A Framework for an Ontology-based E-commerce Product Information Retrieval System

Liyi Zhang*
Center for Studies of Information Resources, Wuhan University, Wuhan, China
Email: lyzhang@whu.edu.cn

Mingzhu Zhu[1] and Wei Huang[1, 2]
1. School of Information Management, Wuhan University, Wuhan, China
2. School of Management, Hubei University of Technology, Wuhan, China
Email: {zhumzhu, tonny_hw}@163.com

***Abstract*—With the rapid development of e-commerce, online shopping has become an important part in people's lives, in order to support the smooth development of e-commerce activities, how to provide users with an efficient and practical product information search method has become an urgent and critical problem. This paper presents a framework for an ontology-based e-commerce product information retrieval system and proposes an ontology-based adaptation of the classical Vector Space Model with the consideration of the weight of product attribute. A computer and components related ontology has been built, which is adopted to annotate the html documents and construct concept vectors of the documents. Then the system test is done and the experimental result indicates that our proposal is better than the traditional keywords based search.**

***Index Terms*—Information retrieval, Product information, E-commerce, Ontology**

I. INTRODUCTION

With the rapid development of e-commerce, the number of online stores is skyrocketing, and the commodity information in the Internet is much richer than ever before. As online shopping has become an important part in people's lives, the product information retrieval mechanism is become more and more important, because information retrieval is the most frequently used method to obtain information in the web, and the purchaser must get access to the product information before a transaction is carried out. However, the current way of information organization and expression is defective, in that it was designed to meet the user's reading needs, rather than to provide semantic information that computer can process automatically, thereby limiting the computer's capacity of automatic analysis and further intelligent process in information retrieval [1].

Refer to electronic commerce (EC), there are many issues that still exist, such as acquiring and storing information, finding and filtering information, securing information, auditing access, universal access, cost management and financial instruments [2]. Among these issues, finding and filtering information is essentially important to online stores as customers need online facilities to help them retrieve information and locate resources that match their expectations and desires. Particularly, customers would like to find products and services at low costs, using languages and terminologies they are most familiar with. However, the rich and diverse descriptions that vendors use to describe their products increase the difficulty of locating products and services accurately and efficiently. As different vendors may adopt different ways to describe the same product - they might adopt different sets of attributes or vocabularies to describe the same product. For instance, (name, classification, brand, accessories) may be a scheme - a set of attributes and their corresponding domains - for computers.

Although there are lots of search engines (google.com, baidu.com, yahoo.com, etc.) in the Internet, they can't satisfy the users' need because of their low recall and precision, because most search engines are keyword-based. While there is a semantic gap between keywords and concept, 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 aforementioned that the same product may have different classifications, different names and different descriptions literally. So when a customer wants to purchase a certain product on the web, he has to browse as many websites as he can to acquire the appropriate information about the desired product's parameters, 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 order to facilitate customers to acquire the desired product information on the Web, we adopt some semantic technologies such as ontology, OWL [3] and SPARQL [4] to enhance the performance of product information retrieval. Firstly, we develop a computer and components related ontology using the protégé system [5]. Then we present a

* Corresponding author. Tel: 86-13607166827; fax: 86-02768752135

framework for an ontology-based system which provides product information retrieval service for costumers in B2C marketplace. This system provides users with two kinds of search patterns, one is OA-VSM based, and the other is SPARQL based. Finally, we make a test of the system and give a brief discussion of our approach after the description of the architecture of the framework and the proposal of an ontology-based adaptation of the classical Vector Space Model (OA-VSM).

The following parts of this paper are as follows. Section 2 introduces related work. Section 3 gives a brief description of the product ontology. Section 4 delineates the architecture of the ontology-based product information retrieval framework, including the details of the proposed OA-VSM approach and the description of SRARQL-based product information retrieval. Section 5 describes the experiment and the results achieved. Finally, section 6 concludes the paper and discusses future work.

II. Related work

The related work to our approach comes from two main areas: Ontology and semantic information retrieval.

Ontology is regarded as a very good solution in information management field which has the problem that the continued growth in information volume, which makes difficult to find, access and maintain information. The use of ontology-based applications is being pointed as one of the most promising ways to deal with this problem [6]. However, in the remaining part of this section we will concentrate on work that is directed at semantic information retrieval applications.

Qiu and Frei [7] are using query expansion based on similarity thesaurus and adopting weighting of terms to reflect the domain knowledge. The query expansion is done by similarity measures. Similarly, Grootjen and Weide [8] describe a conceptual query expansion. There, the query concepts are created from a result set. Both approaches show an improvement compared to simple term based queries, especially for short queries.

OntoSeek system [9] attempts to provide users with interactive semantics query interface by regarding ontology as domain vocabulary with semantics and integrating ontologies with lexicon. Another information retrieval system [10] takes full advantage of ontology, which expands the requirement of users to the semantic words set and provides the document analyzer that can filter the Web pages returned by the search agent, so presents the most relevant documents to the users. An information retrieval server [11] based on multi-agent and ontology integrates several kinds of agents, such as information processing agent with mobile ability, and uses ontologies to classify the domains of documents and assist users to normalize their queries.

Popov et al. [12] define a general framework for document retrieval that is supported by ontology, and integrate full-text search with ontology-based methods. Castells et al. [13] present a system that documents are connected with ontology instances via weighted annotations. They include documents and annotation to the ontology, and directly use the annotation weights to calculate semantic document relevance using the classical *tf-idf* scheme, after executing the ontology-based query. They execute a full-text search separately and combine its returned relevance weight with the semantic query result to diminish the effect of ontology imperfection.

Rocha et al. [14] present an approach to semantic search based on a combination of text search and spread activation. In the first step the user's query terms are used as a query into the KB (consisting of metadata for concept instances) and a set of concepts denoted as start nodes for the spread activation step are returned. Next spread activation is applied to a graph where the concepts are represented as nodes and the relations are represented as edges. The search result returned to the user contains nodes (instances) of the ontology that have a semantic similarity to the query, although the similarity may not explicitly be provided through the user's query terms.

Guha et al. [15] present a view of the Semantic Web where documents and concepts are nodes alike in a semantic network. There are two main problems addressed by [15], the first is the development of a distributed query infrastructure for ontology data in the Semantic Web and the second is the presentation of query execution results, augmenting query answers with data from surrounding nodes.

Lei et al. [16] present a system that lets the user specify queries as keywords, and hides the actual semantics from the user. The keyword query is first disambiguated by trying to find the semantic meaning of the query terms. This is done by performing a text search on the semantic entities (concepts, relations, etc.), returning all matching entities. These are next used to construct formal queries which are queried against the semantic repository. Finally the results are ranked and presented to the user.

Some approaches to ontology based IR can be further sub-divided into Knowledge Base (KB) and vector space model driven approaches which use reasoning mechanism and ontological query languages to retrieve instances. Documents are treated either as instances or are annotated using ontology instances [14], [17], [18]. These approaches focus on retrieving instances rather than documents.

Semantic Portals [19], [20], [21] typically provide simple search functionalities that may be better characterised as semantic data retrieval, rather than semantic information retrieval. As they return ontology instances rather than documents to users, and no ranking method is provided.

III. Product ontology

Ontology was originally a philosophy concept to study the essence of the existence and compositions of objectives [22], and later, researchers in artificial intelligence borrowed this concept for the modeling of domain knowledge. Ontology gives a systematic explanation to the entity existing [23]. Studer et al. [24] propose that ontology is explicit, formatted criterion and explanation that is conceptualized commonly. Its goal is to capture the relevant knowledge in the field, provide common understanding of the domain knowledge, identify common recognition of the vocabulary, and give a clear definition for these terms from different levels and different patterns.

In the field of information and knowledge management, ontology is a conceptual model that is used to express and describe the common knowledge in some areas. Refer to "ontology", we may consider any formalism with a well-defined mathematical interpretation which is capable at least to represent a sub-concept taxonomy, concept instances and user-defined relations between concepts. They represent knowledge on the semantic level, i.e., they contain semantic entities (concepts, relations and instances) instead of simple words. Moreover, they allow specifying custom semantic relations between entities, and also to store well-known facts and axioms about a knowledge domain (including temporal information). This additional expression power allows the identification of the validity context of specific relations.

Ontologies are becoming increasingly important as a component of online commerce offerings. They are useful and arguably necessary in supporting at least navigation, browsing, user-expectation setting, and parametric search. Sources of class taxonomies exist, tools for piecing ontologies together are growing, and some sources of parameter information are becoming available. Challenges remain for users in reusing available ontological information, because as standards are still forming, most vocabulary information needs to be augmented, and although some tools exist, most are still on a development path to becoming complete tool suites suitable for mass deployment. These challenges are surmountable and they should diminish over a short time [25].

Ontology can be constructed for product information retrieval, as product ontology provides the participants in e-commerce activities with a clear and accurate product model which contains rich semantic information, so it can be the basis of the realization of ontology-based product information retrieval. Because product ontology defines the structure of the concept of the product classification strictly, and each concept is defined with all of the corresponding attributes, this makes computers easier to deal with the product data automatically.

In order to facilitate the search and comparison of products information provided by multiple suppliers, a number of standard setting organizations have issued several product ontologies, which are used to guide businesses for the construction of various product classification and the establishment of product information model. There are many product related ontologies at present, such as UNSPSC [26], and eCl@ss [27]. These ontologies have a fairly comprehensive classification level, but they are lack of detailed definitions of product attributes. In the process of developing a product ontology, we try to reuse the product classification defined in the UNSPSC ontology, and add some new classifications and detailed product attributes as well as domain knowledge so as to support semantic search.

We have developed a personal computer and components (PCC) ontology as the basic of the EC product information retrieval system. In the developing process, we followed the method described in [28]. We got the classes of the computer and components by surveying and analyzing several B2C (business to customer) websites, then we integrated these classes to get the core classes, such as computer, PC, notebook PC, and mainboard. According to the knowledge of computer domain, we constructed the class relationship, such as "is-a", "has_part", "has_cpu", and "has_mainboard". Part of the PCC ontology is shown by Fig. 1.

Regarding the choice of the ontology language used to define the ontology, OWL was selected because it is a W3C recommendation and it has the quality of maturity, sufficiency and expressiveness. The decision to use OWL instead of RDF is also because that the ontology contains inverseOf××× properties, and OWL supports some attributes RDF does not support, such as cardinality constraints and data types. Among the ontology development tools available with support for OWL, protégé was selected because of its maturity, ease-of-use and, what is more important, its scalability and extensibility.

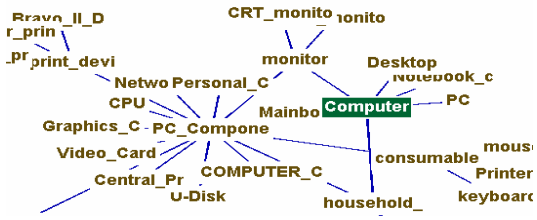

Figure 1. Part of the PCC ontology structure.

IV. ONTOLOGY-BASED PRODUCT INFORMATION RETRIEVAL FRAMEWORK

Product information, traditionally, is displayed in static html pages or dynamically generated html pages. The html pages are full of labels that control display style of the content, but lack of metadata that can describe the structure of the content. As a result, it is hard for the computer to process it automatically. We propose an information retrieval framework, based on which a system is developed. It is a product information retrieval system based on the product metadata constructed by firstly extracting the product information fragment from the html pages and then annotating it using the PCC ontology.

A. Architecture

As Fig. 2 shows, the proposed product information retrieval framework is based on a 3-layered architecture. At the first level, components based on DOM [29] use the predefined rules to extract the product attribute information from the html pages, such as the product's name, brand, parameter, and price, and then the annotation tools use the PCC ontology annotate the extracted information to construct the product data repository which is formatted by OWL. At the intermediate level, an annotated data repository provides access to this metadata and uses it provide users with two kinds of search services, including OA-VSM-based search and SPARQL-based search which will be discussed in detail in the next section. At the third level, we provide users with various search entrance based on keywords and graphical user interface.

B. Information extraction and annotation

There is plenty of product information on the Web, taking B2C websites as an example, each site publishes a large amount of product information, it is critical for information retrieval to obtain the information and analyze it. We have chosen several typical B2C websites (dangdang.com,

amazon.cn, etc.) and located in the section of computer and accessories. At present we have developed a vertical grabber, by which plenty of product pages related to computer and accessories section have been downloaded from some typical B2C websites.

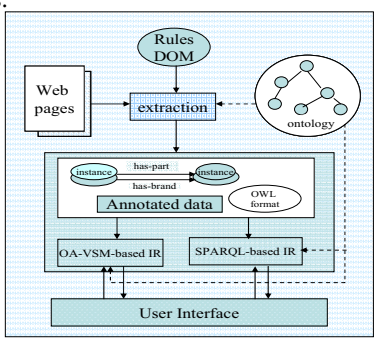

Figure 2. Architecture of the product IR framework.

The structure of product information varies in different websites, In order to process the product information that varies in structure in a unified way, we must unify its structure. After investigating several websites we find that most of the html tags in the product pages have certain features, according to these features we have developed an extractor which is used to extract product attribute information from the product pages, then the html pages are transferred into OWL formatted documents by taking advantage of the PCC ontology. Fig. 3 shows the process of extraction and annotation of product information.

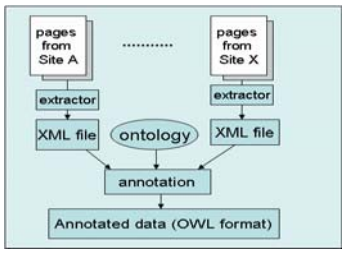

Figure 3. Product information extraction and annotation process.

Fig. 4 shows an example of extracting information from product related pages of a B2C website. The left side contains a sample picture of product information. It is embedded in a page that contains menus, advertisement and other nonrelevant information. On the right hand the resulting content object is shown. In this sample example the product's name, brand, price and other basic parameters are extracted and annotated.

C. Product information retrieval

1) Ontology-based adaptaion of the Vector Space Model

The Vector Space Model (VSM) [30] is a classic statistical model treats a document as a collection of keywords and considers about the frequency information. This model proposes a framework in which partial matching is possible. This is accomplished by assigning weights to index terms in queries and in documents. These terms weights are ultimately used to compute the degree of similarity between each document and the user query. For vector model, the weight w_{ij} associated with a pair (t_i, d_j) is positive. Further, the index

Figure 4. Example for product information extraction and annotation

terms in the query are also weighted, named w_{iq}. Then the query vector is defined as $\vec{q} = \{w_{1q}, w_{2q}, \ldots\ldots, w_{nq}\}$ and the document vector is defined as $\vec{d}_j = \{w_{1j}, w_{2j}, \ldots\ldots, w_{nj}\}$, where n is the total number of index terms in the system. The weights are calculated by most frequently used *tf-idf* scheme [31]:

$$w_{ij} = tf_{ij} \times idf_i \tag{1}$$

where $tf_{ij} = \dfrac{freq_{ij}}{\max_t freq_{tj}}$ and $idf_i = \log\left(\dfrac{N}{n_i}\right)$, $freq_{ij}$ is the number of occurrences of term t_i in document d_j, N is the number of documents in the system, and n_i is the document frequency for term t_i in the system.

The vector space model proposes to evaluate the degree of similarity of the document d_j with regard to the query q as the correlation between the vectors $\vec{d}_j$ and $\vec{q}$. This correlation can be qualified by the cosine of the angle between those two vectors, that is,

$$sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{\left|\vec{d}_j\right| \times \left|\vec{q}\right|} = \frac{\sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{t} w^2{}_{ij}} \times \sqrt{\sum_{i=1}^{t} w^2{}_{iq}}} \tag{2}$$

where $\left|\vec{d}_j\right|$ and $\left|\vec{q}\right|$ are the norms of the document and query vectors.

a) *An adaptation of the tf-idf scheme*

The traditional *tf-idf* scheme has the disadvantage that when it calculates the weights, it only considers the frequency of occurrence of index terms, it neither considers the semantic relationship among them nor considers their location in the document. In an html document, generally, the terms in the title tag are more important than other terms, as they reflect the main content of the page [32]. In particular, in product related pages, the terms that represent the product's attributes (name, brand, price, etc.) should have a higher degree of weights than others. This is because the terms that present the product's attribute can distinguish the objective pages from irrelevant ones better. That is why we propose an adaptation of the classic Vector Space Model. In this model we make a difference

between the important terms that represent the product attribute and the normal terms. To distinguish the terms, the annotated document is adopted.

To illustrate our approach, we divide the product related document d_j into three parts, the first part is the product's name, we assume that the terms presents the product's name have the highest degree, which is named λ_1, as they are most likely to describe the product document. The second part is the normal attributes of the product (brand, components, etc.), the terms represent them have secondary degree, which is named λ_2. The third part is the description and some other information about the product, the terms present them have the lowest degree, which is named λ_3.

Let $freq_{ij\lambda 1}$ be the raw frequency of term t_i in the first part of d_j, $freq_{ij\,\lambda 2}$ be the raw frequency of term t_i in the second part of d_j, $freq_{i\,j\lambda 3}$ be the raw frequency of term t_i in the third part of d_j. Then the adapted frequency of term t_i in document d_j, which is named f_{ij} (or $f_{(ti,dj)}$), can be derived from (3).

$$f_{ij} = \frac{\sum_{t=1}^{3} freq_{ij\lambda_t} \times \lambda_t}{\sum_{t=1}^{3} \lambda_t} \tag{3}$$

For the value of parameter λ_1, λ_2 and λ_3, in this paper, we just set $\lambda_1=0.6$, $\lambda_2=0.3$, and $\lambda_3=0.1$ based on their importance degree.

To illustrate the rationality of this proposal, we give a simple fictitious example to show how our proposal is useful in product relate field. We assume that there are two product documents named d_1 and d_2. After the data preprocessing such as the filtering of stop words and the words segmentation there are 10 words in d_1 and d_2 respectively. Table Ⅰ shows this in detail.

TABLE I. A SIMPLE FICTITIOUS EXAMPLE OF TWO DOCUMENTS

The value of λ_t(t=1,2,3)	The words in d_1	The words in d_2
λ_1=0.6	a, b	b, c
λ_2=0.3	b, c, d, f	a, b, d , e
λ_3=0.1	c, d, e, c	a, b, a, c

We assume that, term *a* presents the main information of d_1, and term *c* presents the main information of d_2. Using the traditional *tf-idf* scheme, we can get $freq_{(a,d1)}$=1, $freq_{(a,d2)}$=3, $freq_{(c,\ d1)}$=3, $freq_{(c,\ d2)}$=2. Using our approach, we can get $f_{(a,\ d1)}$=0.6, $f_{(a,\ d2)}$=0.5, $f_{(c,d1)}$=0.5, $f_{(c,\ d2)}$=0.7. As $freq_{(a,\ d1)}<freq_{(a,\ d2)}$, $f_{(a,d1)}>f_{(a,\ d2)}$, $freq_{(c,\ d1)}>freq$(c,d2), $f_{(c,\ d1)}<f_{(c,\ d2)}$, if users use keywords a or b to make a query, the proposed approach can be more consilient to that query.

b) *The IR process of OA-VSM*

In information retrieval the mismatching of retrieval words to original words is often happened. That because people often use different words to describe the same concept. Human perceptions and comprehensions are based on concepts, which are then used to form structuralized textual content by means of logical organization and textual presentation to record or express thoughts and ideas [33].

There are semantic gaps between keywords and concepts, which causes the low efficiency (low recall and precision) of keywords-based information retrieval. We propose an ontology-based adaptation of the classical Vector Space Model (OA-VSM), with the difference that vector terms are concepts instead of words in a natural language. A schematic description of OA-VSM is shown in Fig. 5. This study employs ontology related techniques in the procedures of concepts determination and concepts weighting. Firstly, domain ontology is used to identify concepts, and it is named concept determination that aims to identify concepts hidden in the textual information. Then, the weighting of each concept is calculated based on the frequency of occurrence and location of words denoting the corresponding concept. To find some relevant documents to a specific query, the query translation procedure is carried out and the similarity between a query and a document is computed, and ranking is made and the results are returned to the user according to the similarity.

c) *Concept determination*

This study adopts the PCC ontology for concept extraction and provides the ontology-based concept determination process which is described as follows:

preprocess: Let N be the total number of documents in the system. For each document d_j ($j \in [1, N]$), the preprocessor eliminates the meaningless words like pronouns, articles, prepositions, conjunctions, and stop words according to stop word list after word segmentation. After that, we get terms set D_j regarding with document d_j. Then let $D=\bigcup_{j=1}^{N} D_j$, and it will be used as input element in the next step.

input: <O,D>O represents the ontology.

begin

for each term t_i ($i \in [1, N]$) in D

find all the classes that matches t_i in O. The results is a concept set named C_{t_i} ={C_1, C_2, C_3,…,C_k }.

if $C_{t_i} = \varnothing$

gather those t_i into candidate terms table which is used to record terms in the document that are not in the domain ontology. The candidate term table will be used to identify those terms with a high occurrence rate but excluded from the ontology, so that we can determine whether the ontology should be updated.

end if

if ($C_{t_i} \neq \varnothing$)

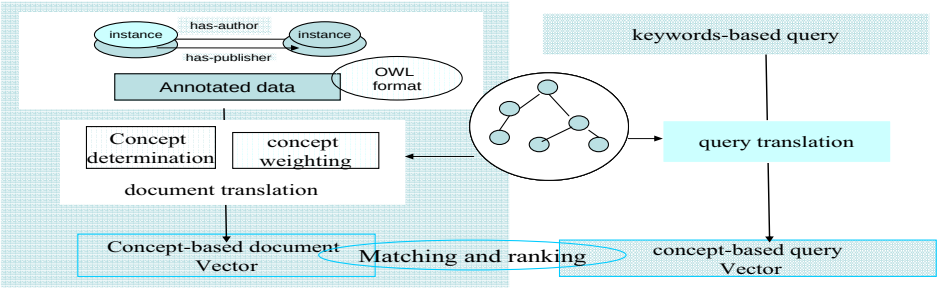

Figure 5. The IR process of OA-VSM.

Label t_i to each of the corresponding concept.

end if

end for

get the union set of C_{t_i}, that is $C=\bigcup_{i=1}^{N} C_{t_i}$.

for each item in C

gather those words with the same labeled concept into the corresponding terms set $C_{i(t)}$, that is $C_{i(t)}=\{t_{i1},t_{i2},\ldots.\}$.

end for

end

Output: $C=\{C_1,C_2,C_3\ldots..C_m\}$, and $C_{i(t)}=\{t_{i1},t_{i2},\ldots\}$, where $C_{i(t)}$ is the corresponding terms set to C_i, C represents all the concepts that the system includes, and t_{ik} represents the word that can be used to denote concept C_i in the system. Upon the completion of concept determination procedure, all concepts in the system and their corresponding words will have been identified through the domain ontology.

d) *Concept weighting*

After the concept determination procedure, we get all the concepts of the system, and the terms set regards with .each concept. Thereby we translate the documents from *n* dimensions of index terms vector space into *m* dimensions of concept vector space. And each document will have a concept set C, which consists of m concepts, and $C=\{C_1,C_2,..C_m\}$. As a concept is denoted in words, each concept will have a word set name $C_{i(t)}$, and $C_{i(t)}=\{t_{i1},t_{i2},\ldots.t_{ik}\}$.

And then, the aforesaid outcomes are used to compute the weighting of a concept based on the frequency of occurrence and location of words denoting that concept. The frequency of appearance of concept C_i in d_j is

$$freq(C_{ij}) = \sum_{t=1}^{k} f_{i_t j} \tag{4}$$

where $f_{i_t j}$ is the adapted frequency of word *t* that denotes the concept C_i in d_j, it is computed according to (3), and *k* is the total number of words denoting C_i in d_j. Thereby the weight of concept C_i can be calculated by

$$w_j(c_i) = tf_j(c_i) \times idf(c_i) \tag{5}$$

where $tf_j(c_i) = \dfrac{freq(C_{ij})}{\max_t freq(C_{tj})}$ and

$idf(c_i) = \log\left(\dfrac{N_c}{n_{ci}}\right)$, N_c is the number of documents in the system, and n_{c_i} is the document frequency for concept C_i in the system.

e) *Query translation, matching and ranking*

As all the documents in the system are presented as an *m* dimensions vectors, to find some relevant documents to a specific query *q*, it is necessary to represent the query *q* in the same way as a document d_j (i.e. vector of concept weights). Similarity between a query *q* and a document d_j is computed as cosine of those two normalized vectors.

$$Sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{\left|\vec{d}_j\right| \times \left|\vec{q}\right|}. \tag{6}$$

2) SPARQL-based product information retrieval

To give the users a more precise way to search the product information, we provide them with a SPARQL-based IR component that facilitates the accurate search through the product's attributes information. The overall retrieval process is illustrated in Fig. 6. Our system takes input as a formal SPARQL query which could be constructed by using keywords or a graphic user interface. The SPARQL query is executed against the annotated data repository, which returns a list of instances that satisfy the query. Finally, the related documents are retrieved, ranked, and presented to the user. The SPARQL query can express conditions involving ontology instances, product attributes (name, price, brand, accessories, etc.)

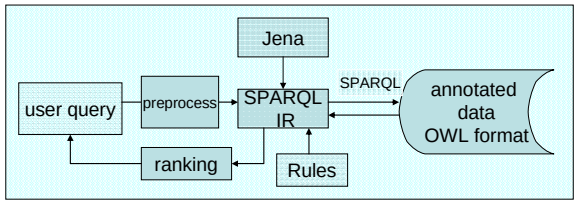

Figure 6. The process of SPARQL-based IR.

or classifications. For instance, Fig. 7 shows a simple example that if a user wants to query a product relating to computer with the condition that its type is “noteBook”(笔记本), name is “联想 IdeaPad Y430”, brand is “lenovo” (联想）, and price is not more than 5000 RMB.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-
syntax-ns#>
PREFIX whuEc: <http://www.ec.whu.ecu.cn/pc.owl#>
SELECT ?products ?price ?brand
WHERE{
    ?products rdf:type whuEc:noteBook.
    ?products whuEc:has_name ?name.
    ?products whuEc:has_brand ?brand.
    ?brand whuEc:has_price ?price.
    FILTER ( (?price<=5000).
             REGEX(?name, "联想 IdeaPad Y430")
            )
}
```

Figure 7. An exmaple of SPARQL query.

The SPARQL query is executed by adopting the Jena framework [34], RACER system [35] and predefined rules. The core code is list as follows:

```
...
FileManager fm=FileManager.get();
Model schema = fm.loadModel(fileSchema);
Model data = fm.loadModel(fileData);
Reasoner reasoner = ReasonerRegistry.getOWLReasoner();
reasoner = reasoner.bindSchema(schema);
InfModel m = ModelFactory.createInfModel(reasoner,data);
OntModel om=ModelFactory.
createOntologyModel(OntModelSpec.OWL_MEM,m);
…
```

V. Performance Evaluation

We have developed a prototype system based on java platform by using Jena toolkit and Jakarta Lucene library [36]. Our system adopts a simplified Chinese word segmentation toolkit developed by ICTCLAS (Institute Computing Technology, Chinese Lexical Analysis System) to segment the simplified Chinese corpus into indexing terms. The average accuracy of the toolkit is about 98% [37].

To construct the annotated data repository, about 5000 computer and components related pages have been downloaded from typical B2C websites (dangdang.com, amazon.cn, etc.). In these documents, about 1500 pages are related to desktop, about 1000 pages are related to laptop, and the others are related to computer components.

The primary factor which is used to evaluate the performance of the retrieval system is the recall and precision. We have tested our proposal with the downloaded pages, and compared it to the VSM-based keyword-only search. Fig. 8 depicts a graph showing the average performance of OA-VSM approach and traditional VSM approach in e-commerce product information retrieval. Our experiment shows that OA-VSM-based approach performs much better than the tradition VSM-based one. However, this approach relies on some different variables (λ_i) which have not been researched yet, so we will do more research to find out how these variables will influence the search results.

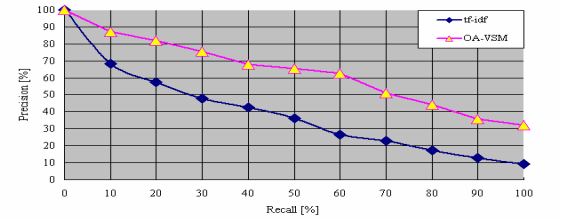

Figure 8. Average recall versus precision figures.

VI. Conclusions

This paper describes a product information retrieval framework based on OA-VSM and its architecture. Based on the framework, a product information retrieval system which is intended to search the product information in B2C environment is developed. The main objective of this system is to facilitate users to search the object product when they carry out e-commerce activities. Because our proposal provide users with two kinds of retrieval methods (OA-VSM-based retrieval and SPARQL-based retrieval), these two methods form a balanced whole so that the users can get a better result.

The defects of our proposal is that we only crawl product related pages from the predefined websites, and the extraction and annotation scheme is just for the specific websites, it deeply depends on the structure of the documents. So in order to improve the performance of the system, a more powerful tool for information gathering should be used in the future. At the same time, the extraction and annotation of a large number of complex unstructured data still requires further study and improvement.

Acknowledgment

The first author was supported in part by the MOE Project of Key Research Institute of Humanities and Social Science in Chinese Universities (NO: 07JJD870220). The authors would like to express our sincere gratitude to the contributing author and to the referees for reviewing papers for this special issue.

References

[1] Berners-Lee T, Hendler J, and Lassila O, “The semantic web,” Sci Am., vol. 284, no.5, pp. 34-43, 2001.

[2] N. Adam and Y. Yesha, “Strategic Directions in Electronic Commerce and Digital Libraries: Towards a Digital Agora,” ACM Computing Surveys, vol. 28, no. 4, pp. 818–835, Dec. 1996.

[3] D. L. McGuinness and F. van Harmelen, OWL Web Ontology Language Overview. W3C Recommendation, 2004.

[4] E. Prud’hommeaux and A. Seaborne, “SPARQL Query Language for RDF,” W3C working draft, http://www.w3.org/TR/rdf-sparql-query, 2006.

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

[6] R. Masuoka, Y. Labrou, B. Parsia, and E. Sirin, "Ontology-enabled pervasive computing applications," IEEE Intelligent Systems, vol. 18, no. 5, pp. 68-72, Sep. 2003.

[7] Qiu, Y., Frei, H., "Concept based query expansion," Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 160-169, 1993.

[8] F. A. Grootjen, and Th. P. van der Weide, "Conceptual query expansion," Data & Knowledge Engineering, vol. 56, no.2, pp. 174-193 , Feb. 2006.

[9] N. Guarino, C. Masolo, and G. Vetere, "OntoSeek: Content-Based Access to the Web," IEEE Intelligent Systems, vol. 14, no. 3, pp. 70-80. May 1999.

[10] Jie Wan, and Zhiyang Teng, "Application of Ontology in Content-based Information Retrieval," Computer Engineering, vol. 29, no.4, pp. 122-124, 2003.

[11] Chenggang Wu, Wenpin Jiao, Qijia Tian, and Zhongzhi Shi, "An Information Retrieval Server Based on Ontology and Multi-Agent," Journal of Computer Research and Development, vo. 38, no. 6, pp. 641-647, 2001.

[12] B. Popov, A. Kiryakov, D. Ognyanoff, D. Manov, and A. Kirilov, "KIM - a semantic platform for information extraction and retrieval," Natural Language Engineeering, vol. 10(3-4), pp. 375-392, Sep. 2004.

[13] P. Castells, M. Fernandez, and D Vallet, "An Adaptation of the Vector-Space Model for Ontology-Based Information Retrieval," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 2, pp. 261-272, Feb. 2007.

[14] C. Rocha, D. Schwabe, and M. P. Aragao, "A Hybrid Approach for Searching in the Semantic Web," Proceedings of the 13th International Conference on World Wide Web, pp. 374-383, 2004.

[15] R. Guha, R. McCool, and E. Miller, "Semantic Search," Proceedings of the 12th international conference on World Wide Web, pp. 700-709, 2003.

[16] Lei, Y., Uren, V., and Motta, E., "SemSearch: A Search Engine for the Semantic Web," EKAW, vol. 4248, pp. 238-245, 2006.

[17] J-F. Song, W-M. Zhang, W. Xiao, G-H. Li, and Z-N. Xu, "Ontology-Based Information Retrieval Model for the Semantic Web," Proceedings of EEE 2005 (EEE'05), pp. 152-155, 2005.

[18] C. Ciorăscu, I. Ciorăscu and K. Stoffel,, "knOWLer - Ontological Support for Information Retrieval Systems," Proceedings of 26th Annual International ACM SIGIR Conference, Workshop on Semantic Web, 2003.

[19] P. Castells, B. Foncillas, R. Lara, M. Rico, and J. L. Alonso, "Semantic Web Technologies for Economic and Financial Information Management," The Semantic Web: Research and Applications - 1st European Semantic Web Symposium (ESWS 2004), Vol. 3053, pp. 473-487, 2004.

[20] P. Castells, F. Perdrix, E. Pulido, M. Rico, V. R. Benjamins, J. Contreras, et al., "Neptuno: Semantic Web Technologies for a Digital Newspaper Archive,". The Semantic Web: Research and Applications - 1st European Semantic Web Symposium (ESWS 2004). Lecture Notes in Computer Science, Vol. 3053, pp. 445-458, 2004.

[21] J. Contreras, V. R. Benjamins, M. Blázquez, S. Losada, R. Salla, et al, "A Semantic Portal for the International Affairs Sector,". Engineering Knowledge in the Age of the Semantic Web - 14th Intl. Conference on Knowledge Engineering and Knowledge Management, Vol. 3257, pp. 203-215, 2004.

[22] R. Neches, R. Fikes, T. Finin, T. Gruber, R. Patil, T. Senator, et al. "Enabling technology for knowledge sharing," AI Magazine, vol. 12, no. 3, pp. 13-56, 1991.

[23] TR. Gruber, "A translation approach to portable ontology specifications," Knowledge Acquisition, vol. 5, no. 3, pp.199-220, 1993.

[24] Studer R., Benjamins VR., and Fensel D., "Knowledge Enineering, Principles and Methods," Data and Knowledge Engineering, 25(1-2), pp. 161-197, 1998.

[25] D. McGuinness, "Ontologies and online commerce," IEEE Intelligent Systems, vol. 16, no. 1, pp. 8-14, 2001.

[26] UNSPSC, http://www.unspsc.org/.

[27] eCl@ss, http://www.eclassonline.com/.

[28] N. F. Noy and D. L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," http://protege.stanford.edu /publications/ontology_development/ontology101-noy-mcguinness.html, 2001.

[29] P. Le Hégaret, DOM Activity Lead. Document Object Model (DOM). http://www.w3.org/DOM/.

[30] G. Salton and M. McGill, "Introduction to Modern Information Retrieval," McGraw-Hill, 1983.

[31] S. Dumais, "Improving the retrieval of information from external sources," Behavior Research Methods, Instruments, and Computers, vol. 23, no.2, pp. 229–236, 1991.

[32] E. Amitay, D. Carmel, A. Darlow, R. Lempel, and A. Soffer, "Topic distillation with knowledge agents," Proceedings of Text Retrieval Conference, 2002.

[33] Hui-Chuan Chu, Ming-Yen Chen, and Yuh-Min Chen, "A semantic-based approach to content abstraction and annotation for content management," Expert Systems with Applications, vol. 36, no. 2, pp. 2360-2376, 2009.

[34] J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, K. Wilkinson, "The Jena Semantic Web Platform: Architecture and design," Technical report, Hewlett Packard Laboratories, 2003.

[35] V. Haarslev, R. Moller. "The Racer user's guide and reference manual," 2004.

[36] Lucene, http://lucene.apache.org.

[37] ICTCLAS, http://ictclas.org/index.html

Liyi Zhang received the B.S. and Ph.D. degrees from Wuhan University, Wuhan, China, in 1988 and 1999, respectively.

He is currently a professor and DEAN OF DEPARTMENT of Information & E-commerce in School of Information Management, Wuhan University, Wuhan, China. He has published five books, over 40 Journal papers. In addition, he has organized several conferences in the emerging areas of Electronic Commerce. His research interests include information system, e-commerce and information retrieval.

Mr. Zhang is a member of E-commerce Major Guiding Committee of China, the Secretary-general of Association of Hubei Electronic Commerce, and a member of AIS(Association of Information System).

Mingzhu Zhu received the B.S. degree from Wuhan University, Wuhan, China, in 2008.

He is currently a Master Degree Candidate of electronic commerce, Wuhan University, Wuhan, China. His research interests include information system, e-commerce and information retrieval.

Wei Huang received the B.S. and Master degrees from Hubei University of Technology, Wuhan, China, in 2002 and 2005, respectively.

He has worked at Hubei University of Technology since 2005. And He is currently a Ph.D. candidate of electronic commerce, Wuhan University, Wuhan, China. His research interests include information system, e-commerce and information retrieval.

Mr. Huang is a member of Association of Hubei Electronic Commerce.