SOSEM: A Self-recommending Ontology Search Extension Mechanism Based on User Behavior

Jianghua Li[1, 2]
1 National Center for Materials Service Safety, University of Science and Technology Beijing, Beijing, China
2 School of Information and Engineering, Jiangxi University of Science and Technology, Ganzhou, Jiangxi, China
Email: jxsimil@hotmail.com

Peng Shi*[1] and Changjun Hu[3]
3 School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, China
Email: pshi@ustb.edu.cn, huchangjun@ies.ustb.edu.cn

Abstract—**Ontology search mechanism plays an important role in the discovery and usage of ontology. The current ontology search engines all adopt keyword-based search mechanism. It is difficult to clearly represent users' search requirements because the number of the keywords input by a user is usually few. Extending the input keyword(s) by using WordNet will increase the recall rates and decrease the accuracy rates, because it enlarges the search scope imprecisely. To improve the performance of ontology searching, a self-recommending ontology search extension mechanism is proposed based on the study of the ontology search requirements and user search behaviors in this paper. The experimental results show that the proposed approach can effectively improve the matched coverage between query and target ontologies. Our method can also help user to conveniently search desired ontologies with high precision rates. Ontology search engines will get benefit from our method, because it will obviously improve the customer's experience during ontology searching.**

Index Terms—**ontology search, extension mechanism, self-recommending, user behavior**

I. Introduction

Ontology is a formal explicit specification of shared conceptualization [1]. It describes the domain concepts and relationships among them. Using ontology to describe the domain resources can enhance machine-processable information from the content level to the semantic level, as a result, the management and usage of the resources will become more intelligent and efficient. Since ontology was put forward, it has been widely and successfully used in knowledge sharing, software reuse, digital library, information retrieval, data integration and other fields. Ontology has been proved to be a pillar for domain knowledge description and the Semantic Web [2].

More and more applications need ontology to realize knowledge sharing and reusing to improve the abilities of system communication, interoperability and automatic processing. The traditional artificial construction of ontology is a time-consuming labor process. At the same time, more and more ontologies have been published on the Internet. Obviously, it is an efficient solution to search suitable ontology from the Web and reuse it by extending, refining and pruning according to specific application demands. For this purpose, several ontology libraries [3-6] and ontology search engines [7-15] have been developed by some researchers and organizations, as well as some prototype systems [16-19]. Nevertheless, ontology search is encountering the same problem as Web document search, the low search precision ratio. Literature [12] showed that the search precision ratio was only about 30%, even in the cases where the recall rates reached at 100%. Therefore, it is still a problem to be studied that what search mechanism should be adopted to efficiently and accurately acquire ontologies meeting users' demands from the ontology library or ontology search engine.

The main contributions of this paper can be summarized as follows: 1) we clearly analyze the requirements and characteristics of ontology search, as well as the reasons causing low precision ratio in ontology search. 2) We propose a self-recommending ontology search extension mechanism (SOSEM) based on user behavior and conduct an experiment to illustrate the high efficiency and precision ratio of our system.

The rest of the paper is organized as follows. Section 2 is dedicated to the related work of ontology search. The demands and characteristics of ontology search, as well as the challenges, are described and analyzed in Section 3. Section 4 discusses user behaviors and characteristics in ontology search, and elaborates the proposed ontology search mechanism. The experiments and evaluation are presented in Section 5. Finally, Section 6 gives some conclusions and future work.

II. Related Work

At present, the frequently-used ontology libraries include DAML library [3], Ontolingua [4], the Protégé OWL library [5], Schema Web [6] and et al. Their main objectives are ontology construction, maintenance, reuse and application. Ontology libraries seldom support query services, such as DAML library, which summarizes stored ontologies according to URI, submission date, keyword, class, property, namespace, and funding source and so on. A query based on keyword can be posed over the library by inputting a name of a class or a property.

doi:10.4304/jsw.8.7.1770-1777

Protégé OWL library just offers a list of ontologies, including the name of ontology, a simple summary about it and source. It provides user an ontology lookup service only by browsing them or read the specifications.

WATSON [13], Falcons [14], AKTiveRank [15], and other search methods in literature [18] [24] all directly run a keyword-based query for ontologies with the keywords input by user, no extension involved.

OLS [27-29] (Ontology Lookup Service) provides an interface to query, browse and navigate biomedical ontologies and controlled vocabularies in a keyword-based way. Different from the above keyword-based search, in OLS, when a user input a search keyword, A collection of close matches as possible suggestions are sent back to the user, which are displayed in a drop-down menu akin to the interface in Google. Users can search all ontologies containing the selected term.

Swoogle [7] is one of the most prominent ontology search engines with the services for ontology search and metadata search, as well as ranking. Its graphic user interface is similar to that of Google, through which a keyword-based query can be executed. To improve the precision ratio of ontology search, Swoogle provides an advanced interface for senior users, which essentially allows them to extend input keywords with some constraints such as URL (Unified Resource Location), content, language and coded format and so on. The ontology search extension method by adding constraints to keywords is also adopted by OntoSearch [9] and ONTOSEARCH2 [16]. OntoSearch applies the Google Web API to search ontology files after extension by adding file type constraints to keywords input by user. For example, when a user input a search word "*keyword*" , OntoSearch invokes Google Web API to run a query with the phrase "*filetype:RDF keyword*" as input for ontologies containing the "*keyword* " with the file type " *rdf*". ONTOSEARCH2 extends keyword-based search by adding specific type to search keywords. For example, when a user input "*keyword pragma: Class"* for ontology search, it means that an ontology meeting the user need should be defined a class named "*keyword*", other input form such as "*keyword pragma: Property*", "*keyword pragma: Source*" can be processed in the same manner.

OntoKhoj [8], OntoQA [11], and other search methods in literature [21] [22] [23] all use WordNet to extend each query keyword with their synonyms, hypernyms and hyponyms. The researchers think this will add matched terms between query phase and targeted ontologies. OntoKhoj uses the keyword input by user to run a query at first. If there is no exact match between the search keyword and any concepts in the ontologies, the synonym matching is performed. If no matched ontology returns, then the hyponymy matching is performed. OntoKhoj traverses the hyponymy link upward until a term which is close to the keyword is found. OntoQA applies the Swoogle Web API to search for ontology files after extension is completed.

In OntoSelect [12], ontologies can be searched by keyword or by document. In keyword-based search only the keyword(s) provided by the user will be used for the search. In document-based search the user can provide either a URL for a web document that represents a specific topic or the user simply provides a keyword as the topic. According to the topic, OntoSelect then automatically links to a corresponding Wikipedia page, from which a linguistically or statistically derived set of the most relevant keywords will be extracted and used for ontology search as search keywords. The similar extension method is that proposed by Jones [26].

III. Characteristics and Challenges of Ontology Search

From the analysis above, the common features of the existing ontology search mechanisms are based on keyword matching. Although ontology search is similar to traditional Web document search in some degree, compared with normal Web document, ontology has good concept hierarchies and abundant semantics. Therefore, traditional Web document retrieval techniques cannot be completely applied to ontology search. The reality is that the precision ratio is all very low if no extension before the query is submitted, whether the traditional Web document search, such as Google, or ontology search, such as Swoogle, the returned result set usually contains many documents which are unrelated to user requirement. Some researchers want to submit a query to search engine after it is extended in some way according to the keyword(s) input by user so as to improve the precision ratio of ontology search. Although Swoogle and some tools adopt several constraints to extend their queries, the constraints only work in syntax level, such as language constraint, specific element type constraint (class or property) and so on. The syntax level constraints cannot add semantic information any more about what a user wants on earth or express it clearly. Both OntoFetcher and SQORE try to construct some triples meeting the RDF grammar as query strings. This method demands users not only to grasp ontology languages and their grammar, but also to be familiar with some domain knowledge. Apparently, it is not suitable to the common users. We should also see that the lexicon WordNet is widely adopted by some ontology search tools to extend their keywords submitted by users with synonyms, hypernyms and hyponyms, which will add some semantic information about user search requirement in a certain degree. The main features lie in two aspects: on the one hand, there is a certain relationship between a hypernym and a hyponym, on the other hand, the keyword input by a user is not a domain term and dose not match any concepts in target ontologies, but its synonym dose. However, this ontology search extension mechanism works at the cost of broadening the searching scope, and it is a blind extension way without the specific application requirement. Meantime, it is not easy to determine extended hierarchies in WordNet because of the less of suitable extension principles. This will lead to many keywords unrelated to user requirement in a query, so what can be foreseen is that there are a lot of unrelated ontologies in search results and the precision ratio is low. Furthermore, WordNet is a limited knowledge base, it is impossible to solve all extension problems in every domain. When a keyword submitted by a user cannot be found in WordNet, the search engine may not work if a search engine depends on it too much.

Ontology is mainly composed of classes, properties and relationships among them. Its structure is usually considered as graph-like structure, in which vertexes represent the classes and edges represent the relationships. When a query demand is submitted to search engine for ontologies, there are two situations of returned results according to the user's expectation. One is that there are matches as much as possible between the input keywords and the concepts and properties in the candidate ontologies. The other is that these ontologies should have been defined complete relationships between classes as far as possible. And the latter is more important.

Suppose that Q(C, R, P) is a user query in which C is a set of concepts, R is a set of relationship operators and P is a set of relationships that C is defined on R, and O is a set of target ontologies. According to the structural characteristics of ontology, P can also be seen as a set of graph fragments. Ideally, an exact candidate ontologies set should be searched if ontology search is taken as a subgraph-matched process between P and that of target ontology. Unfortunately, there are often many ontologies constructed by different users in one domain, and the users maybe use different words to define one same domain concept, such as synonym or homoionym. That means one concept has several different names in different ontologies belonging to one domain. The similar phenomenon also occurs on properties and relationships. Even if there are some high-quality ontologies constructed by domain experts and knowledge engineers only using the domain concepts, the problems are that, all the users are not domain experts, so maybe the keywords submitted by them are not domain concepts, but synonyms or homoionym of the concepts. In addition, even if we assume the two keywords submitted by user are both the domain concepts, there are still some uncertainties in the search process. The problem lies in that, maybe the relationship between the two concepts is defined in some ontologies, maybe not in other ontologies, and maybe defined using different terms. This will lead to little matches between a query and the target ontologies. So it is difficult to construct a query like the form Q(C, R, P) to express user search requirement. Another challenge is that, the number of the keywords submitted by a user is often too small to distinguish boundaries of different domains. This also leads to a low precision ratio of ontology search because of the overlap concepts among domains.

IV. SOSEM Design

A. *Motivation and User Behavior Analysis for Ontology Search*

Following on from the above, the reason for low precision ratio of ontology search essentially lies in that the user interface provided by ontology search tool cannot express user's query requirement exactly, as a result, the matches between user query and the target ontologies are very low. The case is applied to both the way of extension mechanism using WordNet where many unrelated keywords are contained in a query and the way of no extension where the number of keywords in a query is too small. We think a feasible method to improve the situation is: a query should contain keywords as more as possible and there should be more matches between those keywords and the target ontologies. Our motivation derives from user behavior and characteristic analysis for ontology search.

From the viewpoint of user search behavior, on one hand, they just input their familiar keywords according to their different knowledge background, concerned aspects and interests about the application domain. Moreover, because of neglection or simple forgetting, only a small number of keywords are input to search ontologies by a user, usually one or two, it is not enough to clearly express his/her search requirement. On the other hand, all the keywords submitted by different users are not same even if those queries are executed to search ontologies with one same application domain. What should be noticeable is that the input keywords belong to one same domain, although they are different. We can fully leverage it to reduce the disadvantages caused by a single user who just input one or two keywords to search ontologies. Combining all the keywords input by different users together will increase the number of the matched search keywords and improve the performance to distinguish domain, and enhance semantic expression for user ontology search requirement. Therefore, it is applicable to employ different keywords from different users for one same domain ontology search to realize self-recommending extension in order to improve the precision ratio of ontology search. For the convenience of description, three terms are defined as follows at first.

Definition 1 Given a keyword set *kwset* and a keyword *kw*, if $kw \in kwset$, then $\forall kw_i \in kwset \wedge kw_i \neq kw$ is a relevant keyword of *kw*, and the set *kwset* is a relevant keyword set of *kw*.

Definition 2 Given a keyword set *kwset*, and $1 \leq |kwset| \leq n$, $|kwset|$ denotes the number of the keywords in *kwset*. If using a random non-empty subset of *kwset* as input keywords are submitted to search engine for ontology search, it always return the ontology O_i , then *kwset* is the additional index of O_i.

Definition 3 Given the keyword set *kwset* as input keywords submitted to search engine for ontology search, and $1 \leq |kwset| \leq n$, $|kwset|$ denotes the number of the keywords in *kwset*, and *O* denotes the set of the returned candidate ontologies. Given an ontology $O \in O_i$, and $kw_1, kw_2, \dots, kw_m\,(m \leq \mathrm{n}) \in kwset$, if O_i is returned as a candidate just because of $kw_1, kw_2, \dots, kw_m \in O_i$ in the search process, then $kw_1, kw_2, \dots, kw_m$ are called the seeds of O_i.

B. *SOSEM architecture*

The global architecture of the proposed system is shown in Fig.1. SOSEM is composed of four main components: "Ajax Interface," "Recommender," "Indexer," and "Ontology Repository". There are two stages in the process. The first one is called pre-process stage, in which every ontology in the ontology repository is inserted into the table additionalindex, including two attributes, additional index and ontologyURL. The column additional index is be used to store the seeds of the ontology pointed by the value of the attribute ontologyURL, its initial value is NULL. The second stage is the process of searching stage. The core principle of the algorithm is that, if a keyword input by a user has a match with a concept or property of an ontology in the repository, this keyword is then inserted into the column additional index in the table additionalindex as a seed of the ontology. After a certain amount of search, the column additional index of the ontology records the most frequent keywords which would usually be used by users as input keywords to search the ontology. SOSEM uses the additional index as candidate terms to help a user to extend the submitted keyword(s) in a self-recommending way. In a nutshell, the process of the algorithm is illustrated as Fig .2. The system works as follows.

(1) A user inputs his/her familiar or interested keyword(s).

(2) SOSEM presents the relevant keyword sets of the keyword(s) input by the user by querying the additional index. If the additional index is NULL, then go to (4).

(3) User selects one or more keywords from the relevant keyword sets which he is interesting with or are potential domain terms neglected or forgotten by him because of his knowledge background.

(4) SOSEM submits the keyword set after extension as a query to search engine for ontology search. When there is more than one element in the query, the ontologies meeting requirement should match more than a half of the elements in the query.

(5) SOSEM checks every ontology returned by the search engine. If a keyword in the query is a seed of certain ontology and the keyword is not in the additional index of the ontology, inserts it into the additional index.

Example: Suppose that a user gets the candidate ontologies O_1, O_2 and O_3 by using "kw_1, kw_2" as the search keywords, in which "kw_1" as a seed of O_1, "kw_1" and "kw_2" as the seeds of O_2, and "kw_2" as a seed of O_3 are inserted into their additional indexes respectively. Another user gets the candidate ontologies O_2, O_3 and O_4 by using "kw_3, kw_4" as the search keywords, in which "kw_3" as a seed of O_2, "kw_3" and "kw_4" as the seeds of O_3, and "kw_4" as a seed of O_4 are inserted into their additional indexes respectively. SOSEM will present "kw_2" and "kw_4" with checkboxes for the user to choose and extend his search keywords when the third user submits the keywords "kw_1, kw_3", because the word "kw_2" is in the additional index of the ontology O_2 and is a relevant keyword of the word "kw_1". In the same way, the word "kw_4" is a relevant keyword of the word "kw_3". Actually in our definitions, the relevant keywords of a keyword are all seeds of some ontologies and come from their additional indexes, and one of its relevant keyword sets is an additional index of some an ontology.

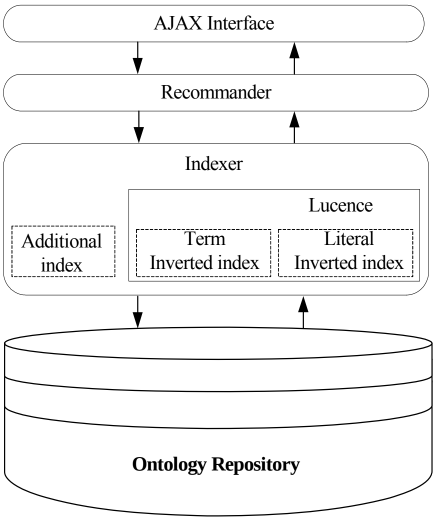


Figure 1. SOSEM architecture

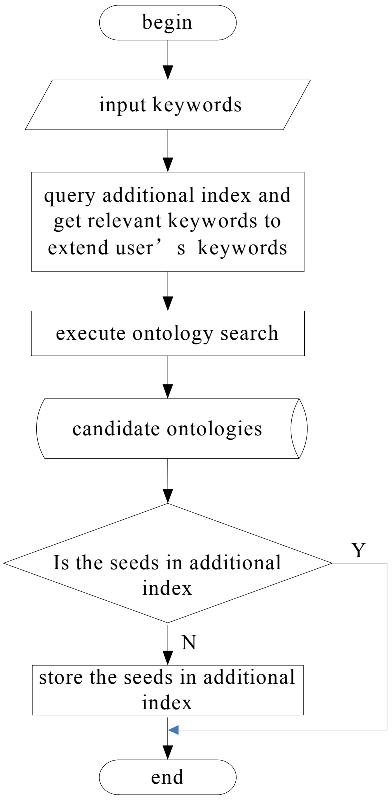

begin
input keywords
query additional index and get relevant keywords to extend user’s keywords
execute ontology search
candidate ontologies
Is the seeds in additional index
Y
N
store the seeds in additional index
end

Figure 2. The process of the algorithm of SOSEM

The component "Ajax interface" constitutes the input text interface for user. Through this interface, a user can input keyword(s) for ontology search, the system acquires the candidate terms generated by "Recommender" and the candidate ontologies is presented to user. The main objective of the component "Recommender" is to determine which words can be candidate terms by querying the table additionalindex, the candidate terms are then presented to the user in a list of checkboxes (see Fig.3) .End user can navigate the checkboxes and select the desired proposed options. In this way, SOSEM guides the user to complete keywords extension for ontology search. To facilitate ontology search, the system analyzes ontology and extracts terms defined in it, including concepts, properties and literals. They are used to match the keywords input by a user and consistent with user behavior characteristics on searching. Traditional information retrieval techniques look a document as a collection of tokens, typically words or N-Grams. In SOSEM, we treat ontology as a bag of structural terms, so the component "Indexer" is responsible for indexing and retrieving ontologies using the technique of inverted index with the extracted terms. The component "Ontology Repository" is used to store ontologies persistently implemented using Jena Semantic Web Framework [30] over MySQL database.

It can be seen from the above process of the algorithm, SOSEM has three advantages. First, it realizes the ontology search extension mechanism by using different keywords input by different users for one same domain ontology search so that the extension is more pertinent, not random and dependent on a knowledge base any more. Second, since the keywords in the additional index are all the words having matched some ontology in the repository, the extended query will have more matches with the target ontologies. So the precision ratio will be improved. Third, the whole process is a cyclic process of self-learning. Therefore with the growing of usage, the additional indexes of the ontologies in the repository will keep being filled with the relevant keywords. The extension pertinence and the search precision ratio will improve tremendously. The function of the additional index in the search process is only to offer users the candidate keywords for self-extension, and ontology search is still conducted according to ontology metadata indexes.

V. Evaluation

In order to gain the ontology dataset for the experiment, we developed a Swoogle clawer to search and download ontologies by invoking Swoogle Web API. The clawer searched ontologies using the words in the first row of Table 1 as the input keywords. After the elimination of invalid and repetitive URL, 268 ontologies including RDF and OWL files are obtained. The inverted index is created and saved in MySQL database after parsing the metadata by Jena. SOSEM prototype system is developed over the ontology repository. The experimental environment includes: dual-core Intel T6570 CPU 2.1GHz, 2GB RAM, 160GB hardware, Window XP system, apache-tomcat 5.5.27，jdk1.6.0_21.

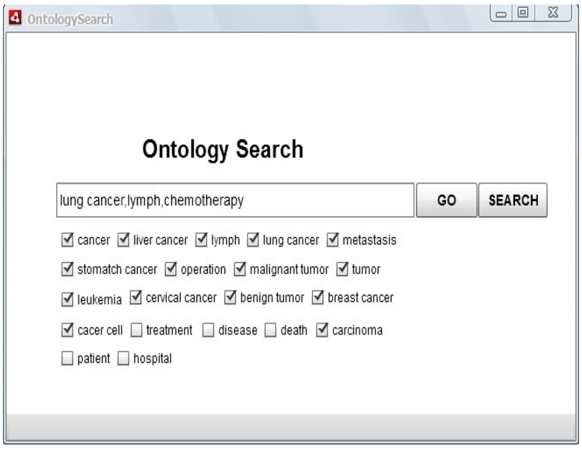
Two experiments were conducted to evaluate the performance of SOSEM. 1) 20 student volunteers were randomly selected to search the ontologies about cancer by inputting keyword(s) to the user interface of our system according to their own understanding and knowledge background one by one to test the precision ratio and recall rates of SOESM. As shown in formula (1) and (2), PC refers to the precision ratio, TP denotes the number of the relevant ontologies returned in the searched results, FP denotes the number of the irrelevant ontologies returned in the searched results, RC refers to the recall rates, and FN denotes the number of the relevant ontologies but not returned in the searched results. 2) The method proposed in [26] is be used to search ontologies in our ontology repository, which extend keyword(s) by inquiring WordNet with their synonyms, hypernyms and hyponyms. Then the precision ratio and recall rates of the two ontology search extension methods are compared.

$$PC = TP / (TP + FP) \quad (1)$$

$$RC = TP / (TP + FN) \quad (2)$$

TABLE I.
TYPE THE SEARCH KEYWORDS IN THE EXPERIMENTS

Instruction	The set of the keywords
The keywords used to search ontologies by invoking Swoogle Web APIs	Medicine, treatment, health, care, doctor, blood, disease, pain, sport, hospital, surgery, department, death, chemotherapy, operation, metastasis, lung cancer, lymph, breast, stomach, cancer, culture, education, hygiene
The keywords input by the 20 volunteers	medicine, cancer, chemotherapy, liver cancer, lymph, lung cancer, death, stomach cancer, breast cancer, metastasis, patient, doctor, operation, malignant tumor, leukemia, cervical cancer, benign tumor, tumor, cancer cell, hospital, treatment, disease, carcinoma
The search keywords after extension by using WordNet[26]	cancer, cell, tumor, patient, document, carcinoma, lymphoma, disease, access, treatment, skin, liver, leukemia, risk, breast, genetic, tobacco, thymoma, malignant, gene, clinical, neoplasm, pancreatic, Tissue, therapy, lesion, blood, study, thyroid, smoking, polyp, human, health, exposure, studies, ovarian, information, research, drug, related, associated, neoplastic, oral, bone, chemotherapy, body, oncology, growth, medical, lung



Ontology Search

lung cancer,lymph,chemotherapy

☑ cancer ☑ liver cancer ☑ lymph ☑ lung cancer ☑ metastasis
☑ stomatch cancer ☑ operation ☑ malignant tumor ☑ tumor
☑ leukemia ☑ cervical cancer ☑ benign tumor ☑ breast cancer
☑ cacer cell □ treatment □ disease □ death ☑ carcinoma
□ patient □ hospital

Figure 3. The user interface of SOSEM

As shown in the Table 1, the words gathered in the second row are the keywords submitted by all the 20 volunteers and those in the third row are the keywords as a query after extension by inquiring WordNet. Table 2 lists the precision ratio and recall rates of the two ontology search extension methods, in which Hi refers to the ith user and WN refers to the ontology search extension method using WordNet. Fig.3 shows the user interface of SOSEM when the 20th user input his keywords. The Figure indicates that when a user inputs his keyword(s), the relevant keywords with checkboxes are send back to UI by the system so that the user can select pertinently the presented keywords to extend his inputs as a new query. The curve in Fig.4 reveals the change characters of the amount of the returned ontologies with that of the search users. The change characters of the precision ratio and recall rates with the amount of the search users is shown in Fig.5 with two curves.

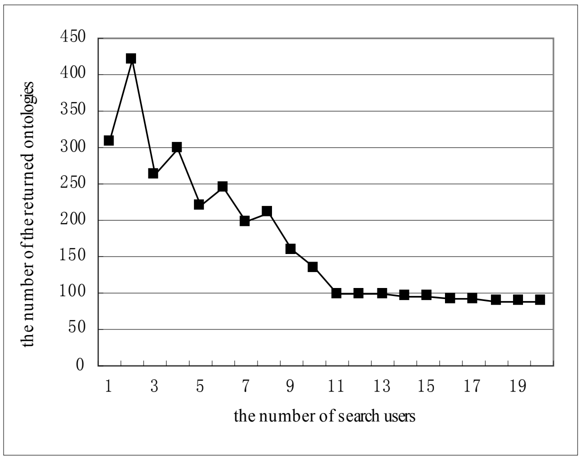


Figure 4. The change curve of the amount of the returned ontologies with that of the search users

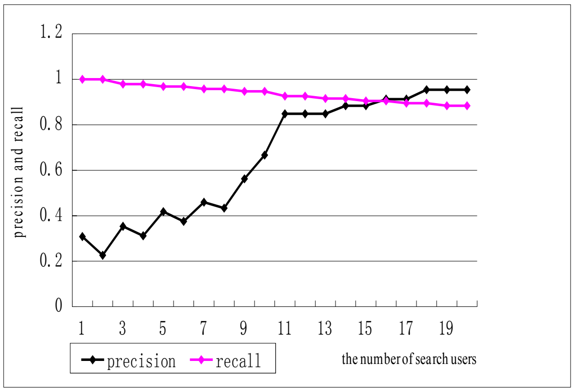


Figure 5. The change curves of the precision and recall with the amount of the search users

It can be seen from Fig.4, with the increase of the number of users, the number of returned ontologies decreases. It indicates that the number of irrelevant ontologies reduces with the increase of the matches between the extended keywords and the target ontologies. It is shown in Fig.5 that SOESM has high and relevantly stable precision ratio and recall rates when the number of the search users reaches a certain amount. In addition, with the number increase of users, the precision ratio rises gradually and becomes stable while the recall rates drops a little and becomes stable because with the increase of the search keywords, the number of the matched concepts in the ontologies meeting search requirement are increased by the system, which may lead to some ontologies that has low matched number are filtered out. According to Table 2, WN has a low precision ratio while the recall rates even reaches 100%. The reason is that, the extended query using WordNet contains too many keywords that are relevant to the user's input at the syntax level but irrelevant to the domain at the semantic level. Therefore, the search scope is enlarged and the returned search results include a large amount of irrelevant ontologies.

TABLE II.
THE PRECISION AND RECALL RATES OF THE TWO EXTENSION METHODS

	Precision	Recall
H11	0.8485	0.9263
H12	0.8484	0.9263
H13	0.8485	0.9158
H14	0.8842	0.9158
H15	0.8842	0.9053
H16	0.9130	0.9053
H17	0.9130	0.8947
H18	0.9545	0.8947
H19	0.9545	0.8842
H20	0.9545	0.8842
WN	0.3755	1

VI. CONCLUSION AND FUTURE WORK

Ontology search mechanism has a remarkable meaning for the discovery and usage of ontology. The current ontology search engines are all keyword-based search. By using WordNet to extend the keyword(s) increases the recall rates and decreases the accuracy rates meantime, because it enlarges the search scope imprecisely. Based on the study of the ontology search requirements and user search behaviors, this paper proposes the Self-recommending Ontology Search Extension Mechanism, which uses the keywords with domain commonness input by different users to extend the query submitted by a user. Domain commonness avoids the randomness of the extension and ensures the pertinence of it. Meanwhile, the matched concepts increases since the words used to extend are all seeds which have matched some ontologies in the repository. The experimental results prove that with the increase of the number of the search users, SOESM obtains high and relatively stable precision ratio and recall rates and it is practical in ontology search.

The returned results from SOSEM are also a list of several ontologies if there is more than one ontology in one same domain. How to help a user or application to choose the most suitable ontology is our future work. Ontology evaluation model can assess the quality and importance of ontology. The ranking result of the evaluated ontology provides practicable reference for user and application to efficiently choose suitable ontology. The known factors influencing the quality and importance of ontology include ontology's linkages with others, contents, concept structures, schema, instances and so on. The existing ontology evaluation models only evaluate ontology according to single factor and produce unsatisfied results. We think the factors influencing the importance of ontology and their influence principles deserve further investigation, an ontology evaluation model will be proposed and integrated with SOSEM to rank the returned ontologies.

ACKNOWLEDGMENT

The work in this paper is supported by National Program on Key Basic Research Project (973 Program) under Grant No.2013CB329606 and the Key Science-Technology Plan of the National 'Twelfth Five-Year-Plan' of China under Grant No.2011BAK08B04. And the work is also partly supported by the Fundamental Research Funds for the Central Universities under Grant No.FRF-TP-12-162A, the Scientific Research Found of Jiangxi Provincial Education Department of China under Grant No.GJJ13411, and the Jiangxi Provincial Ministry of Education Research Fund under Grant No.12770.

REFERENCES

[1] Gruber T R. Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, Vol. 5, No. 2, pp. 199-220, 1993.

[2] Michael Gruninger, Lee. Jintae. Ontology applications and design, Communications of the ACM, Vol. 45, No. 2 , pp. 39-41, 2002.

[3] Dmal. DAML Ontology Library, 2004, http://www.daml.org/ontologies.

[4] Ontolingua. Ontolingua ontology library, 2005, http://Ontolingua.stanford.edu.

[5] Protégé. Protégé ontology library, 2010, http://protege.cim3.net/cgi-bin/wiki.pl? ProtegeOntologiesLibrary.

[6] Schema Web. Schema web library, 2008, http://www.schemaweb.info.

[7] L Ding, R Pan, T Finin. Finding and Ranking Knowledge on the Semantic Web, Proceedings of the 4th International Semantic Web Conference, pp. 156-170, 2005.

[8] C Patel, K Supekar, Y Lee ,E Park. Ontokhoj: A semantic web portal for ontology searching, ranking, and classification, Proceedings of 5th ACM Workshop on Web Information and Data Management, pp. 58–61, 2003.

[9] Y Zhang, W Vasconcelos, D Sleeman. Ontosearch: An ontology search engine, Proceedings of 24th SGAI on Innovative Techniques and Applications of Artificial Intelligence, pp. 145-156, 2004.

[10] Syed Adman Hussain Shah, Adnan Khalid, Muhammad Abdul Qadir. OntoFecher: An Approach for Query Generation to Gather Ontologies and Ranking them by Ensuring User's Context , Proceedings of the 4th International Conference on Emerging Technologies, pp. 247-252, 2008.

[11] Samir Tartir, I Budak Arpinar, Miehael Moore. OntoQA: Metric-Based Ontology Quality Analysis, Proceedings of IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources, pp. 45-53, 2005.

[12] Buitelaar Paul, Eigner Thomas. Evaluating ontology search, Proceedings of the 5th International Workshop on Evaluation of Ontologies and Ontology-Based Tools, pp. 11-20, 2007.

[13] Mathieu d'Aquin, Marta Sabou, Martin Dzbor, Claudio Baldassarre, Laurian Gridinoc, Sofia Angeletou. WATSON: A Gateway for the Semantic Web, Poster Session of the 4th European Semantic Web Conference, 2007.

[14] Qu Yuzhong, Cheng Gong. Falcons concept search: A practical search engine for web ontologies, IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans, Vol. 27, No. 6, pp. 810-816, 2011.

[15] Alani Harith, Brewster Christopher, Shadbolt Nigel. Ranking ontologies with AKTiveRank, Proceedings of the 5th International Semantic Web Conference, pp. 1-15, 2006.

[16] J Z Pan, E Thomas, D Sleeman. ONTOSEARCH2: Searching and Querying Web Ontologies, Proceedings of the IADIS International Conference, pp. 42-49, 2006.

[17] R.Ungrangsi, C.Anutariya, V.Wuwongse. SQORE: An ontology retrieval framework for the next generation Web, Concurrency Computation Practice and Experience, Vol. 21, No. 5, pp. 651-671, 2009.

[18] Wei Yu, Junpeng Chen, Jiaheng CAO. A Novel Approach for Retrieval and Ranking for the Ontologies on the Semantic Web, Journal of Chinese Computer Systems, Vol. 28, No. 6, pp. 1044-1048, 2007.

[19] Ahmed Tolba1, Nabila Eladawi, Mohammed Elmogy. An enhanced indexing and ranking technique on the semantic web, International Journal of Computer Science Issues, Vol. 8, No. 5, pp. 118-125, 2011.

[20] Harith Alani, Christopher Brewster. Metrics for Ranking Ontologies, Proceedings of the 5th International Semantic Web Conference, pp. 1- 5, 2006.

[21] Sabou Marta, Lopez Vanessa, Motta Enrico. Ontology selection for the real semantic web How to cover the Queen's birthday dinner, Proceedings of the 15th International Conference, pp. 96-111, 2006.

[22] Hwang Myunggwon, Kong Hyunjang, Kim Pankoo. The design of the ontology retrieval system on the web, Proceedings of the 8th International Conference Advanced Communication Technology, pp. 1815-1818, 2006.

[23] Marcos Martínez-Romero1, José M. Vázquez-Naya, Cristian R. Munteanu, et al. An Approach for the Automatic Recommendation of Ontologies Using Collaborative Knowledge, Proceedings of 14th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, pp. 74-81, 2010.

[24] Oren Eyal, Delbru Renaud, Catasta Michele, et al. Sindice.com: A document-oriented lookup index for open linked data, International Journal of Metadata, Semantics and Ontologies, No. 3, pp. 37-52, 2008.

[25] Wuwongse V, Anutariya C, Akama K, Nantajeewarawat E. XML declarative description (XDD): A language for the semantic web, IEEE Intelligent Systems, Vol. 16, No. 3, pp. 54–65, 2001.

[26] Jones M, Alani H. Content-based Ontology Ranking, Proceedings of the 9th International Protégé Conference. California, pp. 96-99, 2006.

[27] Richard Côté, Florian Reisinger, Lennart Martens, et al. The Ontology Lookup Service bigger and better, Nucleic Acids Research, No.38, pp. 155-160, 2010.

[28] Harald Barsnes, Richard G. Côté, Ingvar Eidhammer, et al. OLS Dialog: An open-source front end to the Ontology Lookup Service, 2010, http://www.biomedcentral.com/content/pdf/1471-2105-11-34.pdf.

[29] Richard G. Côté, Philip Jones, Lennart Martens, et al. The Ontology Lookup Service more data and better tools for controlled vocabulary queries, Nucleic Acids Research, Vol. 36, No. 8, pp. W372-W376, 2008.

[30] B. McBride. Jena: A semantic web toolkit, IEEE Internet Computing, Vol. 6, No. 6, pp. 55-59, 2002.

Jianghua Li was born in Henan Province, China, in 1976. He received the BS degree and MS degree in computer science from Jiangxi University of Science and Technology in 2000 and 2005 respectively. From 2010 to date, he works as a Ph.D. candidate in the National Center for Materials Service Safety, University of Science and Technology Beijing, China, and an associate professor in the School of Information and Engineering, Jiangxi University of Science and Technology, China. His research interests include Semantic Web, domain data engineering, software engineering and digital watermark.

Peng Shi is an associate professor in the National Center for Materials Service Safety, University of Science and Technology Beijing, China. He received his Ph.D. in the Institute of Computing Technology, Chinese Academy of Science in 2007. His research interests include data and knowledge engineering, cloud computing and Semantic Web.

Changjun Hu is a professor in the School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, China. He received his Ph.D. in the Department of Computer Science, Beijing University in 2001. His research interests include parallel computing and parallel compiler technology, grid computing, cloud computing and service computing, data engineering and software engineering.