Toward a new Information retrieval system based on an e-commerce ontology

Abdelhadi BAHAFID[1], Kamal EL GUEMMAT[1], El habib BEN LAHMAR[2], Mohamed TALEA[1]
[1]Université Hassan II - Casablanca, Faculté des Sciences Ben M'sik, Laboratoire de Traitement de l'Information, Cdt Driss El Harti,BP 7955 Sidi Othman, Casablanca, Maroc

[2]Université Hassan II - Casablanca, Faculté des Sciences Ben M'sik, Laboratoire de Technologie de l'Information et Modélisation, Cdt Driss El Harti,BP 7955 Sidi Othman Casablanca, Maroc

Abstract—E-Commerce is a direct extension of digitalization of our economy; it begins to have an important place in the internet's user's lives. The main problems of e-Commerce are the extraction of relevant informations about products, how to provide users with products that meet their needs, hence the need to have relevant information retrieval system ie with high accuracy and the lack of works in this field. This paper presents an information retrieval system based on an e-commerce ontology representing the different properties of the products on the internet and which will be used to establish the model of the index. At the end of this paper we have applied our product information retrieval system on a famous e-commerce web site, the results show that our system will help users to acquire the desired information on products that meet their needs.

Index Terms—E-commerce, information retrieval, indexing, ontology, web semantic

1 INTRODUCTION

Buying online has become a new trend, now the consumer doesn't need to travel to purchase products, nowadays many online shop have emerged, they offer a range of products and their characteristics.

It makes life easier for users of the internet because it gives user the ability to search and purchase the products they need, that's why these shops or sites shall have a retrieval information system to respond to users requests to find the products that correspond to their needs.

The field of e-commerce confronts several problems among which we quote: the acquisition, storage, retrieval and filtering information… etc. Research and information filtering are two primordial issues for both e-commerce sites and users that allow them to retrieve the product informations they need and be able to choose the products that best meet these needs especially users seek lower cost products using familiar languages and terminologies. However, the rich and various descriptions that sellers use to describe their products increase the difficulty of locating the products and services accurately and effectively. That different vendors can adopt different ways to describe the same product - they could adopt different sets of attributes or vocabularies to describe the same product. For example: name, classification, brand, price, Accessories ….

Although there are lot of search engines (google.com, baidu.com, yahoo.com, http://ese.rfe.org/, etc.) in the Internet, they can't satisfy the user's need because of their low recall and precision [1], because most search engines are keyword-based. While there is a semantic gap between keywords and concepts, for instance, the same keywords may have different meanings in different contexts, as a result, the returned results only match the user's query in words rather than in concepts.

Especially for the search of product information, the current search engines based on keywords have even low recall ratio because of the problem we have already mentioned that the same product may have different classifications, different names and different descriptions literally. So when a customer wants to purchase a product on the web, he has to browse as many websites as he can to acquire the appropriate information about the desired product's characteristics, attributes, performances, prices and so on. Obviously, it is not conducive to carry out e-commerce transactions as customers have to do so much boring and time-consuming work.

In the literature we found several of semantic information retrieval systems for different fields like distance learning [2], but we found no specific semantic search engine for e-commerce. Our goal is to propose semantic information retrieval system to help the user to acquire

the desired information on desired products and address the problems mentioned above to improve the relevance of IRS (Information retrieval system) by adopting semantic web technologies.

We started by creating a product ontology that represents products's informations, our index model that will be fueled by information extracted from e-shopping pages and suggest a system or research approach based on this ontology.

This article will be divided as follows: we define in the next section some web extraction methods, the subsequent section presents our contribution as well as our information retrieval system, the last section presents conclusion and perspectives.

2 RELATED WORKS

The main source of e-commerce information, ie informations about the products on sale is internet or internet web pages. To respond user requests for information about products and display products that meet their needs, we need to extract information from web pages, filter it and display the result to the user. So the first problem we confront is the extraction of information.
Several approaches have been proposed, in this article we will present some of them that we have found important:
-"Gene / Clone approach, the structural prefix and structural suffix" is a generic approach applicable to all areas.
-"Knowledge-based Wrapper Generation by Using XML" approach: This data extraction method is based on the representation of both the domain knowledge and the wrapper by XML files.
-"A Shopping Agent That Automatically Constructs Wrappers for Semi-Structured Online Vendors", proposes a shopping agent with an inductive learning method, to our knowledge this approach is the only specific to e-commerce.

2.1 Presentation of the approach "Gene / Clone approach, the structural prefix and structural suffix"

The objective of the approach "gene / clone approach, the structural and structural prefix suffix" [3] is to generate the gene that is the smallest repetitive substructure containing all values of example instances.
From this repetitive structure, the search of clones allows to retrieve other instances of the relationship to extract. Instead of searching for relevant information, the approach looks for the relevant nodes.

The process of " GENE / CLONE WITH STRUCTURAL PREFIXES AND SUFFIXES " approach

The approach "GENE / CLONE WITH STRUCTURAL PREFIXES AND SUFFIXES" is used to retrieve information from the Web, it is to use a set of examples bodies and work on the structure of the source documents.It

consists of using a set of examples instances and working on the structure of the source documents.
The application of this method, requires to transform the html pages, from which we want to extract the relevant information into xhtml or xml pages in order to benefit of the tree structure of xml documents.

To know the relevant informations to be extracted, the user uses some examples to express his needs in terms of relevant informations to be extracted. Each example is called instance or instance example.
After the user enters k instances examples:
t1 =(t11 ,t21 ,....,tn1) , t2 =(t12 ,t22 ,....,tn2) ,....., tk =(t1k ,t2k ,....,tnk), we seek the gene generated by each example instance tj that we note g(j) (the smallest repetitive substructure containing all the values of the instance tj), then the valid structural prefixes and the valid structural suffixes of the gene g(1) (gene generated by the first instance) are determined by comparing the structural prefixes and the structural suffixes of the nodes of the gene g (1) with the structural prefixes and the structural suffixes of the nodes of other genes.
Finally to find all clones of the gene and recover the other instances of the relation to extract, all the structural information of the G gene (1) and also the structural valid prefixes and structural valid suffixes that valid structural gene are used.

2.2 Knowledge-based Wrapper Generation by Using XML

This data extraction method [4] is based on the representation of both the domain knowledge and the wrapper by XML files. This approach increase modularity, flexibility, and interoperability by providing more formal representation of extraction rules. The implementation of this approach: XTROS includes several modules: interface module, the learner, and the interpreter.

The interface module

Forms a user query by combining the query template of the site with user inputs, and sends it to the corresponding site to obtain the search result pages in HTML form.

The learner

The learner is the main learning module that generates a wrapper for each site. This module analyzes the content of the HTML page passed from the interface, and constructs extraction rules by referring to the domain knowledge.

Domain Knowledge:

Domain knowledge describes the terms, concepts and relationships used to a particular field of application. Each knowledge domain is represented by a structure <KNOWLEDGE> .. </ KNOWLEDGE> in the XML file witch represents the XTROS's knowledge. Each element of knowledge (KNOWLEDGE) contains a list of objects structure (<OBJECTS> .. </ OBJECTS>) whose values will be the extraction target.

For each object we constitute two components <ONTOLOGY> and <FORMAT>.

<ONTOLOGY> lists the terms that are used to recognize the existence of an object. In the example, the PRICE object has PRICE and $ as its <ONTOLOGY> terms, which means that a line will be recognized as PRICE if it contains the string "PRICE", the symbol $, or both.

<FORMAT> describes the data type of the object value and the positional relationship between the ontological terms and the values. For example, the <FORMAT> of PRICE tells that its data type is the digit, and the price value can appear before or after the ontological term.

The interpreter module:

The interpreter module executes the learned wrappers to extract the specific parts of information from the HTML pages.

Wrapper Generation:

The entry HTML page is composed of three parts: the header (H), the list of item descriptions (L), and the tail (T).

(H) and (T) are redundant and does not contain relevant informaions, are ignored.

(L) is a list of descriptions (IDs) that match the query, it can be expressed as: L = <ID1, ID2, ..., IDn>. An ID consists of several attributes that notes ID = <A1, A2 ...>. In the example of real estate, are represented by attributes: price, number of bedrooms…

The learning phase of the wrapper is designed to recognize the beginning and the end of the position of L, to do this the generation of wrapper is divided into three phases: Converting into Logical Lines, determining the meaning of logical lines, and finding the most frequent reason.

Converting Into Logical Lines:

This first phase consists on breaking the HTML page in a logical line, a logical line is conceptually similar to a line that the user sees in the browser and the learner identifies it by detecting HTML tags (delimiter) such as
, <p>, <li>, <td> and <tr>. All HTML tags except <a href=..> and <img src = ..> are considered unnecessary and therefore removed.

Determining the Meaning of Logical Lines

The second phase of the algorithm is to determine the meaning of each logical line by checking the existence of any object in the domain knowledge. A given logical line is recognized to include an object if it contains any <TERM> in its <ONTOLOGY> specification, and also conforms to any <FORM> in its <FORMAT> definition. When a logical line contains more than one object, the original line is divided into several subparts before applying this module.

Once its meaning is determined, each logical line is represented by a pre-defined data structure. This frame-like data structure has 5 slots: object, line, cat, type, and format. Here, object describes the meaning of the line, line contains the original line with the ontology and the data type attached, cat is a category number assigned to the corresponding object, type is the data type of the object value, and format defines the positional relationship between the ontology and the value.

Finding the Most Frequent Pattern

This step consists on finding the most frequent pattern, we seek all candidates in substrings, and this pattern does not contain the same object twice. The matching pattern is the substring in which the frequency is maximum.

Constructing of XML-based Wrapper

A wrapper based on XML is built based on the most frequent pattern obtained from step "Finding the Most Frequent Pattern" The obtained pattern includes multiple objects, a single XML file is built for each object found in the sample page.

Data Extraction:

The actual information extraction is performed by the interpreter XTROS which parses the XML file wrapper, to build extraction rules, and then apply these rules on the search results pages.

2.3 Shopping Agent That Automatically Constructs Wrappers for Semi-Structured Online Vendors

This method ([5]) proposes a shopping agent with a robust inductive learning method that automatically constructs wrappers for semi structured online stores.

Product descriptions may comprise multiple logical lines

and may have extra or missing attributes. This method treats a logical line as a basic unit, it assign a category to each logical line and recognizes the position and the structure of product descriptions by finding the most frequent pattern from the sequence of logical line information in output HTML pages.

Overview of comparison Shopping Agent

The comparison Shopping Agent prototype is called MORPHEUS, it consists on several modules:
The wrapper generator: is the main learning module that constructs a wrapper for each store, it learns two things:
-it learns how to query a particular store by recognizing its query scheme, it generates a query template.
-it learns how to extract a store's content (Product description).

The wrapper interpreter: is a module that executes learned wrapper to get current product information
The uniform output generator generator: integrates the search results from several stores and generates a uniform output.

Presentation:

The main goal of the wrapper generator is to look for the format or the pattern of product description (booktitle, price, author...) in result pages from successful searches, each result page contains one or more product descriptions which is composed of a sequence of items that describe the attributes of product.

Approach

The method used to find the pattern of the product information is divided into three phases:

1-the HTML page is broken down into logical line. A logical line is conceptually similar to a line that the user sees in the browser, so the algorithm recognizes each logical line by examining HTML's delimiter tags such as
,<p>,<hr>,<table>,...

2-Categorize each logical line and assign it the corresponding category number. Five categories are maintained: TEXT, PRICE, LTAG, TITLE and TTAG,

their category numbers are 0,1,2,3 and 8, respectively.
-TITLE denotes the product name

-PRICE denotes the product price
-TTAG denotes table tags such as <tr>
-LTAG denotes the html tags other than TTAG used in logical line breaking.

-TEXT denotes the price a general string that is not recognizable as one of above four categories.

A simple heuristic is used to assign a category number for each logical line, TITLE is assigned to a logical line when

the line contains one of the keywords in sample query and PRICE is assigned to a logical line by recognizing currency (Dollar, Euro or other symbols that represent price unit).
Example: product description obtaine from the amazon bookstore by the query "Korea"

<a href="/exec/obidos/ASIN/3540618724/qid=95863	3
6791/sr=1-3/103-6540613-6072633">Advances in Cryptology-Asiacrypt	
'96 : International Conference on the Theory and Applications of	
Cryptology and Information Security Kyongju, Korea,)	
 	2
by K. Kim(Editor), Tsutomu Matsumoto (Editor). Paperback (November	0
1996)	
</td>	8
</tr>	8
<tr>	8
<td valign=top width=50%>	8
<font face=verdana,arial,helvetica size=-1> Our Price:$73.95	1

Fig1: An HTML source for a book and the categories for logical lines

After this phase, the entire page is expressed by a sequence of category numbers. 3-it consiste of finding the most frequent pattern, it first finds the pattern of each product description unit (PDU) and counts the frequency of each distinct pattern to get the most frequent one. To find a PDU, if first find the price and then backtracks in the sequence to search TITLE (that suppose that the TITLE attribute appears before the price attribute in PDU), the subsequence between the TITLE and PRICE becomes the resulting pattern of PDU.

2.4 Comparison of methods

Tab1: comparison of the approaches

	Domain knowledge presentation	Instances entered by user	Recall	Precision	Limitation
"Gene / Clone approach, the structural prefix and structural suffix" approach		X	99%	98,7%	-
Knowledge-based Wrapper Generation by Using XML	X		not specified	not specified	It only Works for labeled documents, not functioning on table-type documents.
Shopping Agent That Automatically Constructs Wrappers for Semi-Structured Online Vendors	-	-	not specified	not specified	Each product description must contain the price attribute

The most relevant approach is "Gene / Clone approach, the structural prefix and structural suffix" CF.Tab1 seen its precision and recall are high, while the other approaches we could not calculate these two values in addition to the limitations they present.

In our approach we will use this method to extract data from web pages.

3 APPROACH

The objective is to provide an information retrieval system that responds to user queries. To do this we will take as input a web page, from which we will extract the various product information (informations that will index to respond to user search queries). There are several approaches to develop an information retrieval system, we chose the ontological approach seen the benefits it brings [6].

In the following we will present our new system (cf.Fig2) which is based on a new ontology that we created for this need. An IRS (Information retrieval system) includes a set of procedures and operations that allow the management, storage, querying, searching, selection and representation of this mass of information.

Our IRS respects the general hierarchy of IRS, in what follows the indexing and search parties will be detailed.

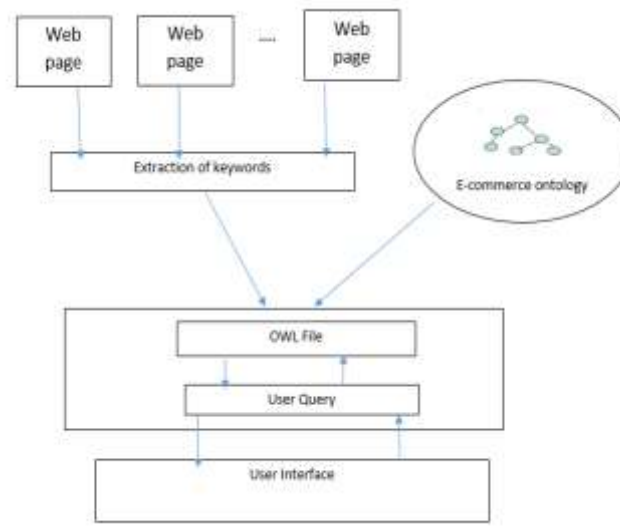


Fig2: Retrieval information System

3.1 Semantic Indexing of e-commerce resources

The semantic indexing will allow us to represent the various product information. The semantic indexing includes two steps:

Disambiguation: get the exact meaning of words extracted from the document to be indexed.

Representation: to represent the document in order to retrieve information.

3.1.1 The Disambiguation

This phase will be done by the approach "GENE / CLONE WITH STRUCTURE prefixes and suffixes" instances we will enter to look for the gene containing different informations we want to extract from the web page (product name, brand , price ...).
The generated gene corresponds to the smallest repetitive structure that contains the description of a single product, from this structure we can extract all the informations about all products displayed in a page.

3.1.2 The Representation

To represent the document, we will use as a model the ontology of products. Ontology was originally a concept of philosophy to study being, ie the study of general properties of what exists. And later the artificial intelligence researchers reused concept for modeling domain knowledge.
Ontology includes concepts that represent all knowledge of a domain in an explicit and formal specification ([7]). It shows the relationships and rules's association between these concepts to allow the one hand to computers the generation of new knowledge through inference, and secondly to users and computer to give common meaning to the terms used in a business area in order to remove any ambiguity during treatments.
As part of our approach we have proposed a new ontology CF.Fig3.

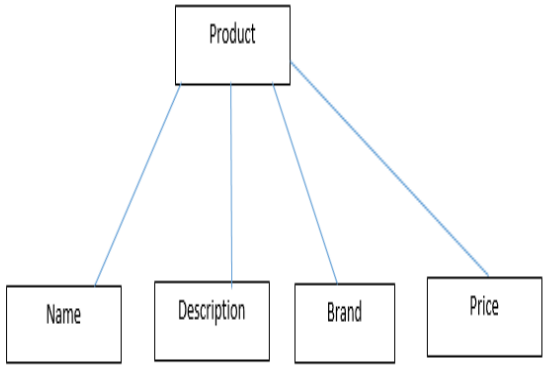


Fig3: Proposed Ontology

This ontology is the most abstract layer of product information, more we specify products more we will have specific attributes for products. To give an overall view of the whole approach including disambiguation and respresen tation, the figure below shows the process :

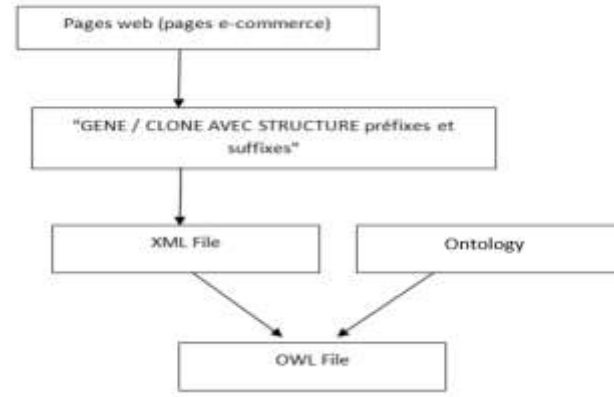


Fig4: The entire process of the approach

Each e-commerce page will be browsed to extract different products's information based on the "GENE / CLONE WITH STRUCTURE prefixes and suffixes" approach and it is this approach that will allow us the disambiguation of information extracted from the page, since for each product information we will match any property of the product (name, price, ...). Once informations are extracted, we will build our model from the ontology of products, which will lead to the creation of an OWL file.

3.2 Search of products

The user begins his search by entering keywords that search (user request), this request will be processed and converted into a SPARQL query, the query will be run on the repository data and returns the list of instances that meet this request, and the link to the relevant documents is established and the result is displayed to the user (CF.Fig5).

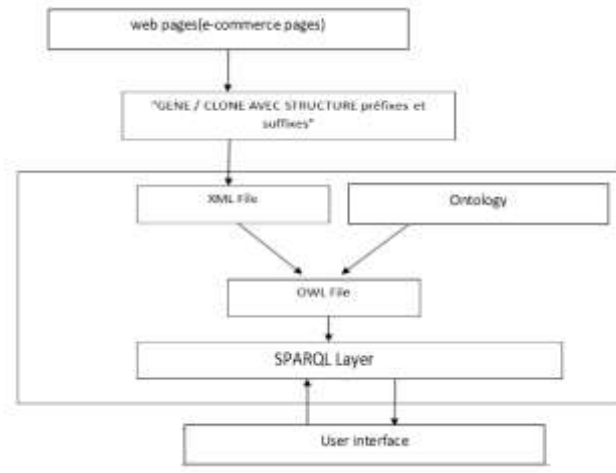
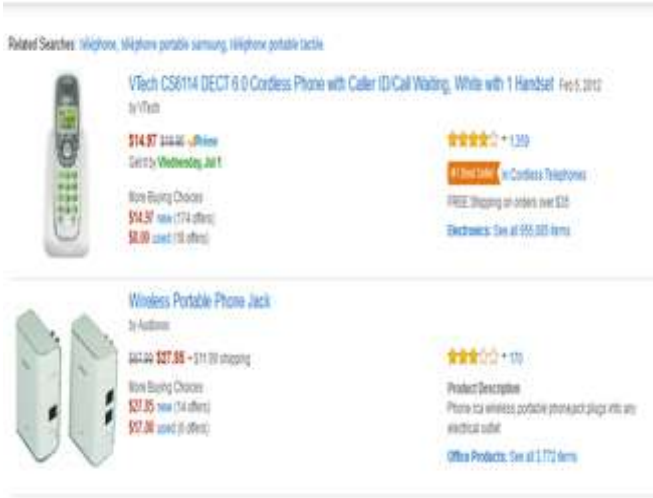


Fig5: Interaction between the user interface and our system

4 APPLICATION ON AMAZON WEBSITE

We will apply our approach to the famous e-commerce



site to ensure the validity of our approach which takes place in several steps:

Step 1: Enter the web page to be indexed (example of Amazon)

Fig6: Amazon products website

Step 2: the gene generated by "Gene / Clone approach, the structural prefix and structural suffix" approach for Amazon

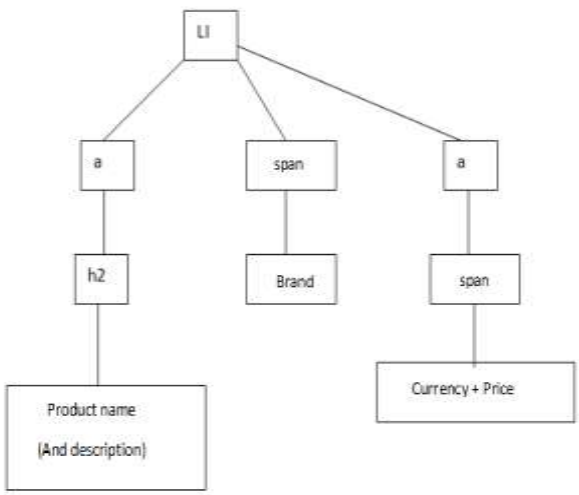


Fig7: Example of the gene for Amazon

Step 3: Ontological representation of the page

We develop this product ontology using the protégé system [8].

```
<?xml version="1.0"?>
<!DOCTYPE Ontology [
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY xml "http://www.w3.org/XML/1998/namespace" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
]>
<Ontology xmlns="http://www.w3.org/2002/07/owl#"
   xml:base="http://www.semanticweb.org/ontologies/e-commerce"
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:xml="http://www.w3.org/XML/1998/namespace"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   ontologyIRI="http://www.semanticweb.org/ontologies/e-commerce"
   versionIRI="http://www.semanticweb.org/ontologies/e-commerce">
   <Prefix name="rdf" IRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#"/>
   <Prefix name="rdfs" IRI="http://www.w3.org/2000/01/rdf-schema#"/>
   <Prefix name="xsd" IRI="http://www.w3.org/2001/XMLSchema#"/>
   <Prefix name="owl" IRI="http://www.w3.org/2002/07/owl#"/>
   <Declaration><Class IRI="#Produit"/>
   </Declaration>
   <Declaration><DataProperty IRI="#Description"/></Declaration>
   <Declaration><DataProperty IRI="#Marque"/></Declaration>
   <Declaration><DataProperty IRI="#Nom"/></Declaration>
   <Declaration><DataProperty IRI="#Prix"/></Declaration>

   <Declaration><NamedIndividual IRI="#Produit1"/></Declaration>
   <Declaration><NamedIndividual IRI="#Produit2"/></Declaration>
   <DataPropertyAssertion>
     <DataProperty IRI="#Description"/>
     <NamedIndividual IRI="#Produit1"/>
     <Literal datatypeIRI="&rdf;PlainLiteral">Free shipping on orders over $35</Literal>
   </DataPropertyAssertion>
   <DataPropertyAssertion>
     <DataProperty IRI="#Marque"/>
     <NamedIndividual IRI="#Produit1"/>
     <Literal datatypeIRI="&rdf;PlainLiteral">VTech</Literal>
    </DataPropertyAssertion>
   <DataPropertyAssertion>
     <DataProperty IRI="#Nom"/>
     <NamedIndividual IRI="#Produit1"/>
     <Literal datatypeIRI="&rdf;PlainLiteral">VTech CS6114 DECT 6.0 Cordless Phone with Caller ID/Call Waiting, White with 1 Handset</Literal>
   </DataPropertyAssertion>
   <DataPropertyAssertion>
     <DataProperty IRI="#Prix"/>
     <NamedIndividual IRI="#Produit1"/>
     <Literal datatypeIRI="&rdf;PlainLiteral">$14,97</Literal>
   </DataPropertyAssertion>
```

```
<DataPropertyAssertion>
  <DataProperty IRI="#Description"/>
  <NamedIndividual IRI="#Produit2"/>

<Literal datatypeIRI="&rdf;PlainLiteral">Phone rca wireless portable phonejack plugs into any electrical outlet </Literal>
</DataPropertyAssertion>
<DataPropertyAssertion>
  <DataProperty IRI="#Marque"/>
  <NamedIndividual IRI="#Produit2"/>
  <Literal datatypeIRI="&rdf;PlainLiteral">Audiovox</Literal>
</DataPropertyAssertion>
<DataPropertyAssertion>
  <DataProperty IRI="#Nom"/>
  <NamedIndividual IRI="#Produit2"/>
  <Literal datatypeIRI="&rdf;PlainLiteral">Wireless Portable Phone Jack</Literal>
</DataPropertyAssertion>
<DataPropertyAssertion>
  <DataProperty IRI="#Prix"/>
  <NamedIndividual IRI="#Produit2"/>
  <Literal datatypeIRI="&rdf;PlainLiteral">$27,85</Literal>
</DataPropertyAssertion>
</Ontology>
```


Step 4: Search of products

Our system supports SPARQL query entered which will be built via the keywords entered by the user, the SPARQL query will then be executed on the OWL file, and the search result will be ranked and presented to the user (CF.Fig8).

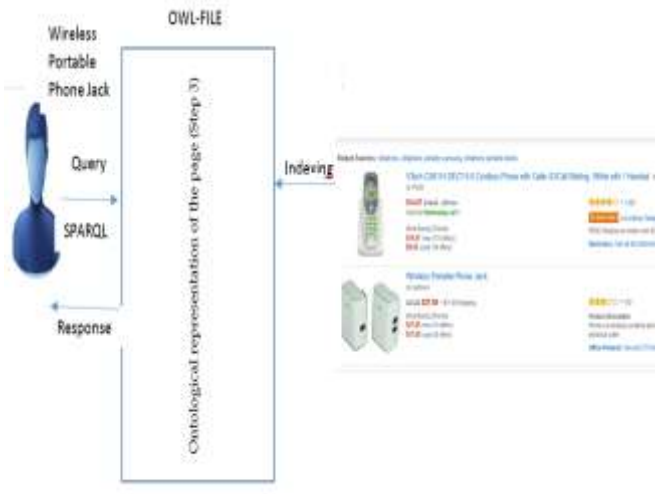


Fig8: Process of search

5 CONCLUSION AND PERSPECTIVES

The result of this work is proposing a new approach that facilitates to the user the search of product information using:

-the data extraction method "GENE / CLONE"

-The proposed Approach of indexing Resources ontology-based

The perspectives for this work, concerning:

Disambiguation phase :

- Propose another approach to facilitate the use of this approach.

The Representation phase :

- Enrichment of the ontology by other product attributes.

Search of products phase :

- Enrich request and response
- Add Search by assistant

The approach :

- Apply this approach on other sites than amazon, having other forms and other languages....

REFERENCES

[1]Kumar, B. T. S., and S. M. Pavithra. "Evaluating the searching capabilities of search engines and metasearch engines: A comparative study." Annals of Library and information Studies 57.2 (2010): 87-97.

[2] Guemmat, K. E., Talea, M., & Lamrani, E. K. (2015). Implementation and Evaluation of an Indexing Model of Teaching and Learning Resources. Procedia-Social and Behavioral Sciences, 191, 1266-1274.

[3]Chara Aziz,Benlahmer EL Habib,Abderrahim Tragha,Mohamed Rachdi-Kamal elguemmat :Generate Tools through a Data Extraction Web Example of the production of meta social network, VOL.2 N°9, September 2012

[4]Seo, Heekyoung, Jaeyoung Yang, and Joongmin Choi. "Knowledge-based wrapper generation by using XML." IJCAI-2001 Workshop on Adaptive Text Extraction and Mining. 2001.

[5]Jaeyoung Yang, Eunseok Lee, and Joongmin Choi : A Shopping Agent That Automatically Constructs Wrappers for Semi-Structured Online Vendors

[6]R.Aravindhan, Mano Chitra.M. "A Review on Ontology Based Search Engine", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 10, October 2014

[7] R. STUDER, R. BENJAMINS, AND D. FENSEL. KNOWLEDGE ENGINEERING: PRINCIPLES AND METHODS. DATA & KNOWLEDGE ENGINEERING, 25(1–2):161–198, 1998.

[8] Protégé, http://protege.stanford.edu/.

[9]Liyi Zhang,Mingzhu Zhu, and Wei Huang : A Framework for an Ontology-based E-commerce Product Information Retrieval System,JOURNAL OF COMPUTERS, VOL. 4, N°6, JUNE 2009

[10]Heekyoung Seo, Jaeyoung Yang and Joongmin Choi : Knowledge-based Wrapper Generation by Using XML, HCILab,Samsung Advanced Institute of Technology,KOREA

[11]Benlahmer EL Habib,Doukkali Seduigi Aziz,El ouerkhaoui Asmaa ,A new solution for data extraction : Gene/clone Method, In International Journal of Computer Science and Network Security, VOL.6 No.7A, July 2006

Mohamed Talea *was born in Casablanca, Morocco in 1964, Professor of Higher Education at the Faculty of Sciences Ben M'Sik, UNIVERSITY HASSAN II MOROCCO CASABLANCA. He obtained his PhD in collaboration with the LMP laboratory in Poitiers University, FRANCE in 2001. He obtained a Doctorate of High Graduate Studies degree at the University Hassan II-Mohammedia in 1994. Actually he is the Director of Information Treatment Laboratory. He has published many papers in conferences and national and international journals. His search major field is on Systems engineering, in security of system information.*

. Abdelhadi BAHAFID *was born in 1985 in Casablanca in Morocco, is a PhD candidate of Mathematics, Informatics and Information Processing in the faculty of science Ben M'sik, Hassan II University, Casablanca, Morocco, took his engineering Degree in software engineering, from ENSIAS (National School of Computer Science and Systems Analysis)-Rabat, Morocco. His interesting area includes: Semantic Indexing, Semantic Web, and Information Retrieval Systems.*

. Kamal EL GUEMMAT, *PhD in computer sciences, Research fields: Semantic Indexing, Semantic Web, Information Retrieval Systems, Automatic Processing of Natural Language, Information and communication technology for teaching and learning, Educational modeling, E-Learning*

BENLAHMAR Elhabib, *Professor of Higher Education at the Faculty of Sciences Ben Sik, Casablanca, Morocco, Research interests: Metasearch, Information retrieval , Semantic web, Automatic Processing of Natural Language.*