Design and Implementation of Ontology-Based Query Expansion for Information Retrieval

Fang Wu[1, 2], Guoshi Wu[1] and Xiangling Fu[1]

[1] School of Software Engineering, Beijing University of Posts and Telecommunications,
Beijing 100879, P.R. China w-fang@hotmail.com xiangling.fu@263.net
[2] Department of Computer Science, Cangzhou Medical College, Cangzhou 061001, Hebei,
P.R. China wuguoshi@email.buptsse.cn

Abstract. In Information Retrieval (IR), the user's input query conditions usually are not detailed enough, so the satisfactory query results can not be brought back. Query expansion of IR can help to solve this problem. However, the common query expansion in IR cannot get steady retrieval results. In this paper, we propose and implement query expansion method which combines domain ontology with the frequent of terms. Ontology is used to describe domain knowledge; logic reasoner and the frequency of terms are used to choose fitting expansion words. By this way, higher recall and precise can be gotten as user' query results. Experimental results show that compared with the results of common query expansion, the method described in this paper can get statistically significant improvement in recall and precise combination.

Keywords: *Search engine, Ontology, Web ontology language (OWL), Knowledge management, Enterprise search*

l. INTRODCTION

In information retrieval (IR), even the best system has a limited recall. Users may miss many important documents which they really need usually. There are two fundamental reasons for this problem. The first one is word mismatch, which means that concepts (or key words) of user queries are often different from the words of the resource documents although these words have similar meanings. Another is that users submit short queries which are not detailed enough for IR, so the bad search performance ensues. Query expansion (QE) can effectively alleviate the problem by adding additional terms which have similar meaning to the original query.

In this study, we proposed a new expansion method which is based on domain ontology and frequency of keyword occurrence in resource documents to filter expansion words. It achieves better performance in both precision and recall.

2. RELATED RESEARCH

QE approaches can be roughly classified into three groups: interactive QE, semantic dictionary QE and the QE method based on documents set. In this section, each approach is briefly explained.

Interactive QE [1]: In interactive QE, a user is shown a list of terms suggested by the system after entering his query. Through human-machine interaction, undesired terms will not be added to the query string. The system can get good result, but the method need people familiar with the professional domain knowledge, that is often beyond normal users' capacity.

Semantic Dictionary [2-5] QE: Many researchers have tried to use semantic dictionary, such as WordNet for QE. But the results have not been as good as expected. In WordNet, a concept may include many related words. In those words, some are useless for query and can bring noises to the result, and they also are added to queries. In addition, WordNet is too broad and can not be used into special domain.

The QE based on Documents Set [6-9]:

Automatic Global Analysis: Global analysis is based on corpus wide statistics such as co-occurrence statistics about all possible pairs of terms, which normally results in a similarity matrix among terms. The terms, which are the most similar to the query, will be used to expand a query. Since the co-occurrence information for every pair of terms in the whole corpus is normally needed, the processing is computational resource consuming.

Automatic Local Analysis: The method assumes that the top-n retrieved documents are relevant, the system uses the terms contained in those documents as expansion terms and retrieves again. But when the top-n documents happen to be irrelevant, the QE will fail.

User Relevance Feedback: It requires users to read every retrieved document and tell the system that which documents are relevant. Terms are extracted from these documents for QE. This method is seldom deployed in practice because it puts burden on users and often irritates users.

Most of the existing methods get the improved recall, but at the same time, some terms added to the query bring noises to the result, and the IR returns many irrelevant documents, which lead to low precision. Avoiding noises when expanding queries is a researchable problem. In this paper, our research range is short query and professional domain IR, and the important research is how to choose expansion words. Because of wide and dim meaning of semantic dictionary, we use ontology instead of it to describe domain knowledge, and logic reasoner and the frequency of terms are used to choose fitting expansion words. The experiments show that the method we propose can get higher recall with the least decrease of precision.

3. RELATED CONCEPT: ONTOLOGY

Ontology [10] can be defined as a formal, explicit specification of a shared conceptualization. That means ontology defines accepted concepts and their relations in some special domain. It is machine-readable and can be reused.

4. SYSTEM ARCHITECTURE

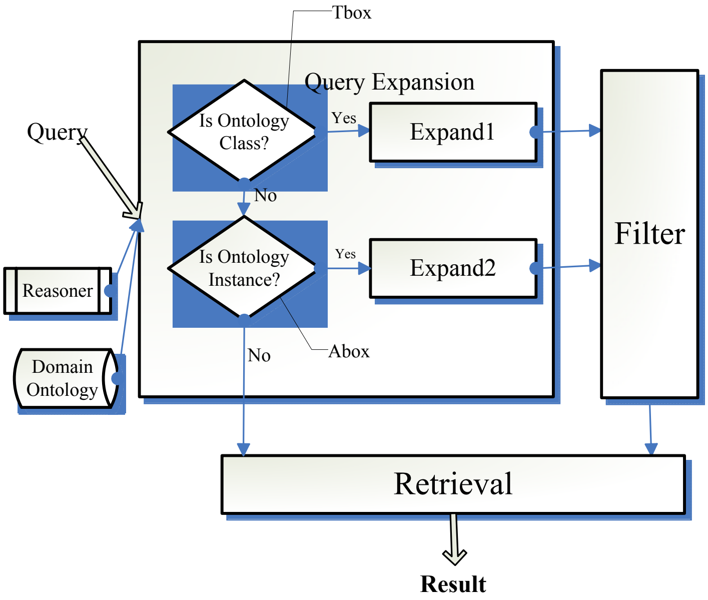

Figure 1. The Framework of Our System

Figure 1 sketches the architecture of our retrieval system. It is made up of Query Expansion, Filter and Retrieval modules. When user inputs a query which includes the terms in the Ontology, QE module provides a preliminary list of expansion words by using Pellet reasoner and domain ontology. Then, Filter module filters out some useless words, and delivers the new list to retrieval module. If the initial query does not include the words which belong to the ontology, it will be delivered to retrieval directly.

QE Module: The module expands the concept with the support of Pellet reasoner and domain ontology. The reasoner judges that the concept is a class or an instance in the ontology, then the module expands the concept by the judgment.

Reason: For a ontology user, the function of a reasoner is to obtain the cryptic knowledge from a ontology. There are two reason modes: TBOX and ABOX. TBOX: Terminology in a special domain, its task is to check the satisfiability of a concept in ontology. ABOX: Assertions about instance, its function is to check an instance belongs to which concept.

In QE module, firstly, a query will be checked if it is a concept in the ontology by TBOX. If it is, its subclasses, equivalent classes, and subinstances will be listed as expansion words (“expand1”in Figure 1). If the query is not a concept in the ontology, it will be check if it is an instance under a concept in the ontology. If it is, its brother instances will be list as expansion words (“expand2”in Figure 1).

Filter Module: The function of filter module is filtering out useless words in expansion words list. Filtering relays on the frequency of the word in document. If the occurrence frequency of a word is higher than a value, the word will be kept. If the frequency is lower than the value, the word will be filtered out. The principle is explained as follow.

1. Two documents: document I and document II

Document I: “徐华在北京工作，我也在北京工作。”

Document II: “他曾经在上海工作。”

2. Word Segment: Filter out useless words, and keep keywords.

Keywords of document I: “ 徐华 北京 工作 我 北京 工作 ”.

Keywords of document II: “ 他 上海 工作 ” .

3. Build Reverse Index:

Table 1. Reverse Index

Keyword	The name of document[Occurrence frequency]	Position
徐华	I[1]	1
北京	I[2]	2,5
工作	I[2],II[1]	3,6,3
我	I [1]	4
他	II[1]	1
上海	II[1]	2

From table1, we can know that “工作” occurs in document I twice and document II once. In the third column, “3,6” means occurrence position in document I, and “3” means the position in document II.The system stores above data as term dictionary, frequencies and positions files. Term dictionary stores the pointer to frequencies and positions. By the pointer, we can obtain keywords' occurrence frequencies in a document rapidly. Then, we can filter expansion words by the frequencies files.

Retrieval Module: The module is a full text IR which is made by Lucene.

5. ALGORITHM

```
 1. Short query from user: Q
{ if (Q belongs to Domain Ontology)
    {  Query Expansion;
       Filtering Expansion;}
else {delivers Q to Retrieval.}
}
Query Expansion//
If (Q is a concept in ontology){
    Listing its subclasses, equivalent classes and subinstances.}
else{
    If (Q is a instance in ontology){
Check which class the instance belongs to;Listing the subinstances of the class;}
}
Filtering Expansion//
For each expansion word, if its occurrence frequency from frequencies file is higher than some value, keep it. Otherwise, filter out the word from expansion word list.
2.  Documents set:
{
   Word segment;
   Building frequencies file;}
```

6. EXPERIMENTS

Our system has two characters: One is domain ontology for QE is closer to the professional documents set than common semantic dictionary, and that can improve the performance of professional IR. Another is that the choice of expansion words is based on text collection. This overcomes noises in IR affectively.

In order to evaluate the performance of our system, we made some experiments. The text collection comes from Sina or Yahoo , and consists of 82 articles. The searching domain is travel knowledge .

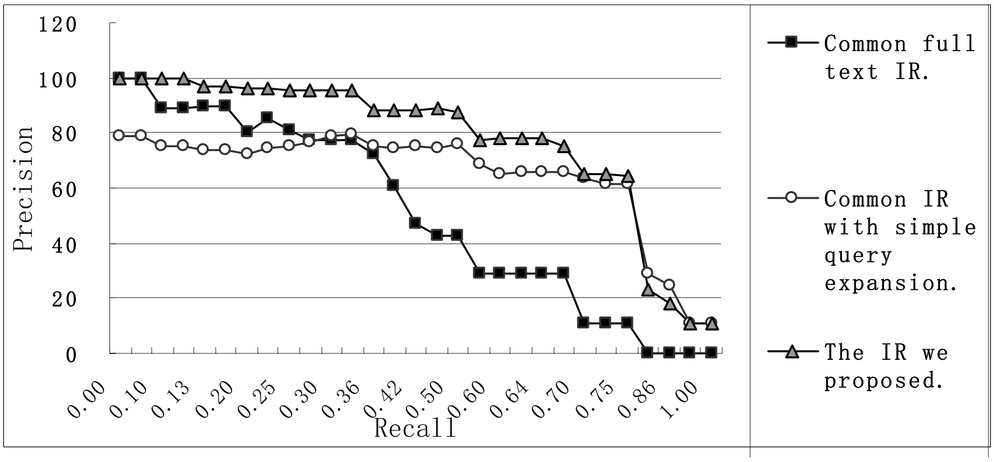

Figure 2. Average Precision/Recall Curves

Three search engines were built: common full text IR, IR with simple QE based on ontology and the IR we proposed in this paper. The average precision/recall curve [11] is effective to evaluate IR performance. Figure 2 shows that compared with full text IR, IR with simple query can not improve the performance of the IR consistently. With the increasing recall, the precision of the system is decreasing. However, the method we proposed in this paper increases the recall and does not decrease the precision.

7. CONCLUSIONS

We proposed a QE method based on ontology and occurrence frequency. This model overcomes two drawbacks in traditional QE: weak description in domain knowledge and noises caused by QE. With the new expansion method, recall and precision are improved at the same time. But in our system, the perfection of ontology affects the performance of IR, perfecting the ontology is an important work.

REFERENCES

1. H. Lee, S. Lin, and C. Huang, Interactive Query Expansion Based on Fuzzy Association Thesaurus for Web Information Retrieval, in *Proc. of the 10th IEEE International Conference on Fuzzy System* (IEEE Press: Melbourne, Australia, 2001), pp.724-727.
2. J. Gonzalo, F. Verdejo, and I. Chugur**,** Using Eurowordnet in a Concept-Based Approach to Cross-Language Text Retrieval**,** *Applied Artificial Intelligence***.** Volume 13, Number 7, pp.647-678, (1999).
3. E.M. Voorhees, Using WordNet to Disambiguate Word Senses for Text Retrieval, in *Proc. of the Sixteenth Annual Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval* (ACM Press: New York, NY, USA, 1993), pp.171-180.
4. R. Mandala, T. Tokunaga, and H. Tanaka, Combining multiple evidence from different types of thesaurus for query expansion, in *Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (ACM Press: Berkeley, CA, USA, 1999), pp.191-197.
5. G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, Introduction to WordNet: an on-line lexical database, *International Journal of Lexicography*. Volume 3, Number 4, pp.235-244.
6. M.E. Maron and J.L. Kuhns, On Relevance, Probabilistic Indexing and Information Retrieval, *Journal of the Association for Computer Machinery*. Volume 7, Number 3, pp.216-244, (1960).
7. J.J. Rocchio, *Relevance feedback in information retrieval*, in *The SMART Retrieval System*, eds. G. Salton (Prentice-Hall, Inc.: Englewood Cliffs, NJ, 1971), pp.313-323.
8. G. Salton and C. Buckley, Improving retrieval performance by relevance feedback, *Journal of the American Society for Information Science*. Volume 41, Number 4, pp.288-297, (1990).
9. L. Song, Y. Cheng, and Q. Shan, Relevance Feedback for Information Retrieval System, *Journal of the China Society for Scientific and Technical Information*. Volume 24, Number 1, pp.34-41, (2005).
10. P. Paggio, B.S. Pedersen, and D. Haltrup, Applying Language Technology to Ontology-Based Querying: the Ontoquery Project, *Applied Artificial Intelligence*. Volume 17, Number 8 & 9, pp.817-833, (2003).
11. B.Y. Ricardo and R.N. Berthier, Retrieval Performance Evaluation, in *Modern Information Retrieval*, eds. X. Sui (China Machine Press: Beijing, 2005), pp.51-57.