Preliminary Investigations into Ontology-based Collection Selection

J. D. King, Y.Li, P.D. Bruza and R.Nayak
School of Software Engineering and Data Communications
Queensland University of Technology
QLD 4001 Australia
{*j5.king, y2.li, p.bruza, r.nayak*}*@qut.edu.au*

Abstract *This article tackles the collection selection problem from the query side. Queries are enhanced by mapping them to subjects in an ontology; the associated subject classification terms are then employed to retrieve collections. An experimental comparison was performed with the state of the art ReDDE system, which relies on estimates of collection size to rank collections. Although the research is preliminary, there is some support to the hypothesis that this approach mitigates the need for collection size estimates in collection selection.*

Keywords Information Retrieval, Document Databases, Digital Libraries

1 Introduction

Currently human experts are better at identifying relevant documents than the state of the art information retrieval methods. Human experts are also currently better at classifying documents than the state of the art automatic classification methods. One factor that makes human experts superior from computer programs is '*world knowledge*'. World knowledge encompasses information on topics such as philosophy, psychology, religion, social sciences, language, natural sciences, mathematics, technology, the arts, literature, geography, and history. In this study we make use of world knowledge stored in an ontology and apply it collection selection. The term "ontology" has a number of conceptions. For the purposes of this article, an ontology is defined to be a hierarchical structure, whereby the nodes correspond to subjects. Each subject is characterized by a set of subject classification terms.

Ontologies have been used in Artificial Intelligence for a variety of applications. However, a major problem associated with building an ontology which covers a large number of domains is the human-hours that would be required to construct it. This problem is called the *knowledge acquisition bottleneck*. The aim of this research was to quickly, cheaply and simply build an ontology which has both a wide range of knowledge and capabilities across many different domains. While some information retrieval systems use terms to describe collections, our method uses subjects to describe collections. The power of a subject based approach is better understood through the following example. If a user issues the query "matrix factorisation methods" into a search engine, he or she would probably expect documents about "singular value decomposition" to be returned. In this article, we attempt to exploit this capability in the following way: An arbitrary query is mapped into the ontology yielding a set of subjects. The subject classification terms of each subject are then accumulated into a query which is used to rank collections. It is important to note that this approach does not rely on estimating the collection size. Even though collection size is acknowledged as being an important feature determining collection selection effectiveness [31, 30], it is also acknowledged that acquiring reliable estimates can be a costly and challenging problem. The hypothesis behind this article is to examine whether a subject based approach may compensate for not having collection size estimates. In other words, we are tackling the collection selection problem for the query side, rather than the collection side.

The rest of this paper is organised as follows. Section 2 introduces our automatic ontology learning method, Section 3 shows our collection selection method, Section 4 shows related work, Section 5 shows our experiment data, Section 6 gives our experiment results, and Section 7 concludes the paper.

2 Automatic Ontology Learning

The problem with many ontologies is that they only cover a small number of domains, and each domain has to be manually added by a domain expert. The method presented in this section automatically creates an ontology covering hundreds of different domains. Automatic ontology learning will be a great improvement, enabling technologies to facilitate the creation of the semantic web.

There are three methods of ontology learning, each offering a trade-off between speed and accuracy. The three methods are:

Proceedings of the 11th Australasian Document Computing Symposium, Brisbane, Australia, December 11, 2006.

1. to generate rules from free text (fast but inaccurate)

2. to generate rules from expert created and/or classified materials such as dictionaries and encyclopedia texts

3. ask domain experts to populate the ontology by manually entering rules (slow but arguably accurate)

The second method is adopted in this research as it provides a balanced approach.

The stages involved in the ontology construction process are:

1. Selecting a classification taxonomy

2. Identifying a training set

3. Downloading a training set and populating the ontology

4. Cleaning up the ontology

We refer to this as the "IntelliOnto" construction process. (See [17] for more details).

Ideally there are several desirable properties in a good expert classified taxonomy. The taxonomy should cover a wide number of subjects, be carefully constructed, be standard across the world, and be available in different languages. It was decided to use the Dewey Decimal System, a widely used library classification system[1]. The system has been used for the classification of a large number and wide range of materials. The Dewey taxonomy is based on a pattern of ten root nodes, with each root node having ten child nodes. Each child node has another ten child nodes with this pattern continuing downwards. There can be many different levels of the taxonomy, depending on how precise the subject match is. There are 1,110 classification nodes at the top three levels of the taxonomy, with many more nodes in the lower levels of the taxonomy. There are some low-level subject nodes that are unused because of depreciation or limited coverage. In this paper only the top three levels of the taxonomy are used.

Figure 1 shows part of the Dewey taxonomy, and Figure 2 shows a more detailed portion of the taxonomy. Each Dewey Decimal Code (DDC) provides the best possible classification for each item.

The desirable properties of a training set are that it is large, of high quality, and covers a wide range of subjects. A data set reflecting these requirements is the Queensland University of Technology Library Catalogue[2], which contains over 500,000 usable items, although we only sampled 80,000 items for this research.

[1]For a full listing of the classifications see http://www.tnrdlib.bc.ca/dewey.html.

[2]See http://libcat.qut.edu.au/ This library web site is excellent for use as a training set because most of the entries have extra meta-information such as descriptions and summaries.

term	term count
software	281
programming	205
security	200
program	191
web	152
object	117
database	117
programs	105

Table 1: Terms that occur most frequently in *005 Computer programming, programs, data*

(It should be noted the Queensland University of Technology Library Catalogue is but an exemplar of an ontology which can be employed. The IntelliOnto method is by no means tied to this particular source of knowledge).

This data set was used to populate the ontology with world knowledge. Each document in our training set is assigned a Call Number. These documents have been carefully classified by experts in the field, and the information is of superior quality to other web based directories.

2.1 Mining From the IntelliOnto Ontology

Once the ontology base has been constructed, classification rules are mined from it. These rules are then used to classify collections. There are many different classification rules that can be mined from the ontology by using the terms, the subjects, and the taxonomic structure. By finding patterns between subject nodes and terms we are able to extract classification rules. These rules can then be made more useful by applying the taxonomic nature of the Dewey Decimal system.

The subject classification terms characterizing a subject need to be carefully selected. These terms should preferably be subject-specific (occurring within few or no other subjects) and should occur frequently within the subject and infrequently in other subjects. It is difficult to decide which terms to select as there are many possible terms to describe a subject. Many terms may not occur in common English dictionaries yet are still valuable for classification. These may include technical or subject specific terms such as conference names, acronyms and names of specialist technology. Some examples from computing are *RMI*[3], *SMIL*[4], *XSLT*[5], and *servlet*[6]. Few standard English dictionaries include these terms, yet if any of these acronyms occur in a document it is likely the document covers a subject related to computing.

Our first term selection method, highest term frequency, involves selecting the most popular terms from

[3]Remote Method Invocation.

[4]Synchronized Multimedia Integration Language

[5]Extensible Stylesheet Language Transformation.

[6]"*A Java application that, different from applets, runs on the server and generates HTML-pages that are sent to the client*" http://www.softwareag.com/xml/about/glossary.htm

Term	Count	Support	Confidence
c#	55	0.00003840	1
j2ee	48	0.00003351	1
javabeans	43	0.00003002	1
fedora	27	0.00001885	1
sax	27	0.00001885	1
awt	25	0.00001745	1
xsl	23	0.00001606	1
jdbc	23	0.00001606	1
oo	20	0.00001396	1
unicode	20	0.00001396	1

Table 2: Terms for *005 Computer programming, programs, data* with a confidence score of one.

each subject. Table 1 shows the most frequent terms for the subject *005 Computer programming, programs, data*. Our second term selection method, highest support and confidence, involves finding the most distinguishing (or unique) terms from each subject based on confidence and support. Table 2 shows the most distinguishing terms for the same subject. These terms cluster around the Dewey Decimal code "005". The nodes are grouped based on the third level of the taxonomy, any groupings below this level are not considered.

A term-subject pair $p(t \rightarrow s)$ in $M(O)$ with their confidence and support values is referred to as a pattern $p(t \rightarrow s) :=< t, s, conf(t \rightarrow s), sup(t \rightarrow s) >$ in this paper, where $t \in T, s \in S, conf(t \rightarrow s) = [0, 1]$ and $sup(t \rightarrow s) = [0, 1]$. We use a modified support and confidence method for our system, in order to accommodate the taxonomy. The $conf(t \rightarrow s)$ and the $sup(t \rightarrow s)$ in the pattern describe the extent to which the pattern is discussed in the training set. The $conf(t \rightarrow s)$ and $sup(t \rightarrow s)$ are defined as follows:

$$conf(t \rightarrow s) = \frac{sf(t, s)}{sf(t)} \quad (1)$$

$$sup(t \rightarrow s) = \frac{sf(t)}{n} \quad (2)$$

where $sf(t, s)$ is the number of child subjects under s (including s) with t occurred in the $termset$. The greater $sup(t \rightarrow s)$ and $conf(t \rightarrow s)$ are, the more important the term t is to the subject s.

Of the two ranking methods, the terms selected with high confidence and support thresholds seemed to be better for collection selection than the terms selected by highest frequency. Some of the more frequent terms were so common across different subjects that they could virtually be considered stopwords. The results presented in this paper therefore only use the highest confidence and support method.

3 Collection Selection

Collection selection is the selection of an optimal subset of collections from a large set of collections for reducing search costs [4, 11, 15, 25, 7, 8, 14, 29, 13, 3]. A central aim of collection selection is to accurately classify the content of each collection being evaluated. Once the content of each collection has been determined, the best subset of collection can be returned to serve an information need[7].

By way of illustration take collections, called *Collection A* and *Collection B*. Collection A contains information on the *creative arts* and no information on *social science*. Collection B contains information on *social science* and less information on the *creative arts* than Collection A. Each collection is treated as a "black box" and no prior knowledge of the contents is assumed. A human expert is used to generate a set of significant

[7]Many collection selection methods require direct access to or communication with each collection, yet few internet collection allow this. Thus other methods of evaluating collection content must be developed.

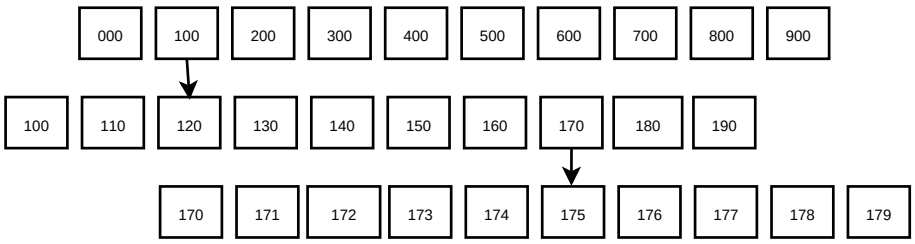

Figure 1: The Dewey Decimal taxonomy

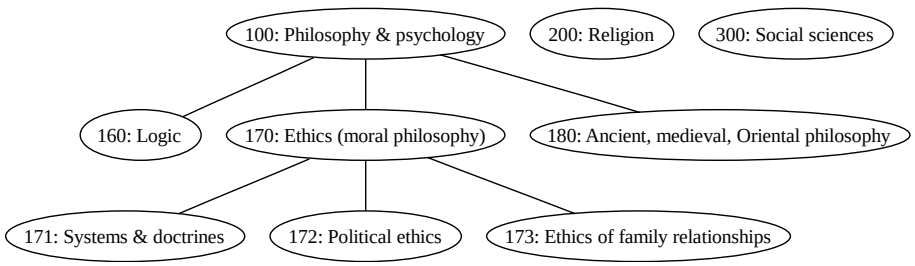

Figure 2: A Portion of the taxonomy

classification terms for each subject. For the *creative arts* subject the set of classification terms may include *opera*, *ballet*, and *Mozart*. The set of terms which best classifies each subject is used to query each collection and the number of times each term occurs in each collection is recorded. Accordingly these results are then used to classify each collection. It can be shown that Collection A is more suitable for finding information about the creative arts than Collection B, and any time a user requests information about the creative arts, Collection A can be returned as the best possible source for information.

As stated above, a collection can be treated as a black box with no prior knowledge of its contents. All that is known is that when some information is sent, some information is returned from it. Based on what is returned some knowledge it gained about its contents. There are two main questions that need to be answered in order to find information about the contents of the black box.

1. Decide what to send to the black box?
2. Decide what to do with the information returned from the black box?

An ontology answers the first question. By using an ontology, we aim to make collection selection more precise. Subject classification terms are used to probe black boxes.

Taxonomy addresses the second question. By transforming the probe term results into a taxonomy, a detailed view of the subjects contained in the black box is achieved.

In collection selection, *query probing* [6, 5] is commonly used to discover the contents of uncooperative collections. Query probing involves sending a set of query terms to a collection and using the results to form conclusions about the collection's content.

3.1 Methods of Collection Selection

The method for evaluating each collection for each query is as follows:

1. Extract "<title>" queries from TREC Topic Queries 51-100.
2. Convert each query into a set of four third-level Dewey Decimal codes using the Q.U.T. Library catalogue search engine.
3. Convert each Dewey Decimal code into a set of query probe terms taking the ten terms with highest support and confidence values for each Dewey Decimal code from the ontology.
4. Send each set of query probe terms to each of the collections one-by-one using the Zettair [1][8] search engine.
5. Extract the number of results for each query probe term from the Zettair results.
6. Sum the results from Zettair together and use them to rank each collection.

In our method the query probe terms from each subject node of the third level of the taxonomy are extracted. While it was difficult to decide how many classification terms to extract for each subject node, the use of more terms allows better results for collections which have a wider but more shallow coverage of a subject. However these collections may not have as high quality results as ones that provide deeper results for part of a subject. The use of fewer terms would result in better results for collections which have a deeper coverage of some aspects of a subject but poor results for collections which have a wider coverage of a subject. In our experiments the top ten results from the highest confidence and support for each subject node are used.

Once the query probe terms for each subject have been extracted from the IntelliOnto ontology they are sent to each collection. The number of results for each term from each collection is extracted and saved.

Once the query probe terms have been sent to the collection, and the results gathered, the terms need to be grouped into Dewey Decimal subject codes. To calculate the Dewey Decimal subject code results, the sum of the set of terms used to query probe the collection for each Dewey Decimal subject is taken. For example, if ten terms from a subject are used to query probe a collection, the results for each of the ten terms will be added together and this result recorded as the result for this subject code.

The query score for each subject in each collection is the sum of the ten results for each of the ten query probe terms.

4 Related Work

In this work a large scale ontology was built and used for collection selection. Literature related to this ontology based collection selection method is now reviewed.

4.1 Collection Selection

Collection selection is becoming more and more important as the number of collections on the internet increases daily. *Collection selection* is the matching of a set of related collections with an information need. The problems of collection selection have been addressed in previous work such as CORI [4] and GlOSS [14]. CORI assumes the best collections are the ones that contain the most documents related to the query. GlOSS uses a server which contains all the relevant information of other collections. Users query GlOSS which then returns an ordered list of the best servers to contact to send the query to. In a comparison of CORI and GlOSS [7] it was found that CORI was the best collection selection method, and

[8] Zettair is a compact open source TREC and HTML search engine from the R.M.I.T. University.

that a selection of a small number of collections could outperform selecting all the servers and a central index.

Web based collection selection introduces its own set of problems, in that there is usually no direct access to a collections statistics, and that there is rarely cooperation between the collections and the collection broker. Our previous work [19, 21] in web based collection selection used query sampling methods that did not require communication with the broker or metadata about each collection. Singular value decomposition was then used on the results of the queries to select the best collection. These techniques were tested on the INEX collection with satisfactory results. In other work [36], a subject based approach was used for information fusion and was found to be promising and efficient. In [20] a short preview of the work presented in this paper was presented.

Si et. al. [31] present a web based modification of CORI called *ReDDE* which performs as well as or better than CORI by using a collection size estimate to supplement selection. They introduce an collection size estimation technique which is more efficient than in other estimation techniques such as the capture-recapture method [24].

Hawking et al [16] presented a method which used both centralised and distributed collection selection techniques. They also made use of anchor text to extract information on collections that have not been indexed.

Si et. al. [32] presented a method for minimising the poor quality results returned by collections which have not implemented good information retrieval methods. By including the retrieval performance of each collection in the collection ranking, this problem can be reduced. A method for approximating the retrieval effectiveness of a collection, known as RUM, was presented. The RUM method was compared to CORI and outperformed CORI in all the experiments conducted.

A common problem with traditional collection selection techniques are that they require communication between the search broker and collections, or that they need topical organisation. In this paper we presented a form of collection which does not need communication between the search broker and collections, and does not need topical organisation.

4.2 Ontology Learning

There is a growing body of work covering automatic and semi-automatic ontology learning. Automatic ontology learning has emerged as a separate entity from other ontology research, drawing from data mining, machine learning and psychology. However, automatic ontology learning is still very difficult to achieve other than in very specialised domains. We will briefly summarize some of the key research to date.

Maedche et. al. [27] presents methods for semi-automatically extracting ontologies from domain text. This includes methods for determining the measure of relationship between terms and phrases. Some ontology mining algorithms have been mentioned in [28, 26], which are the discoveries of the *backbone taxonomy* and the non-taxonomic relation.

Esposito et al. [9] provided semi-automatic ontology learning based methods for transforming raw text into a computer readable representation, enabling a computer to learn a language from training examples.

Faure et. al. [10] claims to have built a machine learning clustering system which learns subcategorization frames of verbs and ontologies from unstructured technical natural language texts. Unfortunately, in this example the methods were only tested within a single limited domain of cooking recipes which is itself highly structured (ie ingredients and cooking methods are fields common to all recipes).

Buitelaar [2] selected 126 classification types and used WordNet as an ontology to assign almost forty thousand polysemic noun terms to one or more types in an automatically generated ontology. Each term could be disambiguated by what set of categories it belonged to or is excluded from. These groupings could then be used to tag corpora to aid automatic processing of data.

Suryanto et. al. [34] applied ontology learning to an existing well structured ontology allowing rapid extraction of rules. Kietz et. al. [18] applied semi-automatic ontology learning tools to a company intranet environment where natural language was mined for information.

Li et. al. [22, 23] presented a method of automatic ontology learning and refinement which can be used to model web user information needs. Stojanovic [33] used an ontology to refine search queries by removing term ambiguity. Queries were taken and mapped to their neighborhoods in a controlled vocabulary, then the neighborhoods were presented to the user for assessment. Gauch [35] uses hierarchal weighted ontologies to create a personalised user profile and to assist web browsing. The ontologies are used to classify web pages and user browsing habits into different categories, and the user profiles are matched to spidered web paged. Gandon [12] provided methods for managing distributed knowledge and assisting corporate activities by using ontologies.

The above references all contain examples of ontology generation and ontology learning. However many of the above examples use only a small, domain specific ontology with limited application. In this work we automatically create a large ontology covering hundreds of different domains.

5 Experiment Data

For the experiments, four well known collection selection testbeds were derived from the TREC Tipster collection ; Trec123-100col-bysource, Trec123-2ldb-60col, Trec123-AP-WSJ-60col, and Trec123-FR-DOE-81col. These testbeds cover a range

of environments, from the base which contains many small collections of the same size (trec123-100col), a mixture of small uniform sizes and two large collections with similar relevant document density (Trec123-2ldb-60col), a mixture of small uniform sizes and two large collections which contain a high concentration of relevant documents (Trec123-AP-WSJ-60col), and a mixture of small uniform sizes and two large collection which contain a low concentration of relevant documents (Trec123-FR-DOE-81col). For a full description of each testbed see Si and Callan [31].

For our queries we take the TREC Topic Queries 51-100 from the Tipster collection. Because the IntelliOnto ontology was computed form the Queensland University of Technology's library catalogue, pilot studies in overlap between TREC queries and the ontology revealed deficiencies in the ontology's coverage. As the primary goal was to assess performance without relying on collection size estimates, only those queries deemed to map suitably into the ontology were employed. The queries from the TREC Topics 51-100 used are 63, 65, 66, 70, 71, 74, 75, 82, 85, 86, 96, 97, and 98. We will cover the full 50 queries in later work. We experimented sending the actual query as both a phrase and as single terms and found that the query performed much better as a phrase than as single terms.

6 Experiment Results

The experiments evaluating the IntelliOnto were done on four testbeds(Section 5). The relevance judgements for each of the TREC Topic Queries 51-100 were taken from TREC website, the file names were *qrels.51-100.disk1.disk2.parts1-5.tar.gz* and *qrels.51-100.disk3.parts1-5.tar.gz*. From these the "ideal" baseline was computed: for each query, collections were ranked on decreasing order of the number of relevant documents they contain.The ideal calculated a baseline for each collection using these relevance judgements. To shed light on the question of collection size estimation, the ReDDE system was used as it employs a well motivated collection ranking algorithm, whereby collection size estimation is a crucial feature.

$$R_k = \frac{\sum_{i=1}^{k} E_i}{\sum_{i=1}^{k} B_i} \quad (3)$$

where B is a baseline ranking and E is the collection selection algorithm ranking. B_i and E_i are the number of relevant documents counted for position i in the ranking. The larger the value R at position k, the better the ranking method E.

The top 20 collections for each testbed were selected. Figure 3 shows the results of the collection selection on the four testbeds. The IntelliOnto system performed best on the trec123-2ldb-60col representative collection, and worst on the trec123-100col-bysource collection. We believe that the reason for this differential in performance is due to collection size; the IntelliOnto method appears to perform better on larger collections. Due to the preliminary nature of this work, we can only speculate as to why this is the case. One possible reason is that the probability of encountering subject classification terms is higher in larger collections thus allowing them to be ranked more effectively. The favourable comparison with ReDDE on the larger collections does lend some support to our hypothesis that effective collection selection is possible without using estimates of collection size. Larger experiments and a more detailed analysis will be needed to bear this out.

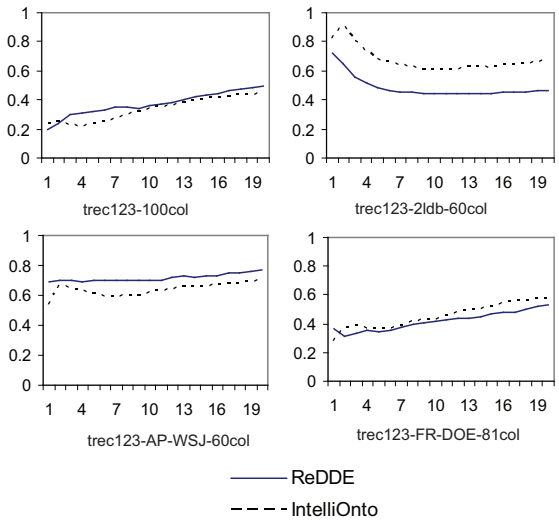

Figure 3: Collection selection accuracy on the testbeds.

7 Summary and Conclusions

This article tackles the collection selection problem from the query side. Queries are enhanced by mapping them to subjects in an ontology; the associated subject classification terms are then employed to retrieve collections. A novel form of ontology based collection selection, IntelliOnto, is introduced. This method was compared to ReDDE, the current state-of-the-art collection selection method. In preliminary experiments, the IntelliOnto method provided encouraging performance on larger collections. Although the research is preliminary, there is some support to the hypothesis that the ontology-based approach mitigates the need for collection size estimates in collection selection. In further work we will fully populate the ontology, and bring in collection size estimates. Work will also focus on using subjects deeper in the ontology with the goal of improving precision. We will also experiment to find how many DDC codes from the search to use for best results, and how many query probe terms to use for best results.

References

[1] B. Billerbeck, A. Cannane, A. Chatteraj, N. Lester, W. Webber, H. E. Williams, J. Yiannis and J. Zo-

bel. RMIT University at TREC 2004. In E. M. Voorhees and L. P. Buckland (editors), *Proceedings Text Retrieval Conference (TREC)*, Gaithersburg, MD, November 2004. National Institute of Standards and Technology Special Publication 500-261.

[2] P. Buitelaar. *CoreLex: Systematic Polysemy and Underspecification*. Ph.D. thesis, Computer Science Department, Brandeis University, 1998.

[3] French J.C. Powell A.L. Callan, J. and M. Connell. The effects of query-based sampling on automatic database selection algorithms. In *Technical Report CMU-LTI-00-162*, Carnegie Mellon University, 2000. Langauage Technologies Institute, School of Computer Science.

[4] J. P. Callan, Z. Lu and W. Bruce Croft. Searching Distributed Collections with Inference Networks . In E. A. Fox, P. Ingwersen and R. Fidel (editors), *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–28, Seattle, Washington, 1995. ACM Press.

[5] Jamie Callan, Margaret Connell and Aiqun Du. Automatic discovery of language models for text databases. In *Proceedings of the 1999 ACM SIGMOD international conference on Management of data*, pages 479–490. ACM Press, 1999.

[6] Nicholas Eric Craswell. *Methods for Distributed Information Retrieval*. Ph.D. thesis, The Australian National University, 2001.

[7] Nick Craswell, Peter Bailey and David Hawking. Server selection on the world wide web. In *Proceedings of the fifth ACM conference on Digital libraries, San Antonio, Texas, United States*, pages 37–46. ACM Press, 2000.

[8] Daryl J. D'Souza, James A. Thom and Justin Zobel. A comparison of techniques for selecting text collections. In *Proceedings of the 11th Australasian Database Conference(ADC'2000)*, pages 28–32, Canberra, Australia, 2000.

[9] F. Esposito, S. Ferelli, N. Fanizzi and G. Semeraro. Learning from parsed sentences with INTHELEX. In Claire Cardie, Walter Daelemans, Claire Nédellec and Erik Tjong Kim Sang (editors), *Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop, Lisbon, 2000*, pages 194–198. Association for Computational Linguistics, Somerset, New Jersey, 2000.

[10] D. Faure and C. N'edellec. A corpus-based conceptual clustering method for verb frames and ontology acquisition. In *In LREC workshop on Adapting lexical and corpus resources to sublanguages and applications, Granada, Spain*, 1998.

[11] James C. French, Allison L. Powell, James P. Callan, Charles L. Viles, Travis Emmitt, Kevin J. Prey and Yun Mou. Comparing the performance of database selection algorithms. In *Research and Development in Information Retrieval*, pages 238–245, 1999.

[12] Fabien Gandon. Agents handling annotation distribution in a corporate semantic web. *Web Intelligence and Agent Systems, IOS Press*, Volume 1, Number 1, pages 23–46, 2003.

[13] Luis Gravano and Héctor García-Molina. Generalizing GlOSS to vector-space databases and broker hierarchies. In *International Conference on Very Large Databases, VLDB*, pages 78–89, 1995.

[14] Luis Gravano, Héctor García-Molina and Anthony Tomasic. GlOSS: text-source discovery over the Internet. *ACM Transactions on Database Systems*, Volume 24, Number 2, pages 229–264, 1999.

[15] David Hawking and Paul Thistlewaite. Methods for information server selection. *ACM Transactions on Information Systems (TOIS)*, Volume 17, Number 1, pages 40–76, 1999.

[16] David Hawking and Paul Thomas. Server selection methods in hybrid portal search. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 75–82, New York, NY, USA, 2005. ACM Press.

[17] Xiaohui Tao Richi Nayak John D. King, Yuefeng Li. Mining world knowledge for analysis of search engine content. *Web Intelligence and Agent Systems: An International Journal*, October 2007. Accepted for publication in September 2006.

[18] Joerg-Uwe Kietz, Alexander Maedche and Raphael Volz. A method for semi-automatic ontology acquisition from a corporate intranet. In *Proceedings of EKAW-2000 Workshop "Ontologies and Text", Juan-Les-Pins, France, October 2000*, number 1937 in Springer Lecture Notes in Artificial Intelligence (LNAI), 2000.

[19] John D. King. Deep web collection selection. Master's thesis, School of Software Engineering, Queensland University of Technology, 2003.

[20] John D King. Large scale analysis of search engine content. In *The Fourth International Conference on Active Media Technology, Brisbane, Australia*, Volume 1, page 451 to 453, 2006.

[21] John D. King and Yuefeng Li. Web based collection selection using singular value decomposition. In *IEEE/WIC International Conference on Web Intelligence (WI'03)*, pages 104–110, Halifax, Canada, 2003.

[22] Y. Li and N. Zhong. Capturing evolving patterns for ontology-based web mining. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 256–263, Beijing, China, 2004.

[23] Y. Li and N. Zhong. Mining ontology for automatically acquiring web user information needs. *IEEE Transactions on Knowledge and Data Engineering*, Volume 18, Number 4, pages 554–568, 2006.

[24] King-Lup Liu, Clement T. Yu and Weiyi Meng. Discovering the representative of a search engine. In *CIKM*, pages 652–654, 2002.

[25] Z. Lu, J.P. Callan and W.B. Croft. Applying inference networks to multiple collection searching. Technical Report TR96–42, University of Massachusetts at Amherst. Department of Computer Science, 1996.

[26] A Maedche and S Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, Volume 16(2), pages 72–79, 2001.

[27] Alexander Maedche and Steffen Staab. Discovering conceptual relations from text. In W. Horn (editor), *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI)*, pages 321–325, 2000.

[28] Alexander Maedche and Steffen Staab. Learning ontologies for the semantic web. In *SemWeb*, 2001.

[29] Weiyi Meng, King-Lup Liu, Clement T. Yu, Wensheng Wu and Naphtali Rishe. Estimating the usefulness of search engines. *15th International Conference on Data Engineering (ICDE'99)*, Volume 1, pages 146–153, 1999.

[30] Milad Shokouhi, Justin Zobel, Falk Scholer and S. M. M. Tahaghoghi. Capturing collection size for distributed non-cooperative retrieval. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 316–323, New York, NY, USA, 2006. ACM Press.

[31] L. Si and J. Callan. Relevant document distribution estimation method for resource selection, 2003.

[32] Luo Si and Jamie Callan. Modeling search engine effectiveness for federated search. In *SIGIR*, pages 83–90, 2005.

[33] Nenad Stojanovic. Information-need driven query refinement. *Web Intelligence and Agent Systems, IOS Press*, Volume 3, Number 3, pages 155–170, 2005.

[34] H. Suryanto and P. Compton. Learning classification taxonomies from a classification knowledge based system. In C. Nedellec P. Wiemer-Hastings S. Staab, A. Maedche (editor), *Proceedings of the Workshop on Ontology Learning, 14 Conference on Artificial Intelligence (ECAI'00)*, Berlin, 2000. Conference on Artificial Intelligence (ECAI'00).

[35] Jason Chaffee Susan Gauch and Alexander Pretschner. Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems, IOS Press*, Volume 1, Number 3, pages 219–234, 2003.

[36] Xiaohui Tao, John D King and Yuefeng Li. Information fusion with subject-based information gathering method for intelligent multi-agent models. In *The Seventh International Conference on Information Integration and Web-Based Applications and Services, Kuala Lumpur, Malaysia*, Volume 2, page 861 to 869. iiWAS, 2005.