Design and implementation of information retrieval

system based ontology

Lachtar Nadia
Preparatory School for sciences and techniques
Annaba, Algeria
nadia_ishak2002@yahoo.fr

***Abstract*—Nowadays, the resources available on the web increases significantly. It then has a large volume of information, but without mastery of content. In this immense data warehouse research of current information retrieval systems do not allow users to obtain results to their requests that meet exactly their needs. This is due in large part to indexing techniques (key words, thesaurus). The result is that the user of the web wasting much of his time to examine a large number of Web page by searching for what he needs, because the Web does not provide service in this direction. The Semantic Web is the solution; this new vision of the web is to make web resources not only understandable by humans but also by machines. To improve the relevance of information retrieval, we propose in this paper an approach based on the use of domain ontology for indexing a collection of documents and the use of semantic links between documents in the collection to allow the inference of all relevant documents. The work involves the implementation of a system based on the use of OWL ontology for research pedagogical documents. In this case, the descriptors are not directly chosen in the documents but in the ontology and are indexed by concepts that reflect their meaning rather than words are often ambiguous. To perform a search based on meaning, documents and their descriptors are stored in OWL ontologies describing the documentary features of a document. The objective is to design two types of OWL ontologies: document ontology reserved for storage of all pedagogical documents and domain ontology reserved for well-structured of documents stored in the level of the document ontology and each document is indexed by its keywords and their synonyms.**

Keywords-component; Pedagogical document; Information retrieval; ontology; sematic web; indexation

I. INTRODUCTION

The information retrieval (IR) is an ancient discipline; it dates back to the 50s. His problematic can be seen as the satisfaction of a need for information of user, which is expressed by a query on a collection of documents called the corpus or collection [14, 12] .The information retrieval systems (IRS) allows you to automate the task of IR. The evaluation of such systems appears to be a necessity. This evaluation is based on the concept of relevance. So, to improve the relevance of IR in IRS, several studies have been made at various levels. Thus, there have been proposed several IR models:

The Boolean model, Boolean queries are composed of words and Boolean operators (AND, OR, NOT). Documentalists have more control over this type of query that is often difficult to formulate for the uninitiated user. This type of query is the most used for access to specialized databases (Pascal), is also available for many search engines on the web such as Google and Yahoo from advanced search interfaces.

The vector model [11], in this model, documents and queries are represented as vectors in the space of words from indexing. The documents are then ordered from their similarity to the query. Several measures (scalar product, Measurement Dice, Jaccard measure, ...) are used to calculate the similarity between the two calculations corresponding to the distance between the two vectors.

The probabilistic model is based on the probability of relevance of a document knowing the query.

The connectionist model, LSI ... Some work has focused on the representation of information needs, the length of the query [15] or the reformulation of the query [8].

Other studies have looked at the indexing process and indexing languages. Several techniques have been proposed: keyword lists have the ambiguity problem due to polysemy, thesauruses unlike semantic networks are not limited to the defining relationship of lexical relations between nodes [1, 9, 10, 16], such as in the medical field Mesh and WordNet [6] for the English language.

The approximate, poor and partial representation of semantic content of documents using indexing techniques (keywords or thesaurus) presents problems of indexing words:

- When different words refer to same sense, the indexer generally favors the word that appears in the document. Assuming that the user uses in his request another word, so, it does not access the documents related to this notion. The synonymy produces documentary silence.
- When the same word refers to different meanings, use of the word by the user results in a response containing documents on these notions, even if it is only interested in one of these concepts. The homonymy produces documentary noise.

This led to the use of knowledge representation formalisms of more accurate and rich expression skills. Among these formalisms, ontologies are used to characterize a domain by a set of concepts and relationships between concepts.

Ontologies in the IRS are used to define other types of inquiries that are based on Semantic Web languages.

To overcome these problems with indexing keywords, the

978-1-4799-3824-7/14/$31.00 ©2014 IEEE

solution is to index documents by concepts, while maintaining links: concept / term (s).

- This solution is in line with the practices of documentalists who already operate thesaurus structured sets of terms for indexing documents.
- The supply of a conceptual indexing comes, on one hand, the exploitation of links: concept / term (s), and secondly, the exploitation of semantic relations structuring the ontology. The user continues on the other hand to use words in its queries.

Our work is situated at the intersection of the semantic web research information using ontologies in this area. It consists in the implementation of a system based on the use of OWL ontology for research pedagogical documents indexed through keywords and documents can be of different types like PDF, HTML, DOC,

The real force of this approach resides in the manipulation of OWL ontology which allows:

- The possibility of inference and reasoning.
- Check the consistency of ontologies through powerful tools for verification.
- Interoperability and integration between different ontologies through existing import statement in OWL.

The application consists of two motors:

- Engine Indexing: aims to create a link between the document instances and their keyword.
- Search engine: aims to filter the user query to return relevant results in a suitable time.

The paper is organized as follows: section 2 introduces the concepts of ontology; Section 3 describes the design of the system. Section 4 talks about the prototype finally in Section 5, we finish with a conclusion.

II. NOTION OF ONTOLOGY

The term "ontology" comes from the field of philosophy where it means science or theory of being. In the field of artificial intelligence, the meaning is different. Neches and others [7] were the first to propose a definition: "Ontology defines the terms and the basic relationships of the vocabulary of a domain and the rules that specify how to combine the terms and relations to able to extend the vocabulary".

The definition of Gruber [3] is the most cited in the literature: "An ontology is an explicit specification of a conceptualization." It was slightly modified by Borst [2]: "an explicit and formal specification of a shared conceptualization." It is thus explained [13]. Explicitly means that the type of concepts and constraints on their use are explicitly defined, formal refers to the fact that the specification must be readable by a machine, is shared refers to the notion that an ontology captures consensual knowledge, which is not unique to an individual, but validated by a group conceptualization refers to an abstract model of some phenomenon in the world based on identifying relevant concepts of this phenomenon.

A. Ontology in information retrieval

In IR, ontology can be used at various levels [5]. First, it helps to refine a system based on a traditional indexing process by increasing the chances of making a request from the terms or descriptors that best reflects the need for information. This method has several advantages:

- Reduce the silence over all documents returned by relying on terms not explicitly present in the query. For this, queries are extended from terms in the ontology and related to concepts present in the request;
- Reduce noise in all documents returned. The idea is to avoid returning documents containing the query terms used but in a different sense. The ontology defines a unique and unambiguous each concept, the goal is to exploit these properties correctly using synonymy relations and disambiguation to preserve in the query than terms expressing clearly the need information.

Ontology can be used for indexing documents. In this case, descriptors are not selected directly in the documents but within the ontology. The texts are then indexed by concepts that reflect their meaning rather than words are often ambiguous. It is suitable in this case to use an ontology reflecting areas of knowledge addressed in the document collection.

While indexing using ontology has some disadvantages such as: the definition of ontology (concepts and relationships between concepts) is hard work and time consuming, and sometimes like its inadequacy (which can be calculated using a lexical analysis and conceptual analysis [4] to the indexed corpus, the benefits of such indexing are multiple:

- Allow a faithful representation of the semantic content of the documents;
- Facilitate the RI in heterogeneous collections by indexing any type of document from the same concepts;
- Enable intelligent search, and that exploiting the semantic relationships between indexing concepts, by inference mechanisms (which allow reasoning elaborated).

We will be interested in the issue of IR by using ontology for researching pedagogical documents indexed through keywords and these documents can be of different types like PDF, HTML, DOC.

III. DESIGN OF THE SYSTEM

Our initially job is to create an ontology contains several concepts of pedagogical documents, then we do the indexation, after we create an OWL ontology navigator allowing the graphical visualization of this kind of ontology and present a new approach of a search tool based on the use of formal ontologies OWL in research and documents

indexing.

A. System 0bjectives

Besides navigation, the second objective of our work is to show how to use a formal OWL ontology in information systems research. The creating and reading of such ontology requires powerful tools, but it offers a promising solution in order to improve system performance.

This section discusses the design of our system which includes a browser allowing the graphical visualization of OWL ontology and a retrieval system consists of three modules operating two ontologies: one represents the domain of expertise and the other representing the database engine. Therefore, the analysis module is intended to extract the features documentaries of document types: PDF, Word, HTML and PPT, the indexing module aims to insert these documents as instances in the document ontology, the search module aims to refine the user query according to the ontology expertise and return the results to the user.

1) The indexation of pedagogical Document: the construction of ontologies required to test by indexing some resources, then modify the ontologies, then reindex some resources, and so on. So in this section, we explore the approach of the Semantic Web applied to indexing pedagogical documents. In this case, the descriptors are not chosen directly in documents but in the ontology. Documents are indexed by concepts that reflect their meaning rather than words are often ambiguous. And to perform a search based on the sense, we propose to store documents and their descriptors in OWL ontology describing the documentaries features a document.

2) The query expansion: ontology can help the user to formulate his query. In presenting the ontology to the user, it is possible to guide him in the selection of terms of its query. For this, queries are extended from term contained in the ontology and related to concepts present in the query.

So our approach is based on two types of OWL ontology: one reserved for storing documents to perform a semantic search in an ontology and not in a database as the case of the current search engines and the other reserved for storage of concepts of an expertise area that represent the specialty engine.

3) Design of document ontology : for the design of the document ontology, we will define the concept document and its properties that allow the storage of all pedagogical documents.

The ontology document describes the documentary features of documents that must be stored, for that she has two concepts which are document described by the following properties: has_URL, has_university, has_ date has_faculty, has_department, has_format, has_langue,, has_keyWord. And document_type described by instances "document_PDF" "document_Word", "document_HTML", "document_PPT."

TABLE I. DESIGN OF DOCUMENT ONTOLOGY

Concept	*Parent*	*Attribute*
Document	Thing	Has_URL Has_Titre Has_Auteur Has_keyword Has_Date Has_univerity Has_faculty Has_department Has_format

In the OWL model: has_keyWord is a property of type OBJECTPROPERTY possessing the range of expertise ontology, so it is used to connect the two ontology (interoperability) and this is achieved by the force of OWL. And during indexing, each document is represented as an instance of this concept.

4) Design of domain ontology: we will organize all pedagogical documents and each document belongs to one module and is indexed by its keywords

TABLE II. DESIGN OF DOMAIN ONTOLOGY

Concept	*Parent*
Module	Thing
BDD	Module
SE	Module
Reseaux	Module
Sqlserver	BDD
Adsl	Reseaux
WIFI	Reseaux

B. System Functions

Other than navigation, our system performs the following functions:

1) Analyzer: The analyzer extracts document descriptors: the document's author, title, date of creation, the summary, the address, and all the keywords belonging to the domain of expertise based on domain membership via reading the ontology.

2) Indexer: The indexer retrieves all document information and by indexing the documents belonging to our domain knowledge. It inserted them as instances of the concept document in the document ontology.

3) Search: The resarch aims: To expand the query based on the concepts of the expertise ontology. Access to

the Document ontology, select documents that satisfy the query.

4) Browser OWL: the ontology navigation is a task from our application allowing viewing the contents and navigating in OWL ontologies, with hyperlinks. The browser provides the following functions:

- Load ontologies written in OWL format and view those terms.
- Recognize the different components of the ontology through the syntax of each (concept, instance and property).
- Display information about a specific concept including its textual description, operations (intersection, union, etc. ...)., Synonyms (equivalence), hypernyms (superclass), hyponyms (subclass), properties and instances if they exists.
- Provide hyperlinks to all the terms in the description page for easy navigation in the ontology.

C. General design of system

According to the three functions cited above, the design is described as follows.

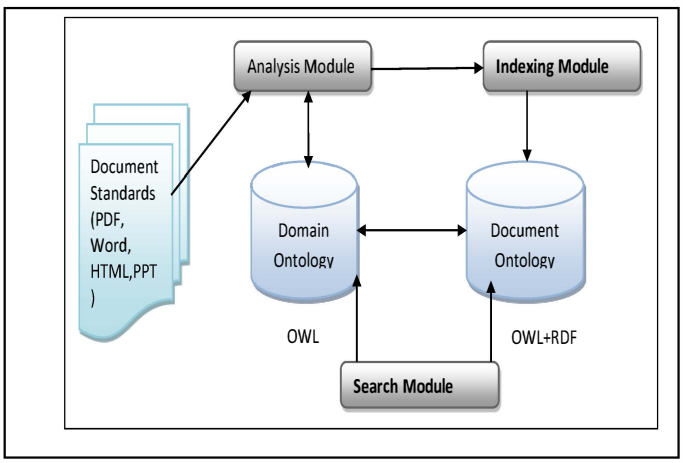

Figure 1. General Design

1) Analyzer: the analyzer role is to:

- Determine the type of document.
- Process the document to extract metadata.
- Treat the document for extracting key words from the document text.
- Send all keywords and metadata to the indexer.

We begin to check the type of document; documents (RTF, TXT ...) are refused. We only accept documents (PDF, Word, HTML and PPT). From the documents collected by the user, the analysis module extracts the following information, depending on the type of document. The metadata extracted are: the date of creation or modification, the author of the document, the document title, all the keywords and other attributes. There is a text to analyze by reading character by character. Every document containing body text is analyzed to extract a list of words. After the text analysis, we obtain a set of words; these words are treated at the end to verify the presence of the words in the field of expertise. Then we accept the words which belong to the ontology. The new list is used to index the document. The purpose of this treatment is to not index the entire list of words.

2) Indexer: The indexation role is to insert automatically an instance of a concept document at the OWL Document ontology with the values of properties (URL, University, Faculty, Author, Date, Format, Keywords, Language and department). .Indexation is done in two steps:

- Organization of document data: We retrieve from the analyzer all information with (URL, University, Faculty, Author, Date, Format, Key Words, Language, and Department).
- Insertion in the document ontology: Before the insertion of all information relating to each documents PDF, Word, HTML and PPT analyzed, it is necessary to perform a test that aims to verify the absence of his document of the ontology of instance, to avoid inserting the same document in the ontology. The test checks for the presence of the URL in the ontology. If this is not the case then the document will be indexed and inserted. But, if the URL is already presented in the database, another test is applied: if the modification date assigned to a document being indexed is greater than that of the same document in the ontology it becomes to insert the result of the new analysis, which is equivalent to an update of the ontology. This type of update is relatively effective for all documents, due to the use of a full-text indexing of documents, so that the smallest change in the document is immediately detected by the change of the modification date.

3) Search: The advantage of using ontology in the information retrieval phase is to be able to return during a search the documents that share the maximum of concepts with the query rather than the maximum of keywords. Once the request is made, we begin expanding with other concepts more specific subclass or more generic superclass or other relationships such as equivalence. For this we need the ontological reasoning in the classification to specify the location of the concept (query) in the hierarchy of concepts and instantiation to clarify the concepts corresponding to an instance and all this may be done in a way interactive with the user. The goal is to help him to properly express his query. The refinement algorithm of the query is as follow:

The query is a class in the ontology, we add to the concept:

- The list of subclasses such as query “pedagogical document” is replaced by the list of subclasses "Support courses, Series TD Series TP, Review, etc.. ".
- The list of instances.
- The equivalence list if it exists.

- If the concept is a restriction of a set of concept (union or intersection), we add the entire list of concepts to take the restriction.

The query is an instance in the ontology; we add the corresponding instance concept.

The query contains Boolean operators such as AND, OR (we used two concepts with an operator), so the query is analyzed according to the operator.

- If the operator is AND, we replace the conjunction by a term representing the intersection of terms.
- If the operator is OR, we replace the disjunction by a term representing the union of the terms.

Once the query is made, the result is a list of concepts without redundancy.

After analysis of the query, we will get a list confirmed by the user and we accede to the Document ontology (RDF part) to return the searched documents.

IV. THE PROTOTYPE

The work consists first, at the edition of the ontology followed by its exploitation in a Java application using SPARQL Jena to search documents stored in the OWL file created from the protégé editor.

We used the following tools: Protégé version 4.2.0, NetBeans IDE version 7.3.1, Jena version 2.6.2.

The first step in the implementation was the creation of ontologies with Protégé

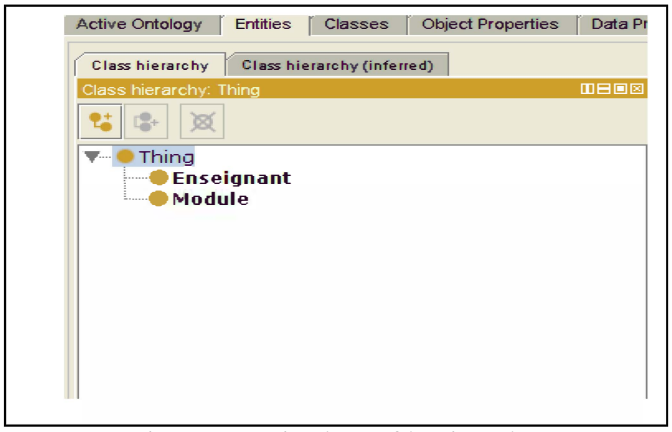

Figure 2. Creating classes of domain ontology

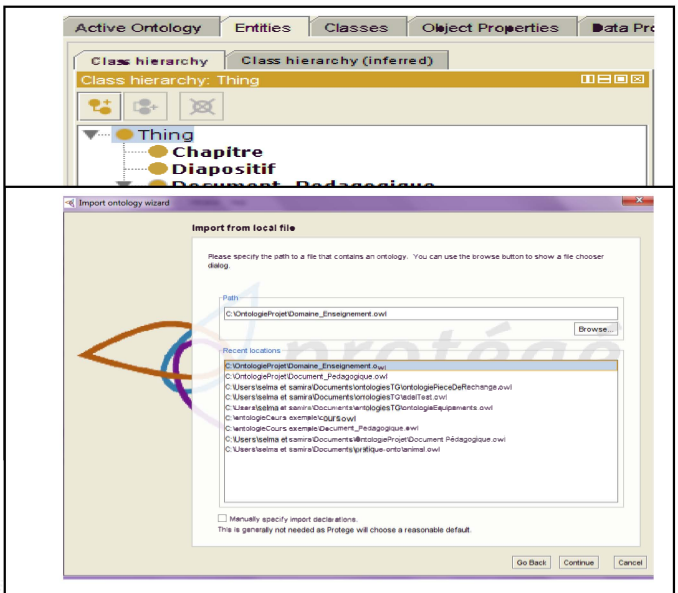

Figure 3. Creating classes of document ontology

Figure 4. Import of domain ontology in document ontology

Figure 5. Graph of two ontologies designed with the plugin OntoGraf

After generating the code OWL, we pass to its operations in a Java program; we used the Jena API that serves as a link between OWL code and a JAVA program. In the following we will present some screen captures of our application.

Once the application is launched the following interface will be displayed:

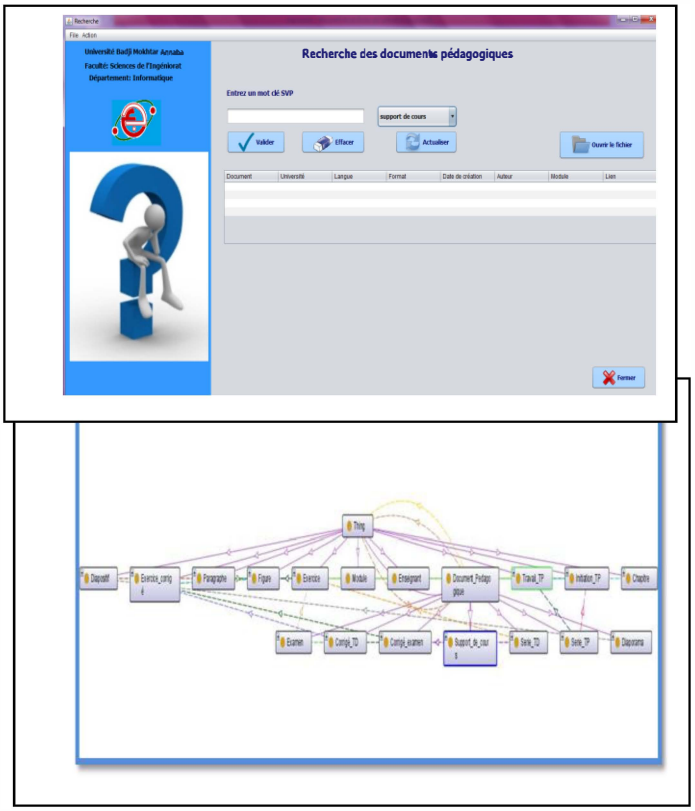

Figure 6. The graphical interface of the application

To get help, user clicks on "Action" and "Help" a window that contains all the information about the application opens.

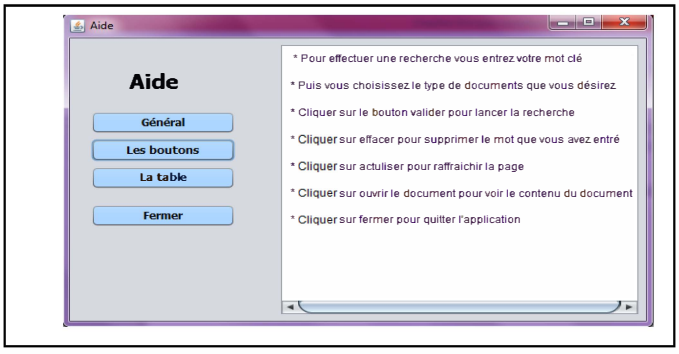

Figure 7. Help window

To perform a search the users enters a keyword in the search field and choose the desired document type:

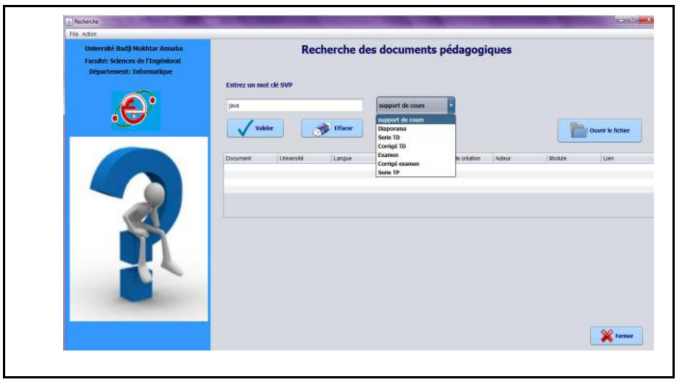

Figure 8. The search

The results will be displayed in the table with key descriptors. If the user starts the search with another keyword that is a synonym, he gets the same result.

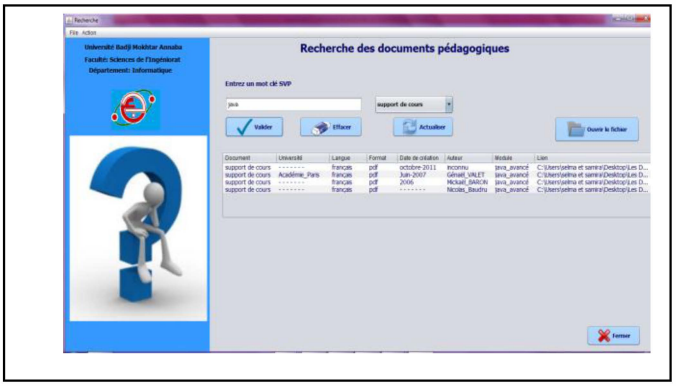

Figure 9. Search of pedagogical document: results

CONCLUSION

The use of ontologies for the specification of user requirements and search for documents corresponding has several advantages. It first makes all the reformulation of queries sent to traditional systems from adding or disambiguation of terms of the ontology. In the case where the representation of documents and query is based on ontology, it also helps to develop matching mechanisms involving the semantic similarity between concepts which compose them.

REFERENCES

[1] M. Baaziz, " Indexation conceptuelle guidée par ontologie," *These de Doctorat, 2005.*

[2] W. N. Borst,, " *Construction of engineering ontologies",* University of Tweenty, Enschede, Centre for Telematica and Information Technology, 1997

[3] T Gruber,. T "A translation approach to portable ontology specification", 1993

[4] N. Hernandez, N. Aussenac-Gilles "Ontologies pour la recherche d'information importance de la dimension terminologique , 2006

[5] C. Masolo, " Ontology driven information retrieval" 2001

[6] G.A.Miller, "WORDNET a lexical database for English", *Communications of ACM,* 1996

[7] R. Neches , R.E Fikes,, T. Finin, T. Gruber, T .Senator, W.R Swartout,, "Enabling technology for knowledge sharing ", *AI Magazine,* 1991

[8] JJ Rocchio., " Relevance feedback in information retrieval", 1971

[9] N. Hernandez, and Others, "Modèle de représentation sémantique des documents électroniques pour leur réutilisabilité dans l'apprentissage en ligne" CIDE, 2006

[10] P Saint-Dizier., E.Viegas, "Computational lexical semantic", Cambridge University Press, 1995

[11] G. Salton, "The SMART retrieval system: experiments in automatic document processing", New Jersey, Prentice-Hall series in Automatic Computation, 1971

[12] G. Salton , M.J McGill,. "Introduction to modern information retrieval ", New York, McGraw Hill Book Company, 1983

[13] R. Studer, , R. Benjamins, D. Fensel, , " Knowledge engineering: principles and methods ", *Data and Knowledge Engineering,* 1998

[14] C.J. Van Rijsbergen., "Information retrieval ", 2nd Ed, Betters Worth, London, 1979

[15] E.M Voorhees, "Variations in relevance judgements and the measurement of retrieval effectiveness", *Information Processing & Management,* Vol. 36, 1996.

[16] R. Abbes and Others , " Apport du Web et du Web de Données pour la recherche d'attributs ", CORIA 2013