontology modules indexing two similar cases.

5.1 Search Results Evaluation

We measured the precision of our tool for finding information online. Several scenarios have been tested. The *first scenario* represents the initial search, which is a keyword search on Google. The *second scenario* represents the situation where there are similar cases in the database. The search is based on the vector model to filter the results. The *third scenario* represents the search for information based on the learning of cases, the query reformulation and the use of the vector model for classification.

The precision measures the rate of relevant documents that are manually assessed by a human domain expert. Its values are calculated for several queries. In Figure 9 we observe that the results show a significant improvement of the relevance of the returned information gradually as the concepts are used in experiments. This demonstrates that the implemented ideas contribute significantly to improve the relevance of search results.

5.2 Ontology Enrichment Evaluation

In Figure 10 we can observe that the relevance of the obtained results shows a marked improvement in the considered scenarios. For the first scenario, the added concepts are from WordNet so this is the contribution of the disambiguation. The added concepts in the second scenario are based on similar cases in the case base, and so that is the contribution of the CBR system. Finally, for the last scenario, the added concepts are from the text, and that is the contribution of enrichment from text.

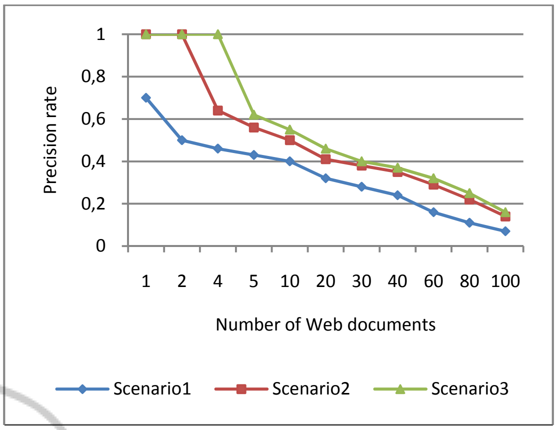


Figure 9: An average precision for a system search.

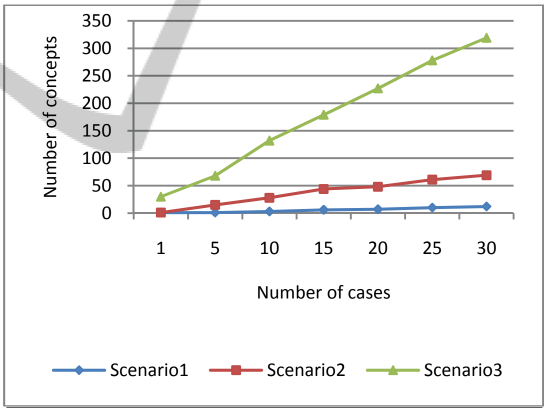


Figure 10: Average number of added concepts for each scenario.

In the first scenario, we use only WordNet. There are many relevant concepts in relation to the total of concepts, but the number of concepts is very low to describe the user’s search (see Figure 11).

In the second scenario we can extract concepts from similar cases. We can observe the increase of the number of relevant concepts in comparison with the first scenario, despite the decrease of the pertinent concepts average in relation to the total of concepts. This can highlight the contribution of case-based -reasoning in our work (see Figure 12).

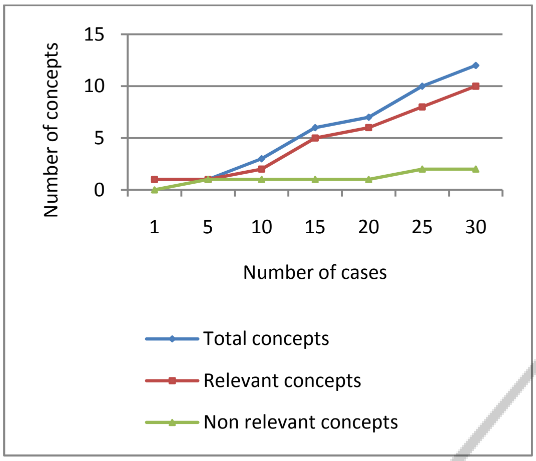


Figure 11: Concepts relevance for the first scenario.

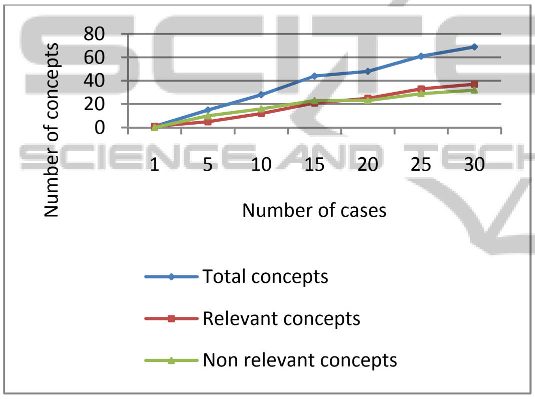


Figure 12: Concepts relevance for the second scenario.

The third scenario represents the concepts extracted from text (from relevant Web documents). We can notice the great number of concepts retrieved by the proposed system. This number of concepts, especially the relevant ones, demonstrates the contribution of ontology enrichment in our work (see Figure 13).

Our system highlights the contribution of case-based-reasoning and the ontology enrichment. However, it presents some limitations. In fact, returned documents for the user's query and ontology enrichment are not synchronised, these two processes can not unfold at the same time. The ontology enrichment process takes time because of the volume of Web documents. That is why we thought to use other ontology techniques based on Web metrics and data returned by search engine to paralellize the search and the enrichment processes in our future work.

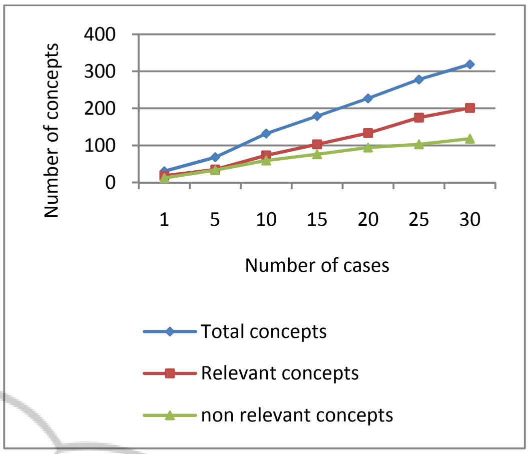


Figure 13: Concepts relevance for the third scenario.

6 CONCLUSIONS AND FUTURE WORK

The challenge addressed by this paper is to find a solution to improve the contextualization of requests based on past users' queries and the enrichment of ontologies from the query context (documents selected by the user). In this paper, we have presented related work that uses CBR technology in IR systems and Ontology Learning for IR. In fact, since intelligent retrieval is one of the main applications of Case-based reasoning paradigm (CBR), the semantic formalization in CBR systems is becoming an increasing research area. In CBR systems, semantics are the main source of reasoning, similarity calculation and case adaptation. Our underlying hypothesis is that case-based reasoning supported by ontologies is a promising approach for achieving semantics aware search and ontology enrichment. In this work, we discussed an evolutive semantic search based on case-based reasoning, by which the traditional information retrieval, ontology and CBR can be integrated. Our recommender approach uses Case-Based Reasoning (CBR), with semantic Web language markup (ontology) for case indexing.

We present in this paper a system that supports the proposed approach (Ben Mustapha et al., 2010) of semantic search based on ontologies and CBR. The main contribution of this work is the ontology fragment building and enrichment within a semantic search engine and linking user queries with ontology fragments using the case base reasoning.

The ontology enrichment is based on our previous work that proposes to analyse the whole Web page to learn new concepts and relations. So, it is obvious that this approach is time consuming to set it as an online task. So, we intend in the future work to study especially ontology learning techniques based on Web metrics (Sánchez and Moreno, 2008a, 2008b).

Besides, since modularization is a very important aspect in order to facilitate the enrichment, maintenance and the reuse of the ontologies, the main motivation of our future work is to adopt ontology learning techniques based on the Web to build modular ontologies by the composition of extracted ontology modules.

ACKNOWLEDGEMENTS

This work has been supported by the Spanish-Tunisian AECID project A/023175/09, "A framework for the integration of Ontology Learning and Semantic Search".

REFERENCES

Agirre, E., and Lacalle, O., 2004. Publicly available topic signatures for all wordnet nominal senses. *In the 4rd International Conference on Language Resources and Evaluation (LREC)* (Lisbon, Portugal, 2004).

Baazaoui Zghal, H., Aufaure, MA., Ben Mustapha, N., 2007 Extraction of Ontologies from Web Pages: conceptual modeling and tourism Journal of internet Technologies.

Baazaoui-Zghal, H. Aufaure, MA., Ben Mustapha, N. A, 2008. Model-Driven approach of ontological components for on-line semantic web information retrieval", *Journal on Web Engineering, Special Issue on Engineering the Semantic Web*, Rinton Press, vol. 6, n°4, pp 309-336.

Ben Mustapha N., Baazaoui Zghal H., Aufaure MA. and Ben Ghezala H., 2010. Enhancing Semantic Search using Case-Based Modular Ontology, In *25th Symposium On Applied Computing (SAC) , track on "The Semantic Web and Applications" (SWA),* 22 – 26, Sierre, Switzerland.

Ben Mustapha N., Baazaoui Zghal, H., Aufaure, MA. and Ben Ghezala, H., 2009a. Combining Semantic Search and Ontology Learning for Incremental Web Ontology Engineering, *International Workshop on Web Information Systems Modeling* (WISM 2009), Amsterdam.

Ben Mustapha, N., Baazaoui Zghal, H., Aufaure, MA. and Ben Ghezala, H., 2009b. Ontology learning from Web : survey and framework based on semantic search, *Second International conference on Web and Information Technologies*, ICWIT2009, Kerkena, Tunisia.

Berners-Lee, T., Hendler, J. and Lassila, O., 2001. The Semantic Web. *Scientific American*, May, pp. 29-37.

Bradley N., 2001. *The XML Companion*, Addison-Wesley Professional Publisher.

Charlet, J., Szulman, S., Pierra, G., Nadah, N., Teguiak, H. V., Aussenac-Gilles, N., et al. (2008). DAFOE: a multimodel and multimethod platform for building domain ontologies. in D. Benslimane, (Ed.), 2èmes Journées Francophones sur les Ontologies, JFO'08, ACM.

Decker S., M. Erdmann, D. Fensel, I. Horrocks, M. Klein,F.van Harmelen, 2000. Oil in a nutshell, In Proceedings of the 12th European Knowledge Acquisition Workshop (EKAW'00).

Doran P.. Ontology Reuse via Ontology Modularisation. In Proceedings of KnowledgeWeb PhD Symposium 2006 (KWEPSY2006), 17th June 2006. Budva, Montenegro.

Guarino, N., 1998. Formal Ontology in Information Systems. In *1st International Conference on Formal Ontology in Information Systems* (FOIS'98). Trento, Italy, pp. 3-15.

Karp R., V. Chaudhri, J. Thomer, 2002. *An xml-based ontology exchange language.*

Hearst, M. A., 1992. Automatic acquisition of hyponyms from large text corpora. In *14th International Conference on Computational Linguistics* (COLING-92). Nantes, France, pp. 539–545.

Lassila O. R. R. Swick, 1999. *Resource description framework (rdf) model and syntax specification w3c recommendation* 22.

Mangold, C., 2007. A survey and classification of semantic search approaches, *In Journal of Metadata, Semantics and Ontology*, Vol. 2, No. 1, pp.23–34.

McGuinness D. L., F. van Harmelen, 2004. *OWL Web Ontology Language Overview*.

Rose, D. E. and Levinson, D., 2004. Understanding User Goals in Web Search. In *International World Wide Web Conference*, New York, USA, pp. 13-19.

Rosina O.Weber, Kevin D. Ashley and Stefanie Brüninghaus, 2005. *Textual case-based reasoning*. The Knowledge Engineering Review, Vol. 20:3, 255–260.

Salton G, Fox E.A, Rocchio J, Extended Boolean information retrieval system.CACM 26(11), pp. 1022 1036, 1983.

Sánchez, D. and Moreno, A., 2008a. Learning non-taxonomic relationships from web documents for domain ontology construction, *Data & knowledge Engineering*, Vol. 63, No. 3, pp. 600-623.

Sánchez, D. and Moreno, A., 2008b. Pattern-based automatic taxonomy learning from the Web, *AI Communications*, Vol. 21, No. 1, pp. 27-48.

Watson, I., 1999. *Case-based reasoning is a methodology not a technology,* Knowledge-Based Systems, Vol. 12, No. 5, pp. 303-308.