Domain Specific Knowledge-based Information Retrieval Model using Knowledge Reduction

Changwoo Yoon and Douglas D. Dankel II

Computer and Information Science and Engineering, University of Florida
E301 CSE, C.I.S.E., PO Box 116120, Gainesville, FL 32611-6120
cwyoon@cise.ufl.edu, ddd@cise.ufl.edu

Abstract[1]

Information is a meaningful collection of data. Information retrieval (IR) is an important tool for changing data to information. Of the three classical IR models (Boolean, Support Vector Machine, and Probabilistic), the Support Vector Machine (SVM) IR model is most widely used. But this model does not convey enough relevancies between a query and documents to produce effective results reflecting knowledge.

To augment the IR process with knowledge, several techniques are proposed including query expansion by using a thesaurus, a term relationship measurement like Latent Semantic Indexing (LSI), and a probabilistic inference engine using Bayesian Networks.

Our research aims to create an information retrieval model that incorporates domain specific knowledge to provide knowledgeable answers to users. We use a knowledge-based model to represent domain specific knowledge. Unlike other knowledge-based IR models, our model converts domain-specific knowledge to a relationship of terms represented as quantitative values, which gives improved efficiency.

Introduction

Conceptually, information retrieval (IR) is the process of changing data to information. More technically, information retrieval is the process of determining the relevant documents from a collection of documents, based on a query presented by the user [Erica, 2004].

We can define data as un-interpreted *signals* or raw observations that reach our senses. When we provide meaning to these data, they become information. Information is more meaningful and useful to humans than raw data. Information retrieval is the process that extracts information from data.

Two commonly used information retrieval models are the Boolean search model and the vector model. In the Boolean search model, we specify a set of query words. These query words are compared to the words in documents to retrieve those documents that precisely contain the given set of query words. We can call the retrieved documents "*information*" but it is hard to call them "*knowledge*," because additional tasks such as browsing each document and selecting more meaningful ones are required to transform the retrieved documents to some form of knowledge. Knowledge is organized information.

The classic vector information retrieval model is an attempt to infuse knowledge to information retrieval results using the frequency of the query terms that are found in documents. Intelligent information retrieval or semantic information retrieval attempts to use some form of knowledge representation within the IR model to obtain more organized information that is knowledge. But, it is difficult to codify or regulate the knowledge. An ontology is the attempt to regulate the knowledge and the specification of a conceptualization [Gruber, 1993]. We are using an ontology such as a knowledge representation or semantic web [Berners, 2001], which is the abstract representation of data on the World Wide Web. It is an attempt to make the semantics of a body of knowledge more explicit [Lee, 2003].

We can classify an ontology as either general domain or closed domain. For example, WordNet [Miller, 1990] is an example of a general ontology, consisting of a thesauri and a taxonomy, which aims to represent general domain documents written in natural language. Compared to the general-domain, a closed-domain generally has its own knowledge repository such as a term dictionary and relations that exist between terms. Good examples of such a repository are the medical field's Unified Medical Language System (UMLS) and Systematized Nomenclature of Medicine (SNOMED). We call these *domain specific knowledge*.

In this paper, we propose an information retrieval model that incorporates domain specific knowledge to provide knowledge infused answers to users.

Intelligence in Information Retrieval Model

In this section we describe the intelligence in the information retrieval models proposed so far.

Vector Space Model

The vector space model of information retrieval takes a geometrical approach. A vector, called the 'document vector', represents each document. This vector is of identical length for all documents with the length equaling the number of terms in the entire documents. The vector space model was first introduced by Gerard Salton [Salton et al., 1975].

Salton [Salton et al., 1975] defined 'term weight,' which is also known as the 'importance weight.' The 'term weight' measures the ability of a term to differentiate one document having a term with other documents having the same term.

There are a number of weighting schemes that can be used within the vector space model. Salton uses two properties: term frequency and inverse document frequency. The term frequency (*tf*) is the intra document importance, which is the occurring frequency of the term within a document. The term frequency measures how well that term describes the document content. A term having a higher term frequency is more important than a term having a lower frequency. The inverse document frequency (*idf*) is the number of documents in the corpus within which the term occurs or the inter-document importance. If a term is uniformly present across the entire system, the term is less capable of differentiating the documents, which means that it has less importance than a term having a small global weight.

After the document and query vectors have been constructed using the weighting scheme, there are various ways to calculate the similarity coefficient. One of the best known is the cosine measure [Salton, 1968], defined for query vector $q = (q_1, q_2, \cdots, q_t)$ and document vector $d_j = (w_{1,j}, w_{2,j}, \cdots, w_{t,j})$ where *t* is number of terms. The cosine similarity measures the angle between the query and document vectors in n-dimensional Euclidean space.

Latent Semantic Indexing (LSI)

The classical information retrieval models use index terms as querying tools. The selection of index terms is based on the assumption that the terms represent the 'user's need,' that is they represent the concept of the user's query intention. But as the search results show, index terms do not really contribute to the concepts of information retrieval. For example, if the user wants to search about 'Major cities in Florida,' the index terms used may be 'Major,' 'city,' and 'Florida.' The search engine may try to find documents containing these keywords. But if there is an intelligent search engine supporting the conceptual matching, it would try to search for keywords such as 'Tampa,' 'Orlando,' and 'Miami' in the same way as human do.

The main idea of Latent Semantic Indexing (LSI) comes from the fact that a document may have words having similar concepts. So LSI considers documents that have many words in common to be semantically close and vice versa [Furnas, 1988]. From the example of previous paragraph , if the words 'major,' 'city,' 'Florida,' 'Tampa,' 'Orlando,' and 'Miami' appears together in enough documents, the LSI algorithm will conclude those terms are semantically close, then return all documents containing terms 'Tampa,' 'Orlando,' and 'Miami' even when those terms are not given as index terms.

The most important point of the LSI algorithm is that all calculations are carried out automatically by only looking at the document collection and index terms. Solving problems like 'Polysemy' (i.e., words having more than one meaning) and 'Synonymy' (i.e., there are many ways of describing the same object) can be performed efficiently without the aid of a thesaurus.

LSI generally uses a statistical method called Singular Value Decomposition (SVD) to uncover the word associations between documents. The effect of SVD is to move words and documents that are closely associated nearer to one another in the projected space. It is possible for an LSI based system to locate terms that do not even appear in a document. Documents that are located in a similar part of the concept space are retrieved, rather than only matching keywords.

Query expansion

Whenever a user wants to retrieve a set of documents, he starts by building a concept for which he is looking. Such a conceptualization is called the 'information need.' Given the 'information need,' the user must formulate a query that is adequate to the information retrieval system. Usually, the query is a collection of index terms, which might be erroneous and improper initially. In this case, the reformulation of the query should be done to obtain a desired result. We call the reformulation process query expansion.

One of the simplest techniques involves the use of a thesaurus to find synonyms for some or all of the terms in the query. These synonyms are added to the query to broaden the search. The thesaurus used can be manually generated in the specific domain, such as medical domains. But for a general domain like the Web, it is hard to generate such knowledge-base-like thesauri because the documents in the general domain are comparably new, large, and dynamically changing.

Various algorithms have been suggested for generating thesauri automatically. For example, Crouch & Yang [Couch, 1992] suggest a method based on clustering and term discrimination value theory.

Another widely used method of query expansion is the use of relevance feedback. This involves the user performing a preliminary search, then examining the documents returned and deciding which are relevant. Finally, terms from these documents are added to the query and the search is repeated. This obviously requires human intervention and, as a result, is inappropriate in many situations. However, there is a similar approach, sometimes called pseudo-relevance feedback, in which the top few documents from an initial query are assumed relevant and used for automatic feedback [Mitra, 1998].

Using Phrase

Many information retrieval systems are based on the vector space model (VSM), which represents a document as a vector of index terms. The classical VSM uses a word as an index term. To improve retrieval accuracy, it is natural to replace word stems with concepts. For example, replacing a word stem with an ULMS code, if the document domain is the medical domain, is a possible way to include the concept in information retrieval. However, previous research showed not only no improvements, but a degradation in retrieval accuracy when concepts were used in document retrieval [Voorhees, 1993].

Replacing word stems with multiple word combinations was also studied. One of the studies used phrases (i.e., a string of words used to represent concepts) as the indexing terms [Wenlei, 2002]. The similarity between two phrases is jointly determined by their conceptual similarity and their common word stems, which increases of retrieval accuracy compared to the classical SVM model.

Separating the importance of weighting in the SVM model is suggested [Shuang, 2004]. Shuang et al. considered phrases to have more importance than individual terms in information retrieval. They used two separated similarity measures between documents and queries like (*phrase-sim, term-sim*), where *phrase-sim* is the similarity obtained by matching the phrases of the query against documents and *term-sim* is the usual similarity measure used in the SVM model. Documents are ranked in descending order of (*phrase-sim, term-sim*) where *phrase-sim* has a higher priority.

Using WordNet

WordNet is an electronic lexical database developed at Princeton University beginning in 1985 [Miller, 1990]. WordNet 2.0 has over 130,000 word forms. It is widely used in natural language processing, artificial intelligence, and information technology such as information retrieval, document classification, question-answer systems, language generation, and machine translation.

The basic building blocks of WordNet are synonym sets (‘synsets’), which are unordered sets of distinct word forms and which correspond closely to what are called ‘concepts.’ Examples of synsets are {car, automobile} or {shut, close}. WordNet 2.0 contains some 115,000 synsets. There are two kinds of relations in WordNet: semantic and lexical relations. Examples of semantic relations are ‘is-a,’ ‘part-of,’ ‘cause,’ etc. Nouns and verbs are hierarchically organized from the top generic concepts to the bottom specific concepts by an ‘is-a’ semantic relation. Examples of lexical relations are synonymy and antonymy.

A recent study on word sense disambiguation in information retrieval [Kim, 2004] shows the possibility of improving IR performance using WordNet knowledge. They proposed a root sense tagging approach. They noticed that the tradition method described in the previous paragraph used a *fine-grained disambiguation* for IR tasks. For example, the word ‘stock’ has 17 different senses in WordNet, which are used in word sense disambiguation. These include ‘act,’ ‘animal,’ ‘artifact,’ ‘attribute,’ ‘body,’ etc. Using these classifications when performing word sense disambiguation (i.e., *coarse-grained disambiguation*) showed an improvement of retrieval effectiveness.

Knowledge-based Information Retrieval Model

This research aims to develop a knowledge base information retrieval model in a closed domain using domain-specific knowledge base. Figure 1 is the architecture of proposed model.

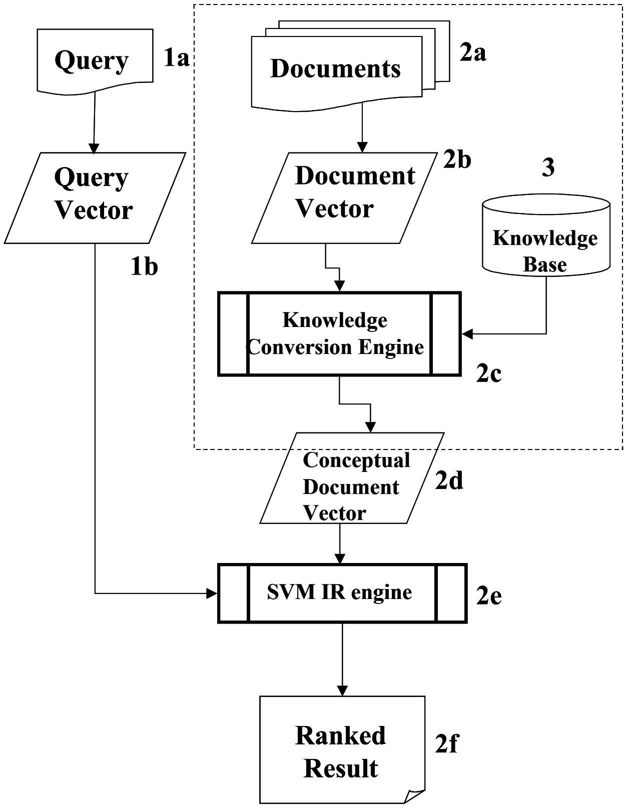
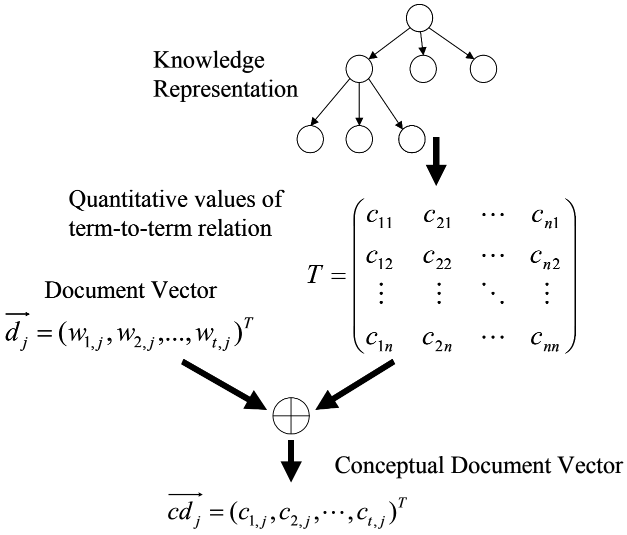


Figure 1. Architecture of the knowledge-based information retrieval model

The overall operation of the proposed model is as follows. A classical vector space model (VSM) information retrieval model using term frequency and inverse term frequency creates the query vector (1b) from a user query (1a) and document vector (2b) from documents repository (2a). The Knowledge Conversion Engine (KCE, 2c) applies the knowledge (semantics) of the Knowledge Base (3) to the Document Vector (2b) to make the Conceptual Document Vector (2d). The conventional VSM IR engine (2e) calculates the relevance between the query vector (1b) and the conceptual document vector (2d) resulting in a ranked document list (2f).

Knowledge Reduction

Our model's main idea is knowledge reduction. Unlike other knowledge-based information retrieval models, this model reduces knowledge represented by the knowledge-base to a quantitative value, which gives a more efficient performance gain compared with the conventional methods. From Figure 1, the applying of knowledge to information retrieval model shown in dotted box is not involved in user's query processes (1a, 1b, 2d, 2f, and 2f).
Figure 2 is a diagram of the knowledge reduction process. The knowledge is converted into a conceptual document vector by combining it with the VSM's document vector. The definitions of the document vector follows



Knowledge Representation
Quantitative values of term-to-term relation
Document Vector
$\vec{d_j} = (w_{1,j}, w_{2,j}, ..., w_{t,j})^T$
$T = \begin{pmatrix} c_{11} & c_{21} & \cdots & c_{n1} \\ c_{12} & c_{22} & \cdots & c_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1n} & c_{2n} & \cdots & c_{nn} \end{pmatrix}$
Conceptual Document Vector
$\vec{cd_j} = (c_{1,j}, c_{2,j}, \cdots, c_{t,j})^T$

Figure 2. Knowledge reduction

Definition 1: A *document vector for a document* d_j *is*
$\vec{d_j} = (w_{1,j}, w_{2,j}, ..., w_{t,j})^T$ *where*
$w_{i,j} \geq 0$ *is a weight associated with the pair* $(k_i\ d_j)$ *where*
k_i *is an index term,* d_j *is a document,and t is the number of index terms in the whole system.*

Definition 2: The *set of all index terms K is*
$K = \{k_1, ..., k_t\}$ *where*
t is the number of index terms in the whole system.

Normally the index terms are words contained in the document. The set is usually confined to only the significant words by eliminating common functional words called stopwords. The VSM uses the term frequency and the inverse term frequency as a weighting scheme associated with the document.

Definition 3: The *weight* $w_{i,j} \geq 0$ *is*
$w_{i,j} = tf_{i,j} \times idf_i$ *where*
$tf_{i,j}$ *is the term frequency of term i in document j and*
$idf_i = \log \frac{N}{n_i}$ *(the inverse document frequency)*
where
N is the number of documents in the collection and
n_i *is the document frequency of term i*

The document frequency is the number of documents in which the term occurs.

Conceptual Document Vector

In the Vector Space Model, term vectors are pair-wise orthogonal meaning that terms are assumed to be independent. There was an attempt to incorporate term dependencies, which gives semantically rich retrieval results [Holger, 2004]. They used a term context vector to reflect the influence of terms in the conceptual description of other terms. The definition of a term context vector follows:

Definition 4: *The set of term context vectors T is*

$$T = \begin{pmatrix} c_{11} & c_{21} & \cdots & c_{n1} \\ c_{12} & c_{22} & \cdots & c_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1n} & c_{2n} & \cdots & c_{nn} \end{pmatrix} \text{ where}$$

n is number of terms and
c_{ik} *represents the influence of term* t_k *on term* t_i.

Definition 5: *The term context vector* $\vec{t_i}$ *is the ith column of matrix T where*
$\vec{t_i} = (c_{i1}, c_{i2}, \cdots, c_{in})^T$ *where*
n is number of terms and
c_{ik} *represents the influence of term* t_k *on term* t_i.

By converting domain-specific knowledge to the term-relation matrix T defined in Definition 4, we can transform each initial document vector $\vec{d_j} = (w_{1,j}, w_{2,j}, ..., w_{t,j})^T$ into a conceptual document vector $\vec{cd_i} = (c_{1,j}, c_{2,j}, \cdots, c_{t,j})^T$ using the equation in Definition 6 [Holger, 2004, p. 240].

Definition 6: $\overrightarrow{cd}_i$ *from* $\vec{d}_i$ *(Definition 1) and* $\vec{t}_i$ *(Definition 5) is*

$$\overrightarrow{cd}_i = \frac{\sum_{j=1}^{n} w_{ij} \frac{\vec{t}_j}{\left|\vec{t}_j\right|}}{\sum_{j=1}^{n} w_{ij}} \text{ where}$$

$\vec{t}_i$ *is the term context vector of term* t_j *and* $\left|\vec{t}_j\right|$ is the length of vector $\vec{t}_j$.

The division of the elements in the term context vectors by the length of the vector is a normalization step.

The Knowledge Conversion Engine (KCE) converts relationships within the knowledge base into a term context vector. In the following, we discuss how the elements of matrix T can be obtained from domain-specific knowledge base representation.

Knowledge Conversion Process

The Knowledge Conversion Engine (KCE) converts the Support Vector Machine (SVM) document vector to a conceptual document vector reflecting the knowledge of the knowledge representation. We use a semantic network as the knowledge representation model. To make the discussion more general and simpler, we show the conversion of two types of relationships exist within the semantic network.

The first type is the hierarchical topology relationship shown in Figure 3. In this type, each node has attributes denoting its characteristics on the hierarchical tree.

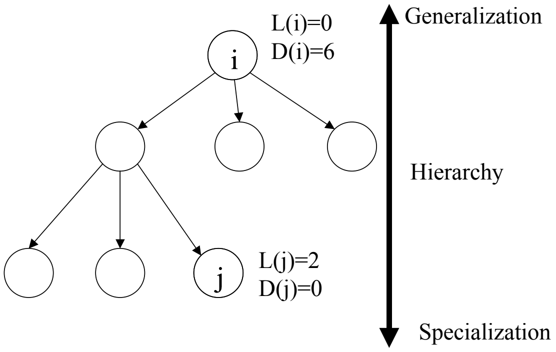


Figure 3. Semantic network's topology

L(i) is the level of term i in a knowledge tree. D(i) is the number of descendents of term node i in the tree. The term influence between i and j is inversely proportional to the distance, which is the difference of the levels. Having many descendents means that a node is a more general term than some node having a smaller number of descendents. So term influence is inversely proportional to the number of descendents. Thus we can calculate the hierarchical topology relationship between two terms i and j:

Definition 7: c_{ij} *from the SNN-KB hierarchical topology is*

$$c_{ij} = C(Sht) \times \frac{1}{d(i,j)} \times \log \frac{1}{D(i)+D(j)} \text{ , where}$$

$C(Sht)$ *is the coefficient for the SNOMED hierarchical topology relation and*
$d(i,j) = \left|L(i) - L(j)\right|$ *where*
L(i) is level of node i and L(j) is level of node j and
D(i) is number of descendents of node i and
D(j) is number of descendents of node j.

A second type is the synonymous relationship, which represents an 'is-a' relation shown in Figure 4.

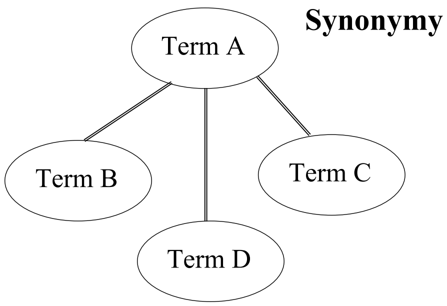


Figure 4. Semantic network's 'is-a' relation

For the synonym relations:

Definition 8: c_{ij} *from the SNN-KB synonym relation is*
$c_{ij} = C(Ss)$ *where*
C(Ss) is the coefficient for the SNOMED synonym relationship.

The value of *C(Sht), C(Ss),* and *C(Sm)* should be optimized by simulation.

SVM IR Engine: Document Retrieval

After the process of converting the document vector to the conceptual document vector, the system can start accepting queries. A query is expressed identical to a document vector where the query terms are the vector elements. The query vector q is compared with the conceptual document vector cd_i using the cosine similarity measure.

Definition 9: The *similarity* between $\vec{q}$ and $\overrightarrow{cd}_i$ *is*

$$\cos(\vec{q},\overrightarrow{cd_i}) = \frac{\vec{q} \bullet \overrightarrow{cd_i}}{\left|\vec{q}\right|\left|\overrightarrow{cd_i}\right|} = \frac{\sum_{i=1}^{n} q_i cd_i}{\sqrt{\sum_{i=1}^{n} q_i^2 \cdot \sum_{i=1}^{n} cd_i^2}}$$

The similarity measure produces a ranked list of relevant documents related to the query.

Conclusion

This information retrieval model is a knowledge-based information retrieval model. Unlike other models, which performing ontology level information retrieval tasks such as ontology comparison and ontological query expansion, the proposed model reduces the knowledge level represented by the knowledge base to a statistical model such as the vector space model's document vector.

Using knowledge reduction enables the off-line processing of the application (calculation) of knowledge to information retrieval procedure. Because only the conceptual document vector, which can be obtained from document vector and knowledge-base, is involved in the on-line process of producing ranked results by comparing a user's query and documents.

Even if the proposed model uses domain-specific knowledge, this model can be used in an open-domain application if some types of knowledge bases are supported. The possible candidate for the open domain knowledge base is WordNet, which has a thesaurus and relations from the natural language domain.

We defined some examples of knowledge reduction methods using a semantic network. The semantic network is an example of knowledge representation, which is an artificial intelligence's field handling ontology. Our model has flexibility on the type of knowledge representation if we can define the knowledge reduction scheme of the selected knowledge representation model.

References

Berners-Lee, T., Hendler, J., and Lassila, O., The Semantic Web*, Scientific American*, 284(5):34-43, May 2001.

Chai, J. Y., and Biermann, A., The Use of Lexical Semantics in Information Extraction, *Workshop in Automatic Information Extraction and Building of Lexical Semantic Resources*, 61-70, 1997

Crouch, C. J., and Yang, B., Experiments in Automatic Statistical Thesaurus Contruction, *ACM SIGIR*'92, 77-88, 1992.

Erica Brown, Terms and Definitions, *URL: http://www.geocities.com/ejb_wd/ Definitions.html*

Furnas, G. W., Deerwester, S., Dumais, S. T., Landauer, T. K., Harshman, R. A., Streeter, L. A., and Lochbaum, K. E., Information Retrieval using a Singular Value Ddecomposition Model of Latent Semantic Structure, *ACM SIGIR '88*, 465-480, 1988

Gruber, T. R., A Translation Approach to Portable Ontologies, *Knowledge Acquisition*, 5(2):199-200, 1993

Holger Billhardt, Daniel B., Victor M., A Context Vector Model for Information Retrieval, *J. of Am. Soc. for Information Science*, 53(3):236-249, 2002

Kim, S. B., Seo, H. C., and Rim, H. C., Information Retrieval using Word Senses: Root Sense Tagging Approach, *SIGIR'04*, 258-265, 2004

Lee Iverson, A Digital Library Research Agenda, *URL: http://www.ece.ubc.ca /~leei/weblog/Access2003.pdf*

Miller, G. A., Wordnet: An On-Line Lexical Database, *International Journal of Lexicography*, 3(4):235-312, 1990

Mitra, M., Singhal, A. and Buckley, C., Improving Automatic Query Expansion, *ACM SIGIR*, 206-214, 1998.

Rila Mandala, T. Takenobu, and T. Hozumi, The Use of WordNet in Information Retrieval, *COLING-ACL'98*, 31-37, 1998.

Salton, G., Yang, C. S., and Yu, C. T., A Theory of Term Importance in Automatic Text Aanalysis, *Journal of the American Society for Information Sciences*, 26(1):33-44, 1975

Sanderson, M., Retrieving with Good Sense, *Information Retrieval*, 2(1):49-69, 2000.

Shuang, L., Fang Liu, Clement Yu, An Effective Approach to Document Retrieval via Utilizing WordNet and Recognizing Phrases, *ACM SIGIR'04*, 266-272, 2004.

Voorhees, E., Using WordNet to Disambiguate Word Sense for Text Retrieval, *ACM SIGIR*, 171-180, 1993.

Voorhees, E. Query Expansion using Lexical-Semantic Rrelations, *ACM SIGIR*, 61-69, 1994.

Wenlei Mao, and Wesley W. C., Free-text Medical Document Retrieval Via Phrase-based Vector Space Model, *AMIA'02*, 489-493, 2002.