A new Formal Concept Analysis based learning approach to Ontology building

Haibo Jia, Julian Newman, Huaglory Tianfield

School of Engineering and Computing, Glasgow Caledonian University, Glasgow UK
{Haibo.Jia, J.Newman, H.Tianfield}@gcal.ac.uk

Abstract: Formal Concept Analysis (FCA) is a concept clustering approach that has been widely applied in ontology learning. In our work, we present an innovative approach to generating information context from a tentative domain specified scientific corpus and mapping a concept lattice to a formal ontology. The application of the proposed approach to Semantic Web search demonstrates this automatically constructed ontology can provide a semantic way to expand users' query context, which can complement a conventional search engine.

1. Introduction

Development of web technology provides a networked platform for a distributed research community to disseminate their research contributions and acquire others' research findings. Digital libraries, E-Journals, E-Prints, scholarly websites and search engine tools offer researchers great capability to obtain online information. However massive amounts of information, lack of formalized domain knowledge representation and non-unified terminology bring about either an "information explosion" as result of polysemy or "information loss" where synonymy is overlooked. This inevitably affects the efficiency and effectiveness of researchers' information searching and browsing.

Current search engines employ user–specified keywords and phrases as the major means of their input. Digital libraries, such as ACM DL[1], add a metainformation layer, so that from a given author, journal, conference proceedings and predefined topic description, publications can be found. Google also use document similarity to extend the result of users' query. However, these services are not able to augment context in the process of search. They cannot assist the user to generate a proper query term according to the topics of interest and even cannot expand query terms according to semantic similarity and different semantic generality. A novice researcher often finds it difficult to define a query term which closely matches his/her information demand; furthermore he/she often wants to constrain or extend his query by terms at different levels of generality during the process of searching in order to discover more suitable documents. For instance, when we query Googlescholar [2] with the term "semantic web", the search engine can not return us what research topics this term is associated with and what other terms this term could be similar to.

[1] http://portal.acm.org/dl.cfm
[2] http://Scholarly.google.com

Semantic web provides a knowledge-based environment in which information can be well defined by ontology and intelligent application can better process linked data to improve the interactions between the user and computer system. Ontology is a conceptualization of a domain into human understandable but machine readable format consisting of entities, attributes, relationship and axioms [1]. In the document query scenario mentioned above, factors in the ontology could be used to expand the users' understanding of query term so that an extended query context will be provided.

In this paper, a new concept clustering based learning approach is proposed for ontology building. The application using the constructed ontology for query expansion is also demonstrated. The rest of this paper is organized as follows. Section 2 discusses related work. Section 3 discusses the detail of using formal concept analysis to extract ontology from a scientific corpus. Section 4 discusses the application of using semantic web technology to expand query context. Conclusion is given in the Section 5.

2. Ontology building by machine learning: states of the art

Ontology building is the process by which concepts and their relations are extracted from the data which can be free-text, semi-structured or data-schema. In general, it is unrealistic to use general-purpose ontologies for guiding such learning in a specific scientific or research domain. Krovetz & Croft [2] points out that 40% of the words in canonical form in the titles and abstracts of the Communications of the ACM are not included in the LDOCE (Longman Dictionary of Contemporary English). Recently, some researchers have looked for possible ways to automatically build domain ontology from scientific corpus and develop scholarly semantic web based on this ontology [3][4][5][6]. Consequently, a number of machine learning approaches are applied to ontology building, including scientific text structure based learning, syntactic pattern based learning and conceptual clustering based learning etc.

In [7], Makagonov observes that scientific text is highly hierarchical text so that words at different levels of abstraction are located in different parts of a scientific document. In most cases concepts in the domain description are more general than concepts in the conference or journal title and the later are more general than concepts in the individual document title and so on. Based on this observation, a scientific text based level-by-level automatic ontology learning is proposed for ontology building. This approach is simple but quite efficient for ontology building when learning from small corpus. However, the learning result could be negatively affected due to the authors giving their documents an inexplicit title (for example, a PhD thesis entitled "who are the Experts? E-Scholars in the Semantic Web"). This is, unfortunately not unusual in scientific publication. Furthermore, the amount of generality level has been decided before the learning, which may not be consistent with real concept structure.

In [3], an approach of extracting taxonomy using syntactical patterns is discussed, which is based on the work presented by Hearst [8]. In this approach, linguistic syntactical patterns such as "NP such as NP", "NP including NP" etc are used to recognize the concept and semantic relation namely is-a relation, here NP refer to

"Noun Phrase". It has been proved that this approach can result in quite high quality ontology. However, Hearst's patterns appear relatively rarely even in big corpora [9], So many useful concepts could be neglected.

Clustering algorithms have been broadly studied within the Machine Learning and Data Analysis community. Hierarchical concept clustering has been applied in the learning to build ontology. In [10], Bisson et al design general framework and a corresponding workbench –Mo'k- for user to integrate concept cluster method to build ontology. An agglomerative clustering algorithm is also used to present the result. However, this framework is general for various cluster methods. In [11], Philipp compares Formal Concept Analysis (FCA), Divisive and Agglomerative Clustering for learning taxonomies from text. The result shows FCA has low efficiency but very good traceability compared to other two methods. The clusters learnt by FCA also have their own intentional meaningful descriptions, thus facilitating users' understanding of generated clusters. Another advantage of FCA is that its final outcome is concept lattice rather than tree like forms produced in other two methods. Lattice form assures a concept may have more than one super or sub concept, which reflects real-life concepts organisation. Generally, the principle paradigm in most of these approaches is based on a distributional hypothesis, which assumes that terms are similar to the extent to which they share similar linguistic contexts and similar terms can be clustered together to form a concept. Thereafter according to various linguistic contexts, corresponding ontology can be developed.

In learning from scientific corpus to build scholarly ontology, the topical context rather than linguistic context is more widely considered. In [4], Quan proposes a Fuzzy Formal Concept Analysis (FFCA) framework to automatically generate ontology for scholarly semantic web. The information context is built by scientific documents (object) and keyphrase (attribute). The most frequent keyphrases occurring in the same papers are clustering to form hierarchical clustered concept to represent different research areas in a particular domain. However, Zhang [5] argues that document-keyword context allows a keyword that only occurrs in one document to be selected to compute concept lattice, which could result in large noisy information due to the authors' misuse of a keyword. He points out that information context should be built from the viewpoint of collection rather than an individual. So Zhang builds information context using keywords as both objects and attributes, where each object has particular keywords as attributes if that keyword occurs along with the object keyword in a document and meets a specified support threshold in the whole collection. The concept hierarchies learnt from this information context can bring improved precision for document classification.

3. Formal Concept Analysis based Learning approach to Ontology building

Different methods have been proposed in the literature to address the problem of (semi-) automatically deriving a concept hierarchy from scientific text. Our motivation is that this concept hierarchy should be applied in the users' query to expand query context. Users' query term should be identified in this structure; topical

similar terms should be clustered into the same concept and the intentional description of the concept should be better understood and commonly accepted by the practice of community. The previous research has shown FCA is an effective technique that can formally abstract data into a hierarchical conceptual structure with good traceability and understandability. In our research, the selection of context and the mapping from formal concept lattice to formal ontology representation are major consideration. The work flow of our proposed approach can be depicted as in Figure 1.

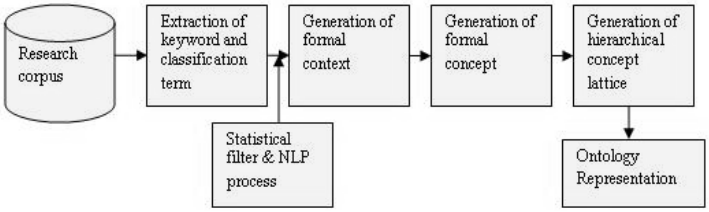

Fig.1. Ontology Learning Work Flow

3.1 Formal Concept Analysis

In order to better interpret our approach, we briefly recall some basic terminologies and definition of FCA and further detail can be found in [12].

Definition 1. A Formal Context is a triple (G, M, I) , where G is a set of objects, M is a set of attributes, and I is a binary relation between G and M (i.e $I \subseteq G \times M$) . $(g,m) \in I$ Can be read as object g has attribute m.

Definition 2. Formal Concept of context (G, M, I) is a pair (A, B) with $A \subseteq G, B \subseteq M, A' = B, B' = A$ where iff $A \subseteq G$ we define $A' := \{m \in M \mid \forall g \in A : (g,m) \in I\}$ and iff $B \subseteq M$ we define $B' := \{g \in G \mid \forall m \in B : (g,m) \in I\}$

Definition 3. Sub-Super concept relation: $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2 (\Leftrightarrow B_2 \subseteq B_1)$ is defined as (A_2, B_2) is superconcept of (A_1, B_1).

FCA uses order theory to analyze the correlations between objects, G, and their attributes, M. A concept is composed by a set of objects which are similar according to the interpretation of attributes. Inclusion relation between the object sets can reflect the sub-super relation between different concepts. The concepts and their relations can construct a concept lattice which will finally be converted to domain ontology; in this ontology only subsumption relation can be extracted rather than other enriched relation. The selection of object and attribute will vary based on different application.

3.2 Information Context Construction

In FCA, the selection of formal context is a crucial step. Different formal context will model different aspects of the information and result in different applications to

consume the information. For our case, we intend to model keywords in a research domain by topical similarity and subsumption which should expand users' search context and improve the interactive capability of traditional search engines.

In [5], information context is developed based on keyword-keyword pairs. If two keywords are correlated with the same keyword set, these two keywords are assumed to be semantically similar. The correlation is determined by cooccurence of keywords in the same document. In the concept lattice, a concept will be described by a set of keywords. However, in our point of view concept descriptors should be more intensively meaningful and controlled terms, which should be more easily understandable for users. On the other hand, the large number of keywords in the corpus will increase the size of attribute list so that the efficiency of FCA is to be affected. The candidate of attribute in the information context should be common topical terms in a domain which should be easily understandable and have suitable level of generality. In addition, its size should be restricted in a controlled range.

In ACM digital library, a Computing Classification System (CCS) is well defined. It is an existing knowledge base in computer science discipline to assist the author to classify their works. In CCS, the tree structure of topical terms normally used to classify the paper could be used to describe the keywords clustered within such topic because a set of keywords can be regarded as the representative of the paper. In most papers from ACM, author defined keywords and classifiers (i.e., classification terms) from CCS may occur explicitly. We propose a novel approach to utilising these resources to construct keyword-classification term context in computer science. In this context, keywords are explicit definition from author, which are also widely used by user to raise queries in routine search, and classification terms are from the controlled terms defined in CCS. Moreover the correlation between keywords and classifier can be implicitly discovered from the real classification of the paper made by authors themselves.

In order to implement our approach, we have downloaded 900 papers from ACM digital library to construct a tentative domain-dependent scientific corpus. Every paper's primary classification is below "**H**.3 INFORMATION STORAGE AND RETRIEVAL". The selection of a specified domain can avoid the sparse distribution of the keyword in the corpus. All the metadata information is parsed from the information page of the paper and then they are feed in predefined database; the data schema is depicted in Figure 2. The 8 keyword fields are defined, which basically

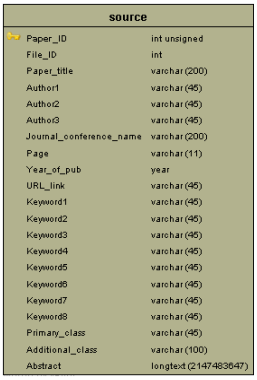

Fig.2. metadata table schema

cover the maximum number of author defined keyword in a paper. Due to existence of the multidisciplinary nature of the papers, it is very common that a series of classification terms are used to classify the paper. According to different degree of relevance, classification terms are further splited into primary and secondary. Although in current research we equally treat these two types of classifications, we envision different degree of relevance will affect the weighting of correlation between keyword and classification term. So in this scheme, we define two classification fields which could be used in the future research. In a record of the data table, all keywords and corresponding classification term are calculated and the relation weight between them is incremented. The pseudo-code of the corpus-wide method which, given a metadata data table of the corpus, returns the information context is presented in Figure 3. The snippet of the context with weighted relation is shown in Figure 4

```
HashMap <String, HashMap> Context=new HashMap<String,HashMap>();
HashMap <String, Integer> Relation;
for-each Record in Recordset
    for-each Classification in Record
         for- each Keyword in Record
             if (context.containKey(Keyword)==null)
                   Relation=new HashMap<String,Integer>;
                   Relation.put(Classification,1);
                   Context.put(Keyword,Relation);
             else
                  Relation=Context.get(Keyword);
                   if (Relation.containKey(Classification))
                       Integer_ weight=Relation.get(Classification)++;
                       Relation.put(Classification,Weight);
                   else Relation.put(Classification, 1);
                  end if-else
              end if-else
         end for
      end for
  end for
```

Fig.3. Pseudo-code of the context construction

personal agents----{Information Search and Retrieval=1, Distributed Artificial Intelligence=1, Learning=1}
web searching----{Information Search and Retrieval=1, Systems=1, CODING AND INFORMATION THEORY=1, Online Information Services=1, Content Analysis and Indexing=1}
scorm - lom----{Systems and Software=1, Online Information Services=1, Document Preparation=1}
replication----{Systems and Software=1, PERFORMANCE OF SYSTEMS=1}
computer mediated communication----{Information Search and Retrieval=1, Public Policy Issues=1}
edit distance----{Online Information Services=1}
ontology----{Systems and Software=4, Information Search and Retrieval=14, Interoperability=1, Library Automation=1, Formal Definitions and Theory=1, Models=1, User Interfaces=1, Distributed Artificial Intelligence=1, SOFTWARE ENGINEERING=1, Online Information Services=9, Information Storage=1, Learning=1, Knowledge Representation Formalisms and Methods=2}
daily delta----{INFORMATION STORAGE AND RETRIEVAL=1, Hypertext/Hypermedia=1}
order of insertion----{DATA STORAGE REPRESENTATIONS=1, DATA STRUCTURES=1, Content Analysis and Indexing=1}
service composition----{Online Information Services=1}
passages----{Information Search and Retrieval=1}
discovery query----{Information Search and Retrieval=1}
collaborative filtering----{Hypertext/Hypermedia=1, Digital Libraries=1, Systems and Software=5, Information Search and Retrieval=21, Group and Organization Interfaces=1, Clustering=1, Distributed Systems=1, Communications Applications=1, User Interfaces=2, Learning=1, Online Information Services=5}
linguistic analysis of web text----{Digital Libraries=1, Information Search and Retrieval=1, SPECIAL-PURPOSE AND APPLICATION-BASED SYSTEMS=1, Content Analysis and Indexing=1} ……

Fig.4. Information Context snippet

3.3 Statistic filter and Natural Language Processing

The initial information context consists of 2201 keywords and 107 corresponding classification terms. According to FCA, a set of keywords will be clustered to the concept and the associated classification terms will be description of the concept. To some extent, ontology is consensus in a domain. So it is necessary to assure that the keywords and their associated classification terms are reasonable and commonly acceptable. Here we propose an empirical statistic approach to filtering out uncommon accepted keywords and associated keyword- classifier relations, which could be caused by author's individual preference, their error prone definition or correlation approach we use to construct information context. In this approach, we are mainly concerned about the occurrence frequency of the relation between keyword and associated classification term. We assume that in our corpus if the number of associated relations between a keyword and a corresponding classification term is higher than a threshold, this relation will be retained for final ontology construction. Otherwise this relation will be deleted from the context; furthermore if there is no relation between a keyword and any classification term, the keyword will be deleted. Likewise, if no keywords are related with a specified classification term, this term will be removed from the context. The thresholds n=1,2,3,4,5 respectively are used to this information context, the result is shown as in Figure 5. Empirical study shows n=3 is the idealist threshold value, which can give a good trade-off between the quality of keywords and the quantity of classification. After filtration 173 keywords and 12 classification terms are kept in the context, multi-value relation is also replaced by binary relation. The new information context is shown as in Figure 6.

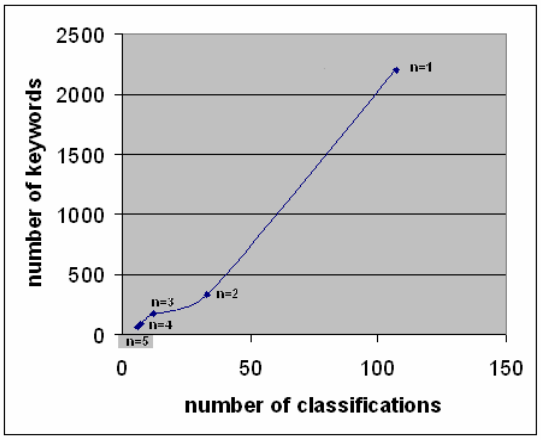

Fig.5. Filtration with different threshold

Additionally, the problem of phrase morphology is also considered. Keywords such as “page rank” vs “page ranking”, “language model” vs “language models” etc are the same phrase with different form, which should be treated as the same keyword. So each word in the keywords is stemmed using porter algorithm [13] during the process of the context generation.

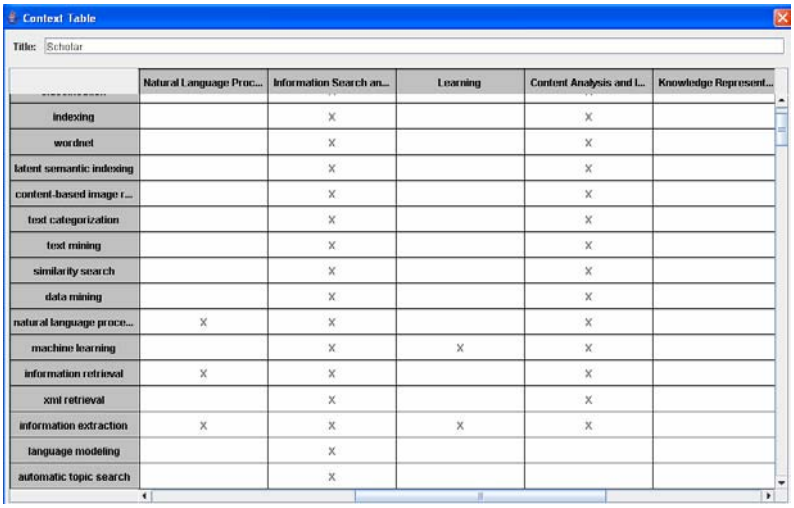

Fig.6. Information context with binary relation

3.4 Formal concept generation

After constructing the context the concepts and concept lattice can be generated using Formal Concept Analysis. Here we use ToscanaJ [14] – Java implementation open source of Classical FCA tool named “Toscana”. Constructed context is input; an

extracted concept lattice is produced shown as in Figure 7. In the concept lattice, each node represents a concept that has object (white box) and attribute (grey box). In this case, object is represented by a set of keywords and attribute is represented by classification terms. The link between two concepts represents super-sub concept relation. The object of each concept is a union of all the objects from the subconcept and itself, likewise each attribute is an intersection of all the attributes from superconcept. For example in Figure 7, Concept 3 is the subconcept of Concept 1 and concept 2; so the attribute of concept 3 should be labelled by {Information Search and Retrieval}^{Content Analysis and Indexing}. In addition, the keywords in Concept 3 and its subconcept can be referred to as a conceptual cluster described by this label. Obviously the subconcept has fewer keywords and more restrictive classifier than superconcept. This structure can be used to expand users' query.

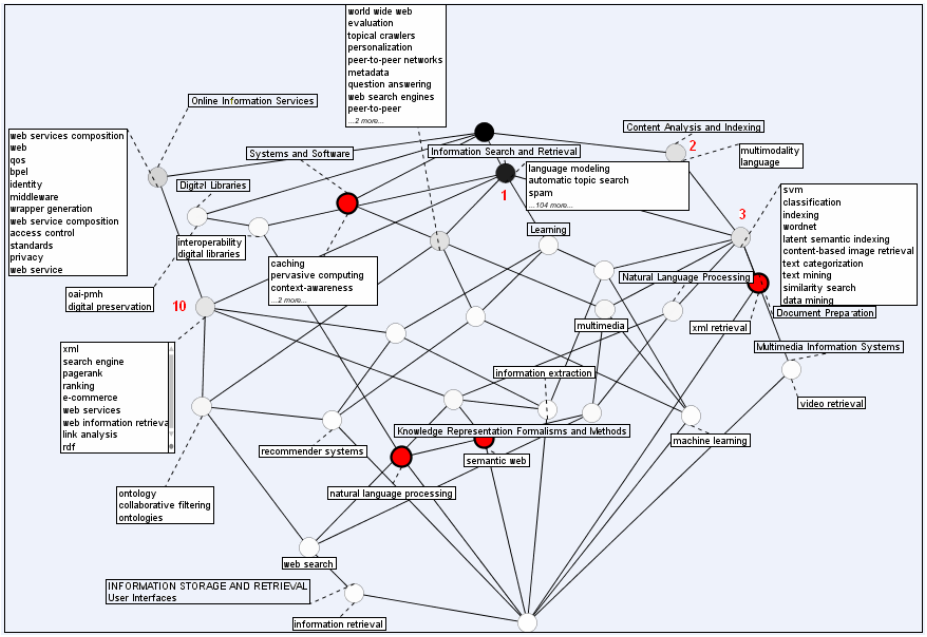

Fig.7. Concept Lattice

3.5 Formal Ontology Representation

Ontology is formalization of concepts and their relations between concept, which can be utilized by agent to better interpret and consume the information. Generally ontology can be formally defined by <C,P,I,S,E>, where C refers to Class; P refers to property of Class; I refers to instance of Class; S refers to subsumption relation and E refers to other Enriched relation. In our case, only subsumption relation is considered. In the above concept lattice, the concept can be mapped to class in the ontology definition; the keywords in each concept can be mapped to the instance of ontology; the element of attribute will be mapped to the property of ontology and finally sub-super concept relation is equivalent to subsumption relation in ontology. This

ontology can easily be represented by standard ontology language RDF/RDFS. The snippet of ontology is shown as in Figure 8.

```
<rdfs:Class rdf:ID="Concept1"/>
<rdfs:Class rdf:ID="Concept2"/>
<rdfs:Class rdf:ID="Concept3"/>
<rdfs:Class rdf:ID="Concept4"/>
<rdfs:Class rdf:ID="Concept6">
    <rdfs:subClassOf rdf:resource="#Concept4"/>
    <rdfs:subClassOf rdf:resource=”#Concept2”/>
</rdfs:Class>
…….
 <rdf:Property rdf:ID="Online_Information_Services">
    <rdfs:domain rdf:resource="#Concept1"/>
 </rdfs:Property>
<rdf:Property rdf:ID="Digital_Libraries">
    <rdfs:domain rdf:resource="#Concept2"/>
  </rdf:Property>
<rdf:Property rdf:ID="System_and_Software">
    <rdfs:domain rdf:resource="#Concept3"/>
  </rdf:Property>
<rdf:Property rdf:ID="Information_Search_and_Retrieval">
    <rdfs:domain rdf:resource="#Concept4"/>
 </rdf:Property>
 <rdf:Property rdf:ID="Learning">
    <rdfs:domain rdf:resource="#Concept8"/>
 </rdf:Property>
 …….
<Concept1 rdf:ID="middleware"/>
<Concept7 rdf:ID="question_answering"/>
<Concept17 rdf:ID="ontology"/>
<Concept16 rdf:ID="xml_retrieval"/>
<Concept7 rdf:ID="topical_crawlers"/>
<Concept4 rdf:ID="spam"/>
<Concept10 rdf:ID="xml"/>
<Concept9 rdf:ID="data_mining"/>
…….
```

Fig.8. Ontology Snippet

4. Query Expansion Application

The concept in the above ontology can be regarded as the query context, the expansion within a context and among the context provides a solid mechanism to expand users’ query. In our work, we intend to build ontology driven expansion functionality on the top of search engine rather than replacing it.

In this application, SPARQL can be used to express queries across RDF data sources to discover the relevant concept. If a user raises the initial query using keyword “xml”, the narrowest concept including this keyword can be located by SPARQL statement (1) in Figure 9. In our example corpus, concept 10 will be returned (cf. Figure 7). “search engine” “pagerank” “ranking” “RDF” “web service” “link analysis” etc keywords list in this concept can also be recommended by

statement (2). By executing statements (3), all the classification terms associated with this concept and its super concept can be obtained. As there is no inference function in the SPARQL itself, so a program model is needed to trace all the super concepts. Here classification terms {Online Information Services}^{Information Search and Retrieval} can be obtained as the description of this query context. If this query context is too narrow to user's requirement, user can filter this classification term set. Statement (4) can be used to relocate new concept which implicitly represent a new query context, thereafter more relevant keywords will be recommended. Likewise if the user wants to narrow his query to obtain more pertinent query context and keywords, he can navigate the subconcepts to relocate new query context and keyword list. After this process, the selected keywords will be utilized by search engine to return documents.

```
Prefix :<http://www.owl-ontologies.com/scholar.owl#>
Prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
Prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>

(1) Concept= SELECT  ?Concept
             WHERE {:xml  rdf:type  ?Concept.}

(2) KeywordList= SELECT ?keywords
                 WHERE {?keywords rdf:type :Concept}

(3) SuperConcepts= SELECT ?SuperConcept
                   WHERE {:Concept rdfs:subClassOf ?SuperConcept}

  Classifications= SELECT ?Classification
                      WHERE{
                     {?Classification rdfs:domain :Concept} UNION
                     {?Classification rdfs:domain :SuperConcept1} UNION
                ….. UNION{?Classification rdfs:domain :SuperConceptn}}

(4) Concept= SELECT ?Concept
              WHERE {
              {:Classification1 rdfs:domain  ?Concept} UNION
              {:Classification2 rdfs:domain ?Concept} UNION
               ….. UNION {:Classficationn rdfs:domain ?Concept}}
```

Fig.9. SPARQL Statements Example

5. Conclusion and Future work

In this paper, we have presented a Formal Concept Analysis based learning approach to building domain specific ontology from scientific corpus. The keyword-classifier context has been utilised to generate information context. The semantic web technology has been adopted to demonstrate function of query expansion driven by this ontology, which can be applied to complement the capability of search engine in digital library.

Currently, the author-defined keywords are used as resource. In the future, an approach to automatically extracting keywords from text will be investigated. A larger scientific corpus in a broader computing domain will be constructed; integration system between our approach and normal search engine will be developed.

References:

[1] Guarino, N., Giaretta, P.: Ontologies and Knowledge Bases: Towards a Terminological Clarification. IOS Press, Amsterdam (1995)

[2] Krovetz, R., Croft, W.B.: Lexical Ambiguity and Information Retrieval. Lexical Acquisition: exploiting on-line resources to build a lexicon, pp.45-65. Hillsdale, New Jersey, Lawrence Erlbaum Associates (1991)

[3] Novacek, V., Smrz, P., Pomikalek, J.: Text Mining for Semantic Relations as Support Base of a Scientific Portal Generation. In Proceedings of 5th International Conference on Language Resources and Evaluation, pp1338-1343. ELRA, Genova (2006)

[4]Quan, T., Hui, S., Fong., A., Cao,T.: Automatic Generation of Ontology for Scholarly Semantic Web. In The Semantic Web – ISWC 2004, LNCS, pp726-pp740. Springer, Hiroshima (2004)

[5]Zhang, G., Troy,A., and Bourgoin, K.: Bootstrapping Ontology Learning for Information Retrieval Using Formal Concept Analysis and Information Anchors. In 14th International Conference on Conceptual Structures. Alborg (2006)

[6]Zhao, P., Zhang, M., D., Tang, S.: Finding Hidden Semantics behind Reference Linkages: an Ontological Approach for Scientific Digital Libraries. In The Database Systems for Advanced Applications, 10th International Conference, LNCS, pp699-710. Springer, Beijing (2005)

[7]Makagonov, p., Figueroa, A., Sboychakov, K., Gelbukh, A.: Learning a Domain Ontology from Hierarchically Structured Texts. In Proc. of Workshop "Learning and Extending Lexical Ontologies by using Machine Learning Methods". At 22nd International Conference on Machine learning. Bonn (2005)

[8]Hearst, M., A.: Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 14th conference on Computational linguistics, pp539-545. Morrisotown, NJ, USA (1992)

[9]Cimiano,P., Pivk,A., Thieme,L.: Learning Taxonomic Relations from Heterogeneous Sources of Evidence. Ontology Learning from Text: Methods, Evaluation and Applications Volume 123 of Frontiers in Artificial Intelligence, pp 59-73. ISO Press (2005)

[10]Bisson, G., Nédellec, C., Caňamero, L.: Designing clustering methods for ontology building – The Mo'k workbench' in proceedings of the ECAI Ontology Learning Workshop. Berlin (2000)

[11]Cimiano, P., Hotho, A., Staab, S.: Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. In Proceedings of the European Conference on Artificial Intelligence (ECAI), pp 435-439. Valencia (2004)

[12]Carpineto, C., Romano, G.: Concept Data Analysis – Theory and Applications. John Wiley & Sons Ltd, England. (2004)

[13] The Porter stemming algorithm, http://www.snowball.tartarus.org/algorithms/porter/stemmer.html

[14] ToscanaJ Suite, http://toscanaj.sourceforge.net.