

2711.3150.8

**Université de Montréal**

V.014  
11488964

**Using Domain-Specific Knowledge to Improve  
Information Retrieval Performance**

Par

Li-Fang Liu

Faculté des arts et des sciences  
Département d'informatique et de recherche opérationnelle

Mémoire présenté à la Faculté des études supérieures  
en vue de l'obtention du grade de Maîtrise  
en informatique et recherche opérationnelle

August, 2003

©, Li-Fang Liu, 2003



QA  
76  
U54  
2004  
v.014



**Direction des bibliothèques**

**AVIS**

L'auteur a autorisé l'Université de Montréal à reproduire et diffuser, en totalité ou en partie, par quelque moyen que ce soit et sur quelque support que ce soit, et exclusivement à des fins non lucratives d'enseignement et de recherche, des copies de ce mémoire ou de cette thèse.

L'auteur et les coauteurs le cas échéant conservent la propriété du droit d'auteur et des droits moraux qui protègent ce document. Ni la thèse ou le mémoire, ni des extraits substantiels de ce document, ne doivent être imprimés ou autrement reproduits sans l'autorisation de l'auteur.

Afin de se conformer à la Loi canadienne sur la protection des renseignements personnels, quelques formulaires secondaires, coordonnées ou signatures intégrées au texte ont pu être enlevés de ce document. Bien que cela ait pu affecter la pagination, il n'y a aucun contenu manquant.

**NOTICE**

The author of this thesis or dissertation has granted a nonexclusive license allowing Université de Montréal to reproduce and publish the document, in part or in whole, and in any format, solely for noncommercial educational and research purposes.

The author and co-authors if applicable retain copyright ownership and moral rights in this document. Neither the whole thesis or dissertation, nor substantial extracts from it, may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms, contact information or signatures may have been removed from the document. While this may affect the document page count, it does not represent any loss of content from the document.

Université de Montréal  
Faculté des études supérieures

Ce Mémoire intitulé :  
“Using domain-specific knowledge to  
improve information retrieval performance”

présenté par:  
Li-Fang Liu

a été évalué par un jury composé des personnes suivantes:

Esma Aïmeur

Max Mignotte

Jian-Yun Nie (directeur de recherche)

Mémoire accepté le: 23 Décembre, 2003

## Abstract

Textual information is becoming increasingly available in electronic forms. Users need tools to sift through non-relevant information and retrieve only those pieces relevant to their needs. The traditional methods such as keyword-based search have somehow reached their limitations. An emerging trend is to combine the traditional information retrieval (IR) and artificial intelligence techniques, for example, knowledge representation and knowledge organization systems to improve IR effectiveness. This thesis explores the possibility of extending traditional information retrieval systems with knowledge-based approaches to improve the retrieval performance. Domain-specific knowledge bases such as the Canadian Thesaurus of Construction Science and Technology, and Canadian Building Digest are used in this project. The retrieval process incorporates the domain knowledge to find domain-specific information on the Web. In our case, the system is applied to the construction area. Experiments are also conducted using different search strategies. Our results show that an increase in retrieval performance can be obtained using certain knowledge-based approaches.

**Keywords:** information retrieval, domain-specific knowledge, thesaurus, construction.

## Résumé

L'information textuelle devient de plus en plus disponible sous formes électroniques. Les utilisateurs ont besoin d'outils pour filtrer l'information non-appropriée et pour retrouver seulement les éléments répondant à leurs besoins. Les méthodes traditionnelles telles que la recherche sur les mots-clés ont atteint quelque part leurs limites. Une tendance naissante est de combiner les techniques de recherche d'information (RI) traditionnelle et d'intelligence artificielle, par exemple, la représentation de la connaissance et un système d'organisation de la connaissance, afin d'améliorer la performance de la RI. Ce mémoire explore la possibilité de prolonger les systèmes traditionnels de recherche documentaire avec des approches basées sur la connaissance pour améliorer l'exécution de récupération. Les bases de connaissance spécifiques à un domaine comme le Thesaurus Canadien de la Science de Construction et Technologie, et le sommaire canadien de bâtiment sont employés dans ce projet. Le processus de recherche intègre les connaissances du domaine pour trouver des informations du domaine sur le Web, Dans notre cas, le système est utilisé pour le domaine de construction. Des expériences sont entreprises en utilisant différentes stratégies de recherche. Nos résultats prouvent qu'une augmentation de performance de recherche peut être obtenue en utilisant des approches basées sur la connaissance.

**Mots clés :** recherche d'information, connaissance du domaine, thesaurus, construction.

# Contents

<b>ABSTRACT</b> .....	<b>III</b>
<b>RÉSUMÉ</b> .....	<b>IV</b>
<b>CONTENTS</b> .....	<b>V</b>
<b>LIST OF TABLES</b> .....	<b>VII</b>
<b>LIST OF FIGURES</b> .....	<b>VIII</b>
<b>ACKNOWLEDGMENTS</b> .....	<b>IX</b>
<b>INTRODUCTION</b> .....	<b>10</b>
1.1 PREVIOUS APPROACHES.....	10
1.2 OUR APPROACH AND RESULTS.....	11
1.3 ORGANIZATION OF THE THESIS .....	12
<b>LITERATURE REVIEW</b> .....	<b>14</b>
2.1 IR.....	14
2.2.1 <i>Indexing process</i> .....	15
2.1.2 <i>Retrieval process</i> .....	17
2.1.3 <i>Discussion</i> .....	18
2.2 USING AI TECHNIQUES IN IR .....	20
2.2.1 <i>Ontology, thesaurus and IR</i> .....	22
2.2.1.1 <i>Ontology and thesaurus</i> .....	22
2.2.1.2 <i>Utilization in IR</i> .....	25
2.2.2 <i>CG and IR</i> .....	29
2.3 DISCUSSION.....	30
2.3.1 <i>Difficulty of forming a complete CG</i> .....	30
2.3.2 <i>Utilization of word co-occurrence information</i> .....	31
2.3.3 <i>Our approach</i> .....	32
<b>USING DOMAIN-SPECIFIC KNOWLEDGE TO IMPROVE IR PERFORMANCE</b> .....	<b>34</b>
3.1 INTRODUCTION.....	34
3.2 SEMANTIC INTERPRETATION MODULE .....	35
3.2.1 <i>Domain knowledge Source</i> .....	36
3.2.1.1 <i>The TC/CS thesaurus</i> .....	36
3.2.1.2 <i>Definition of semantic concepts</i> .....	41

3.2.1.3 Algorithm for semantic tagging.....	43
3.2.2 <i>Semantic representation</i> .....	44
3.2.2.1 CG .....	44
3.2.2.2 Simplified CG representation .....	47
3.3. RELEVANCE MEASUREMENT MODULE .....	51
3.3.1 <i>Training corpus</i> .....	51
3.3.2 <i>Reliability assessment</i> .....	52
3.4 FRAMEWORK OF OUR SYSTEM .....	53
<b>SYSTEM IMPLEMENTATION .....</b>	<b>56</b>
4.1 GENERAL DESCRIPTION OF THE SYSTEM .....	56
4.2 OKAPI IR SYSTEM.....	57
4.2.1 <i>Okapi system overview</i> .....	57
4.2.2 <i>Passages retrieval</i> .....	58
4.2.3 <i>BM25 formula</i> .....	60
4.3 PRE-PROCESSING .....	62
4.3.1 <i>Segmenting passages</i> .....	62
4.3.2 <i>Simple word stemming</i> .....	62
4.3.3 <i>Semantic interpretation</i> .....	63
4.4 RELEVANCE MEASUREMENT PROCESSING .....	68
4.4.1 <i>Entity measurement</i> .....	68
4.4.2 <i>Passage measurement</i> .....	69
<b>SYSTEM EVALUATION .....</b>	<b>72</b>
5.1 EVALUATION IN IR .....	72
5.1.1 <i>Traditional evaluation methods</i> .....	72
5.1.2 <i>Average precision and recall</i> .....	73
5.1.3 <i>Evaluation with test collections</i> .....	74
5.2 EVALUATION EXPERIMENT .....	75
5.3 DISCUSSION.....	78
<b>CONCLUSION.....</b>	<b>80</b>
6.1 PROJECT .....	80
6.2 BASIC APPROACH .....	80
6.3 RESULTS.....	81
6.2 FUTURE DIRECTIONS.....	81
<b>REFERENCES.....</b>	<b>84</b>
<b>APPENDIX.....</b>	<b>88</b>



## List of Tables

Table 1: an explanation of the relationships and their abbreviations used in TC/CS.....	38
Table 2: An example of descriptor in TC/CS for concept “metallic materials” .....	39
Table 3: A fragment of the result of the subjective evaluation.....	42
Table 4: Structure of the database.....	65
Table 5: Results-1 of system evaluation .....	77
Table 6: Results-2 of the system evaluation .....	78
Table 7: Results-3 of the system evaluation .....	78



## List of Figures

Figure 1: A fragment of the hierarchy tree for “NT” relationship in TC/CS.....	40
Figure 2: DF of CG for “John is going to Boston by bus”. .....	46
Figure 3: LF of CG for “John is going to Boston by bus”.....	46
Figure 4: CGIF for “John is going to Boston by bus”. .....	46
Figure 5: Framework of the system .....	54
Figure 6: Algorithm of the recursive function for semantic tagging .....	67




## Acknowledgments


The work presented in this thesis would not be possible without the help of many people. I would firstly like to thank Professor Jian-Yun Nie, my research supervisor, for his invaluable advices and ideas on the research. He helps me with everything from basic concepts to nuances of my design; his support and expertise point me in the right direction to solve several problems.

This work is part of a research project supported by the Bell-LUB. I would like to thank Bell-LUB for providing me with a scholarship.

I would also like to thank my main partners: Zhuo Zhang and Qi Zhang. Many parts of this thesis are results of long discussions and cooperation with them.



Finally, I would like to give a special thanks to my family and my friends for their incredible amount of support and encouragement. I thank them all for being a part of my life.



# Chapter 1

## Introduction

Information Retrieval (IR) systems are designed with the objective of providing references to documents that could contain the information desired by the user. In other words, the system is intended to identify which documents the users should read in order to satisfy their information requirements. With the rapid growth of on-line information, it is becoming increasingly difficult for users to find the information they need. The phenomenon of posing a query to a Web search engine and receiving many thousands of "hits", of which few are really relevant, is familiar to everyone. A well-known contributor to this problem is that search is organized around words (contained in the target documents) rather than the concepts which those words denote. As a word can denote many concepts (polysemy) and a concept can be denoted by many words (synonymy), a user's query may both miss relevant documents and hit irrelevant ones. The need for more precise IR is growing rapidly.

### 1.1 Previous approaches

Much of the current research has attempted the exploitation of a richer document context to extract concepts or knowledge that may improve the system's retrieval effectiveness. In these systems, the query and documents are encoded using special formalism (such as description logic, semantic nets, etc.), which is more expressive to accurately represent text's meaning. This will result in increased precision. If the formalism adopted for text encoding is powerful enough, we can then form and match arbitrary descriptions for the text, which could support a higher level of abstraction in information search. By providing a more understandable, semantics-rich concept space, the information search is performed within a concept space rather than within a word.

One of the approaches to this end is to conduct search in terms of concepts, rather than words. Instead of matching documents on a word-by-word basis using the words of the

query, the texts are analyzed to extract the underlying concepts to which these words are related, the concept-based search attempts to find matches for the documents and the query on semantic level. Since the relationship between natural language and conceptual structure is not straightforward, how to extract the semantic concepts from the text written in natural language and how to present the semantic information more efficiently and easily are the main concern for these approaches. Ontology is widely used as a solution for this problem. It promises to provide semantically rich vocabularies and metadata for describing and discovering information resources. However, such application is frequently thwarted by the high cost of building an adequate ontology (conceptual vocabulary) in the first place. In general, this approach cannot be used because it is difficult to build a general-purpose ontology to cover all the concepts of the world, and it is high costly and slow pace. However, when application area is limited to some narrow task domain, the knowledge base tends to be equally limited. This approach can then be taken in some specialized areas because the specialized knowledge in such an area is usually bounded, and can be organized manually. In fact, in many specialized areas, such as computer science, construction and so on, such an “ontology” (or more precisely, thesaurus), already exists. Therefore, one can exploit it for extracting concepts.

## **1.2 Our approach and results**

The integration of such domain-knowledge into IR leads to a specialized IR system. Our task in this project is to develop such a specialized search tool for professionals in the construction sector. Our goal is to find a flexible way in which domain-specified knowledge can easily be incorporated into IR search, and then to explore the possibility of extending traditional IR systems with knowledge-based approaches to improve the retrieval performance.

In our approach, we address issues concerning the application of domain knowledge to IR. We develop a knowledge-based application, which exploits the domain knowledge by using a large, pre-build, technical thesaurus. As an important domain knowledge source, this thesaurus plays a key role in semantic information extraction. Combining with

simple AI techniques, our system can conduct search at a semantic level and improve the system precision.

In the system design, in spite of the simplicity and efficiency, firstly, we decide to use Okapi passage retrieval system as the platform to implement the idea proposed in this project instead of re-building a retrieval system. This allows us to take advantage of the existing IR system and then incorporates semantic information to further improve it. Additional features in the system include a process of semantic information extraction for both queries and answers, and a semantic level matching algorithm for answer re-ranking. The semantic analysis is assisted by a domain-specific knowledge base: TC/CS thesaurus. Experimental results show that an improvement in retrieval performance can be obtained by using this approach.

### 1.3 Organization of the thesis

The remainder of this thesis is organized as follows:

- Chapter 2: Literature review. In this chapter, we survey the literature of the IR and AI. After an overview of the retrieval models and techniques used in IR, we introduce some knowledge representation formalisms (e.g. CG) and knowledge organization system (e.g. thesaurus). In particular, as thesaurus and CG play an important role in our work, we will review some relative aspects about them and we focus on their contribution to IR.
- Chapter 3: Using domain-specific knowledge to improve the IR performance. In this chapter, we introduce the principle of our approach. We provide the theoretical background for designing two main modules of our system. In particular, the TC/CS thesaurus, which serves as domain knowledge base, is described in detail. Rules and algorithms for thesaurus terms semantic tagging and semantic information extraction are also presented respectively. We also present the CG, a well-known formalism of knowledge representation, and then we extend it to a simplified CG for domain-knowledge representation formula used in this approach.

- Chapter 4: System implementation. This chapter describes the realization of the system designed in the previous chapter. We firstly describe the Okapi IR system, which serves as the platform with which our approach is coupled. Then we present the detail of the system implementation.
- Chapter 5: System Evaluation. The experimental tests are described in this chapter. It includes a summary of experimental methods in IR and presents a detailed analysis of the results of the experiments performed with our system.
- Chapter 6: Conclusion and future work. In this final chapter, we draw some conclusions from this study, and point to some future research directions.

## Chapter 2

### Literature review

The amount of available information keeps growing at an incredible rate; a particular example of this is the Internet. Its rapid increase leads to information overload because there are no means for separating relevant from irrelevant information. To utilize this information, whether for business or leisure purpose, we need techniques and tools to allow for fast, effective and efficient access to information. The fields of IR and AI have been looking at this problem. The IR field has developed successful methods to deal effectively with huge amounts of information, whereas the AI field has developed methods to learn user's information needs, extract information from texts, and represent the semantics of information. They converge to the goal of describing and building large-scale systems that store, manipulate, retrieve and display electronic information. The aim of this chapter is to give a survey on methods from IR and AI for searching and retrieving relevant information. It will describe how current techniques from IR and AI can be used for this purpose.

### 2.1 IR

IR is concerned with the organization and retrieval of information from a large number of documents. The primary objective of IR is to locate as many relevant documents as possible while at the same time retrieving as few non-relevant documents as possible according to the information needs expressed in a query. There are two basic tasks in IR. The first one is indexing, where the documents are indexed and classified with the goal of building an internal representation as the translation of the contents of the documents. The second task is retrieval (or search), where a set of documents expected to be relevant to the user's query is obtained by comparing the query with the document representations. In the following subsections, we will describe these two parts respectively.



### 2.2.1 Indexing process

Indexing is a key process in IR that converts a natural language documents into an internal representation. This internal representation, called *document-representation*, must reflect the key information contained in the document and can be handled efficiently by computers. The main goals of indexing process are: 1) the selection of the index terms, generally keywords. 2) the determination of their importance for the document [8].

At the simplest level, a document can be represented as a simple list of words, which are extracted from the appropriate documents. For the first task of the indexing process -- the selection of the keywords, a number of methods have been developed in previous IR systems. For example, one can filter out function words (e.g. prepositions, pronouns, etc.) by utilizing the ubiquitous “stop-list”, which consists of a number of function words, plus words which might not be particularly discriminating for a given subject area of document collection. For example, “computer” may be included in a stop-list if the document collection is composed of computer-related articles. On the contrary, the system can also confine its selection of keywords within those that are put in another list - - “controlled dictionary”. This is the exact opposite of filtering out function words. The system also can bring down the number of terms to be indexed by applying some truncation or stemming algorithm. This causes a mapping of several morphologically related words on the same index entry. More sophisticated methods include statistical methods, which select the keywords based on computing their relative importance weight. It has been identified as the most important method for index terms selection.

Statistical methods consider the frequency of word occurrences for choosing and measuring the index terms for a document. A word that appears very often in a document is considered as denoting an important concept for the document. Frequent words could possibly characterize the content of the document. Early studies [11] suggested that from a retrieval point of view, the most discriminating words in a document were those that occurred with relatively medium frequency. High frequency words, such as pronouns, conjunctions etc, could not distinguish a document from the others. On the other hand,

low frequency words unlikely have enough discriminative power. If a word is dense in a document and sparse in the collection, it is thought to be a good discriminator for the document. The best discriminators then are words that have both high information value and discrimination value. This observation laid the foundations for frequency-based techniques for index term extraction.

In order to determine the best discriminator for the documents, the index terms may be automatically associated with frequency-based weights as suitability measure that reflect the importance of that word as a keyword for the particular document. This association is called *term weighting*. Various term-weighting schemes have been proposed. They can be grouped under two heads: *word-weighting* or *word-document weighting*.

- Word-weighting is related only with the frequency of the word. Among various word weighting schemes, the discrimination value [12] is the most widely used one, which is a measure for the variation in average document-document similarity that is observed when a keyword is withdrawn from or added into the index. This weight is often used as a threshold that can cause keywords to be filtered out altogether.
- Word-document weighting considers not only the frequency of the word in the document but also its distribution in the entire document collection. It uses the frequency of the words within documents and over the database. The most popular scheme is the so-called *tf\*idf* weight [12]. The *tf\*idf* is composed of the term frequency (*tf*) and the inverse document frequency (*idf*). One of the *tf\*idf* formulas is as follows:

$$w_t = [\log(f(t, d) + 1)] * \log(N/n)$$

where  $f(t, d)$  is the frequency of the term  $t$  in the document  $d$ ,  $N$  is the total number of documents in the collection, and  $n$  is the number of documents containing  $t$ . The part  $[\log(f(t, d) + 1)]$  is derived from the term frequency  $f(t, d)$ , and  $\log(N/n)$  is what we call *idf*. This weight is also applied as a threshold that can cause certain document-keyword combinations to be ignored.

### 2.1.2 Retrieval process

Once a set of important keywords has been identified from the documents, we need some mechanism for defining which of the documents meets the requirements of the request. During the retrieval process, the query is compared against each member of the set of document representations; an evaluation method is used to estimate the relevance degree between the documents and the query, a so called similarity measurement. The documents that have a high relevance degree with the query are presented to the user as the retrieval result. The two most used models are Boolean Model and Vector Space Model.

#### Boolean Model

In the classical Boolean Model, the user expresses a query as a Boolean combination of words. The query terms may have been combined using the logical operators, AND, OR and NOT, to form a complex query. The documents are represented as a set of keywords. The evaluation method manipulates those sets with the Boolean operators. Thus for the term  $t$  and document  $d$ , the relevance degree  $R$  may be defined as follows:

$$R(d, t) = wt$$

$$R(d, q1 \wedge q2) = \min(R(d, q1), R(d, q2));$$

$$R(d, q1 \vee q2) = \max(R(d, q1), R(d, q2));$$

$$R(d, \neg q1) = 1 - R(d, q1).$$

where  $wt$  is the weight for the  $t$ , it may be obtained from the  $tf*idf$  weighting or a binary value,  $q1$  and  $q2$  are sub-expressions of the Boolean query, which may be single terms or complex expressions.

#### Vector Space Model

Vector Space Model (VSM) was developed thirty years ago by Salton and his collaborators in the context of the SMART project [13] and it has been the underlying model for many experiments and improvements since. In VSM, each document, as well

as query, is represented as a vector in which each dimension corresponds to a word. The value of a dimension represents the relative importance of the word in the document/query. The document collection is represented as a vector space. Given a vector space as follows:

$$\text{Vector space: } \langle t_1, t_2, \dots, t_n \rangle$$

A document and a query may be represented as the following vectors of weights:

$$d \rightarrow \langle w_{d_1}, w_{d_2}, \dots, w_{d_n} \rangle$$

$$q \rightarrow \langle w_{q_1}, w_{q_2}, \dots, w_{q_n} \rangle$$

where  $w_{d_i}$  and  $w_{q_i}$  are the weights of  $t_i$  in document  $d$  and query  $q$ . The relevance degree is measured by calculating the similarity  $sim(d, q)$  between the query vector  $q$  and each document vector  $d$ . The following is the formula that is the most often used in IR, Cosine formula:

$$sim(d, q) = \frac{\sum_{i=1, n} (w_{d_i} * w_{q_i})}{[\sum_{i=1, n} (w_{d_i}^2) * \sum_{i=1, n} (w_{q_i}^2)]^{1/2}}$$

### 2.1.3 Discussion

Classical IR models are commonly based on keyword in the search. The inherent limitation of these keyword-based IR systems is that they only use individual keywords as representation of the texts. On one hand, such representation is easily extracted from the texts and easily analyzed. But on the other hand, as a word can denote many concepts (polysemy) and a concept can be denoted by many words (synonymy), this kind of search may both miss relevant documents and hit irrelevant ones. It restricts the precision and the diversity of the search results. For example, a search including the word “Java” could return information on coffee beans, a country, and a programming language and there is no way to limit the results correctly in classical IR. On the other hand, a document about “unix” may not be returned as relevance to a query about “operating system” if the words “operating system” are absent in that document [8].

As a shallow representation of text, keywords allow for a fast analysis of the texts and a quick response to the queries. However, the quality of the response may not be satisfactory. In order to solve this problem, a new generation concept-based search has been introduced. In this model, sets of words, noun-phrases, and terms are mapped to the concepts they encode, and a content of an information object is described by a set of concepts. The system search for information object based on their meaning rather than on the presence of the keywords in the object. Such systems mostly employ ontology/thesaurus or some other kind of Knowledge Organization Systems (KOSs) as the basis for the concepts extraction. These knowledge bases have been used to solve the problems of using different terminology to refer to the same concept or using the same term to refer to different concepts. We will describe in more detail in the next section on ontology and thesauri that have been used as an important tool in IR.

Although the concept-based search avoids the problems of the keyword-based search, it comes with problems of its own. It still lacks information about the semantic relations between those keywords or their underlying concepts. To take a simple example, a query on “college juniors” will not be distinguished from “junior colleges” using traditional representation. The question is how to create a more elaborated representation in which not only terms are represented, but also the relationship between them. It is clear that the more information about documents is preserved in their formal representation used for information retrieval, the better the documents can be evaluated and eventually retrieved [6]. Recently, there is a tendency to use more elaborated knowledge representations developed in AI, i.e., CG (CG), to represent the contents of text. For these systems, the query and data encoding language is much more expressive. Not only are the terms represented, but also the relationships between them. For example, a phrase as “University of Montreal located in Quebec” could lead to the following representation:

University of Montreal → (location) → Quebec

Instead of a set of simple keywords “University, Montreal, Quebec”, the retrieved results may not be confused with the documents about the “UQAM”.

The need for more effective IR has become the motivation of creating more intelligent search systems. These systems employ different techniques of AI, for example, knowledge representation or knowledge organization system. These AI techniques are being applied to store, express and classify the large bodies of information, making the extended search possible. In the next section, we will briefly describe some related aspects of AI and their utilization in IR.

## 2.2 Using AI techniques in IR

In AI field, the state-of-the-art AI techniques enable intelligent information process in information seeking. *Knowledge Organization Systems* are mechanisms for organizing information. *Knowledge representation language* is concerned about using languages of mathematical logic to represented information. They play an important role for intelligent information access. Based on these techniques, IR can be enriched to direct information access and automated search fulfillment.

### Knowledge representation language

Knowledge representation language is one of the central concerns in AI. Based on knowledge representation, there exist many powerful tools for transforming contextual knowledge into machine-readable form [2]. A number of standards for knowledge representation have been developed to facilitate knowledge sharing. The NCITS L8 committee on Metadata has been developing two different notations with a common underlying semantics [4]:

- 1) *Knowledge Interchange Format (KIF)*. This is a linear notation for logic with an easily parsed syntax and a restricted character set that is intended for interchange between heterogeneous computer systems.
- 2) *Conceptual Graphs (CG)*. This is a graphic notation for logic based on the existential graphs of C. S. Peirce [4] augmented with features from linguistics and the semantic networks of AI. It has been designed for a smoother mapping to and

from natural languages and as a presentation language for displaying logic in a more humanly readable form.

Both KIF and CG have identical expressive power, and anything stated in either one can be automatically translated to the other. For the standardization efforts, any other language that can be translated to or from KIF or CG while preserving the basic semantics has an equivalent status. Since in our approach, we create a simplified CG as representation language for semantic information. Our presentation of knowledge representation language will concentrate on CG and its utilization in IR, which will be given in section 2.2.1.

### **Knowledge organization systems**

Knowledge Organization Systems (KOS) are mechanisms for organizing information; they are heart of every library, museum, and archive [26]. It is used to organize materials for the purpose of retrieval and to manage a collection. A KOS serves as a bridge between the user's information need and the material in the collection. With it, the user should be able to identify an object of interest without prior knowledge of its existence. According to [26], KOSs are grouped into three general categories: *term lists*, *classifications and categories*, and *relationship lists*.

- 1) Term lists include glossaries, dictionaries and gazetteers, which emphasize lists of terms often with definitions.
- 2) Classifications and categories include subject headings, classification schemes; taxonomies and categorization schemes, which emphasize the creation of subject sets.
- 3) Relationship lists include thesaurus and ontology, which emphasize the connections between terms and concepts. All of these examples of knowledge organization systems, which vary in complexity, structure, and function, can provide organization and increased access to information source. Among them, ontology and thesaurus has been traditionally an important tool in IR. They provide explicit domain theories that can be used to make semantics of

information explicit and machine processable. We will focus on them in following section.

## **2.2.1 Ontology, thesaurus and IR**

### **2.2.1.1 Ontology and thesaurus**

Ontology is the study of the kinds of things that exist and what their basic properties are. The knowledge-management community develops ontology as specific concept models for the purpose of enabling knowledge standardization, sharing and reuse. It is written as a set of definitions of formal vocabulary. In this context, an ontological commitment is an agreement to use a vocabulary (i.e., ask queries and make assertions) in a way that is consistent (but not complete) with respect to the theory specified by ontology. They can represent complex relationships among objects, and include the rules and axioms missing from thesauri.

A thesaurus can be considered as an early, although simple, kind of ontology [7]. Some features in a thesaurus are common to ontological theories, but some others aren't. The common features include organization of terminology and hierarchical structure. Both ontology and thesaurus utilize a hierarchical organization to group terms into categories and subcategories. An important difference between them is that the relationships available for organizing the terms in thesaurus are formally defined. Ontology can introduce a host of structural and conceptual relationships including superclass/subclass/instance relationships, property values, time relationships, and other depending on the representation language used. A thesaurus attempts to show the relationships between terms, whereas an ontology attempts to define concepts and show the relationships between concepts. Thus the machinery for representing concepts in an ontology must be much stronger. Ontology must include a mapping from terms to concepts. No such mapping is formally recognized in a thesaurus. In practical applications, this distinction implies that an ontology will be better than a thesaurus when



it is used in searching. However, such an approach is frequently thwarted by the lack of an adequate ontology in the first place. In general, it is difficult to be used because of the high cost of building an ontology from scratch. Therefore, many applications use an existing thesaurus as a reasonable replacement of ontology.

Many users in library sciences are familiar with thesaurus. For well over a century, librarians have made use of thesauri for building subject classifications and cataloging documents within subject headings. The thesaurus provides a structured representation among terms in a domain; hence it is a kind of meaning representation [27]. Thesauri used in IR may be divided into two categories according to their construction: manually construction or automatically construction [9] [30].

- **Automatic thesaurus**

Automatically constructed thesaurus is usually based on statistics on word co-occurrences: the more two terms co-occur in the same context, the stronger they are considered to be related. The contextual information gathered from a text collection is used to construct a thesaurus automatically using co-occurrence information between terms obtained from text collection. Context may vary from a document, paragraph to sentence.

- **Manual thesaurus**

Manual thesaurus usually contains a set of semantic relationships between words or terms in a specialized domain, or in general domain. There are standards for the development of monolingual thesauri and multilingual thesauri. In these standards, the definition of a thesaurus is fairly narrow. Standard relationships are assumed, as is the identification of preferred terms, and there are rules for creating relationships among terms. The definition of a thesaurus in these standards is often at variance with schemes that are traditionally called thesauri. Many thesauri do not follow all the rules of the standard but are still generally thought of as thesauri.

Thesauri are constructed on concepts and terms, and they show relationships among terms. In a manual thesaurus, relationships commonly expressed include hierarchy, equivalence (synonymy), and association or relatedness. These relationships are generally represented by the notation BT (broader term), NT (narrower term), SY (synonym), and RT (associative or related term). Preferred terms for indexing and retrieval are identified. Entry terms (or nonpreferred terms) point to the preferred terms to be used for each concept. Another type of thesaurus structures concepts and terms not as hierarchies but as a network or a web. Concepts are thought of as nodes, and relationships branches out from them. The relationships generally go beyond the standard BT, NT and RT. They may include specific whole-part, cause-effect, or parent-child relationships. The most known thesaurus is Princeton University's WordNet, which is now used in a variety of search engines.

Many thesauri are large; they may include more than 50,000 terms. Most were developed for a specific discipline or a specific product or family of products. Some domain-specified manual thesauri often use semantic relations of particular salience in the domain. For example medical thesauri may include relations such as "located", "prevents" and "diagnoses". A resource such as the Unified Medical Language System (UMLS) is a highly sophisticated object incorporating a very large quantity of medical knowledge and supporting inference of various kinds. Examples also include the Food and Agricultural Organization's *Aquatic Sciences and Fisheries Thesaurus* and the *National Aeronautic and Space Administration (NASA) Thesaurus* for aeronautics and aerospace-related topics.

An automatically constructed thesaurus has some limitation. In particular, the relations created may not be true. Their utilization in early IR systems shows that their impact on the global effectiveness is limited [5]. It is possible that ever worse system effectiveness is obtained when such a thesaurus is used. Recent works pay more attention to manually constructed thesaurus. In a manual thesaurus, the term relationships established by human experts are more accurate. The use of statistical thesauri in the previous tasks was often

due to the lack of suitable manual thesauri. With more and more large general or domain-specific thesauri available, the thesaurus-based applications have gained in popularity.

### 2.2.1.2 Utilization in IR

The uses of ontology and thesaurus in IR are numerous, but the major ones include [19]:

- Assistance in the selection of appropriate search terms for more accurate information retrieval;
- Enhancing the weight of particular subject terms, as opposed to simple free text searching, thereby enhancing the level of relevancy;
- Guiding the user through changes in the nomenclature being used within particular subjects. This particularly applies to historical collections of records where the meaning and usage of words may change over time.

There are many ways to apply the above ideas to aid the information retrieval. They include: query expansion, applying search term guidance and assistance in similarity measurement. In the following subsections, we give an explanation on each of them in a little more detail.

- **Query expansion**

Techniques for automatic query expansion have been extensively studied in information retrieval as a means of addressing the word mismatch problem between queries and documents [8]. Query expansion works as follows: Given an initial query of the user, some new related words are added and this forms a new query. The addition of the new words extends the original query so that it has a wider coverage than the original query. Therefore, even if a document does not use the same words as the original query, it may still be judged to be relevant if it contains the words that are added through query expansion. As a consequence, more relevant documents may be retrieved, and the recall ratio be increased. The key problem is to add the appropriate words. Otherwise, the new query will depart from the original query (in meaning). So an important question is what words should be added.

Many solutions have been used in previous studies.

1. One solution is to use word co-occurrence. Research on word co-occurrence information used in automatic query expansion was already under way for many years. The idea was to obtain additional relevant documents through expanded queries based on co-occurrence information of query terms. It could be achieved by using measures of association between keywords based on their frequency of co-occurrence, which is the frequency with which any two keywords occur together in the document collections. Then a query is expanded by choosing carefully additional search terms on the basis of these statistical co-occurrence data.
2. Another solution is to use relevance feedback mechanism. Relevance feedback aims to solve the problem that the user's query is a bad description of the information need. One could know the words but misunderstand them and know the concepts but for whatever reason, use words for them that are less typical. After the result documents have been retrieved, the user is asked to judge them by indicating a set of them as relevance ones that near his information need. Based on these indications, the system updates the query towards those documents and away from undesirable ones by incorporating the words found in the relevant documents (or increasing their weights), and eliminating those in the irrelevant documents (or decreasing their weights). The typical query reformulation using relevance feedback is that of Rocchio (Salton and McGill 1983):

$$\text{New\_Query} = \alpha * \text{Old\_Query} + \beta * R - \gamma * \text{NR}$$

where R and NR are the centroids of the set of relevant and irrelevant documents judged by the user;  $\alpha$ ,  $\beta$  and  $\gamma$  are factors that determine the importance of the old query, the relevant and irrelevant documents to the new query. As we can see in the formula, the new query becomes closer to the relevant documents and more distant from the irrelevant documents.

3. An alternative query expansion relies on a thesaurus. An automatic process first tries to identify the most closely related words from the thesaurus and then adds them to the query. The words to be used in query expansion can be determined in a number of ways:

- Some form of semantic distance measure can be used to determine which words should be added into the query.
- The types of relation traversed can be constrained. A particularly useful form of this would be to augment the query with words of the initial and narrower concepts.
- A word selection tool would allow the user to manually select the concepts used.

Usually, automatic query expansion consists of selecting the related words that are linked with the original query words through some types of relationship that are judged to be “strong” relationships. It is also observed that the so-found new terms may have very different meanings than the original terms. Therefore, it is a common practice that the related terms are weighted less than original terms. In addition, the longer the relationships path one has to traverse to link the two terms, the lower the new term is weighted in the new query.

- **Applying search terms guidance**

Since a thesaurus is a tool for vocabulary control, by guiding indexers and searchers about which terms to use, it can help improving the quality of retrieval. A thesaurus encodes not only the conceptual vocabulary but also semantic relationships between concepts [3]. All the terms in the thesaurus are connected to each other in a network of semantic relationships. The relationships among terms are clearly displayed and identified by standardized relationship indicators, for example, BT for broad terms, NT for narrow terms, and RT for related term and SN for the scope note. These links between thesaurus terms can help to direct user to the right term and make the meaning of a term clearer.

A number of concepts and index terms may be linked together offering the user opportunities to broaden or narrow their search. This is done in thesaurus by the use of broad terms (BT) and narrow terms (NT) to indicate the hierarchy of concepts and terms. The use of broader and narrower terms in research can help the user to select the

appropriate search term more accurately and can help in making the results of the search more relevant. An interface of the thesaurus concepts network can be presented to the user. By navigating around network, concepts can be reached. This enables the user to refine or expand their query. Furthermore, when a user tries to investigate a number of aspects of a particular subject, he may wish to broaden his search on related aspects to the particular subject. In this instance the use of a thesaurus can be an invaluable aid in the process of lateral thinking that this may require. In this instance, the use of related terms (RT) can help widen a search to terms, which cover the related aspects. The user obtained more information that may be also relevance to the particular, than the search with the initial terms. Another use of thesauri is in the judicious use of scope notes. Scope notes contain information relating to the thesaurus term selected that assists in interpreting its suitability as a search term. A scope note may be used to reflect the changing use of a particular word in response to developments in technology or science and the reuse of existing terms for new concepts. For example, the term 'aids' was, in the early 1980s, taken to mean aids for disabled people such as hearing aids, walking sticks, wheelchairs, etc. Since the 1990s, however, AIDS has taken on a new meaning as an acronym for Acquired Immune Deficiency Syndrome. Similarly the term 'micro', as used in English in the 1970s, referred to a diminutive of 'microwave oven', whereas since the 1980s this word has normally is taken to refer to a 'microcomputer'. The scope note is very useful in mapping the syntactic changes occurring over an extend timeframe. It indicates the changes in the use of terminology over a particular time period, points to the correct term to use at any particular time. This use of pointers to the appropriate term when a number of equally appropriate terms could be used.

- **Assistance in Similarity measurement**

A central feature of thesaurus is their hierarchical or network organization. This offers many benefits for language engineering, including the potential for measuring semantic similarity between two word meanings by finding the length of the shortest path between them across the network. WordNet has been used extensively in this way, with various measures proposed and explored.

### 2.2.2 CG and IR

In AI, the study on knowledge modeling is a core theme. CG is a well-known knowledge representation method created by John Sowa in the 1980's [17]. It has been developed as a graphic representation for logic with the full expressive power of first-order logic and based on the semantic networks of AI. With their direct mapping to language, CG can serve as an intermediate language for translating computer-oriented formalisms to and from natural languages. With their graphic representation, they can serve as a readable, but formal design and specification language.

A CG is a finite oriented connected bipartite graph [28,17]. The two different kinds of nodes of this bipartite graph are concepts and relations. The concept nodes represent entities, attributes, or events; the relation nodes identify the kind of relationship between two concept nodes. A CG is a network of concepts nodes and relation nodes. Build and manipulating CG is mainly based on six canonical rules [28]. Two of them are the generalization rules: *un-restrict* and *detach*.

- Un-restrict rule generalizes a conceptual graph by unrestricting one of its concepts either by type or referent. Un-restriction by type replaces the type label of the concept with its super-type; un-restriction by referent substitutes individual referents by generic ones.
- Detach rule splits a concept node into two different nodes having the same attributes (type and referent) and distributes the relations of the original node between the two resulting nodes. Often this operation leads to separating the graph into two unconnected parts.

For the purpose of IR, it is important to be able to approximately compare two pieces of knowledge represented in CG. Different similarity measures have been described for comparing the query with graphs from the knowledge base. One of the comparison criteria most widely used for CG is that if the query graph is completely contained in the given graph, then the given graph is relevant for the given query graph [6]. This criterion

means that the contents of the found piece of information have to be more particular than the query piece. More flexible matching criterion is proposed in [6], it is based on well-known strategies of text comparison, (i.e., Dice coefficient) with new elements introduced due to the bipartite nature of the CG. The comparison consists of two steps: 1) find the intersection of two graphs, 2) based on the intersection graph, measures the similarity between the two graphs combining two types of similarity: the concepts similarity and relational similarity.

## 2.3 Discussion

### 2.3.1 Difficulty of forming a complete CG

While CG theory provides a framework in which IR entities can be represented adequately, much of the representation task involves intellectual analysis of documents so that we capture and store concepts and relations for IR. The analysis usually relies on a natural language processing (NLP). NLP is one way to determine the semantic information by analyzing syntax and semantics of natural language text. Many of techniques in NLP are applied to various tasks related to document retrieval process. An interesting application of NLP is the analysis of text to identify various relationships among the linguistic units. To carry out this task, the analyses of the natural language need to be performed at morphological, lexical, syntactic and semantic levels.

- The morphological level involves processing of the text at individual word forms level and identification of prefixes, infixes, suffixes and compound words.
- The lexical level deals with operations on full words, such as identification of stopwords, and misspelling, handling of acronyms and abbreviations, and assignment of parts of speech categories to lexical items.
- The syntactic analysis of natural language texts deals with recognition of structural units, such as noun phrases.
- The semantic level of analysis involves representing the meaning of the natural language text.



Morphological, lexical and syntactic analyses have been used in IR research. These researches attempt to identify multiword phrase, and syntactic variants that refer to the same underlying concepts. However, semantic analysis requires a more deep understanding of text. That is an extremely difficult task. For many cases, it is impossible to create a complete CG for the text. As a result few IR systems are based on general semantic analysis. For example, in [6], they use a traditional keyword search as a platform to select the potentially relevance documents for a query, then the extraction process, which is used to construct the CGs for the retrieved documents, is only performed on parts of the potentially relevance documents: on titles and abstracts of the documents. They also only use 4 syntactic relations to create the CG representation. Therefore, [6] uses semantic analysis techniques usually used as a helpful tool to improve the results of conventional retrieval methods, not as a replacement of conventional methods.

Due to the complexity involved in NLP processing, many applications try to find another way to extract the semantic information of the texts, for instance, using word co-occurrence information.

### **2.3.2 Utilization of word co-occurrence information**

Word co-occurrence information is normally used to choose related words in automatic query expansion. It also can help derive the conceptual 'meaning' of a word depending on the context it was used. By recording the frequency of co-occurrence between words in the text, we could use this distribution information as a profile of the word's usage; accurately associate those words that have strong connection. This information could be very helpful in disambiguating the domain-dependent word senses from their common senses in domain-specific information retrieval. In domain-specific information retrieval, some domain-dependent terms refer to the special meaning. They should not be interpreted in their common word senses. Such unusual word senses strongly call for inference from domain-dependent lexicon information. Using co-occurrence word information trained from the domain-specific document collection could help

disambiguate unusual word sense from their common word senses and improve the IR efficiency.

Our approach is an extension to this idea; we propose a technique for extending word co-occurrence to semantic representation. By recording co-occurrence information between concepts in the text, we attempt to use this distribution information as a profile of document content into IR search.

### 2.3.3 Our approach

Our goal in this project is finding a flexible way in which domain-specified knowledge can easily be incorporated into IR search, and to exploring the possibility of extending traditional IR systems with knowledge-based approaches to improve the retrieval performance. Since our project is a prototype for this purpose, we do not intend to deal with all kinds of relationships. Actually, we cannot do it because we do not have elaborated description of every concepts and the possible relationship it can have with others. These descriptions and relationships are necessary for the creation of CG. Based on this consideration, we decide to use co-occurrence relations in semantic presentation to replace the exact relation. Instead of finding relations between words, we restrict to record the co-occurrence information between concepts. The strength of association of two concepts could be measured based on their co-occurrence. The statistics can be reliably used to estimate the co-occurrence probability of the concepts.

Another challenge for this work is the construction of ontology appropriate to the domain of interest. To address this, we have used a technical thesaurus as the initial ontology, seeking to exploit the many years of effort already spent by librarians and specialists in constructing a conceptual vocabulary for a domain. In this thesis, we describe a knowledge-based application, which addresses this issue by using a large, pre-build, technical thesaurus as an initial ontology, combined with simple AI techniques, conducting search in terms of concepts rather than words. The significance of this work

is that it demonstrates the utility of domain knowledge, i.e., thesaurus and document collection, for information search.

## Chapter 3

### Using domain-specific knowledge to improve IR performance

This chapter describes our approach to improve IR performance by using domain-specific knowledge. By combining the traditional IR and AI techniques, our project explores the possibility of extending traditional IR systems with knowledge-based approach to increase the retrieval effectiveness. Domain-specific knowledge bases are adapted in this project; it consists of a technical thesaurus and a collection of publications related to our particular subject area -- construction science and technology. As a valuable resource for domain knowledge, the thesaurus is rich with relevant notions, important concepts, and professional information in the field of construction.

#### 3.1 Introduction

There are several kinds of knowledge that IR model has to tackle with [1]. *Content knowledge* consists of the domain concepts that describe the semantic content of basic objects. *Structure knowledge* is made of links between basic objects. The concept of domain knowledge consists of these two types of knowledge. For example, in the construction field, the expression "the corrosion of the stainless steel" contains two domain concepts:

“corrosion” → [degradation of material]

“stainless steel” → [metallic material]

The “corrosion” is a main type of material degradation that must be considered in material engineering. It belongs to the concept “degradation of material”. “Stainless steel” is one of the most important metallic materials; it is compatible with the concept “metallic material”. Moreover, particularly in construction domain, this expression implicitly represents a relationship between these two domain concepts: corrosion is a

chemistry phenomena usually taking place on the surface of metal, it is a basic chemical property of metallic material. There is a semantic linkage between these concepts:

corrosion (degradation of material) → related to → stainless steel (metallic material)

In order to exploit domain knowledge into information retrieval, firstly, we need a domain knowledge source to help the identification of the semantic concepts specific to the domain. Secondly, we need an explicit representation language capable of describing both semantic concepts and relations between them. To address these issues, in our project, we use a construction thesaurus (TC/CS thesaurus) as conceptual vocabulary to identify the underlying domain concepts. Based on CG, we create a simplified triple form as our representation language. A collection of articles published in the construction field, *Canadian Building Digest*, is also used as a training corpus to estimate the co-occurrence information among the domain concepts.

In this chapter, we explain the principle of our approach by presenting two main components of the system: semantic interpretation module and relevance measurement module. The remainder of this chapter is organized as follows. In section 3.2, we give an introduction of the semantic interpretation module, specific characteristics of the construction technique thesaurus and appropriate formalism for semantic information representation. In section 3.3, we present the relevance measurement module, the weighting scheme and the role of the document collection in relevance measurement processing. Finally, we briefly describe the framework of our system.

## 3.2 Semantic interpretation module

For establishing semantic representation of the domain knowledge, it is necessary first to identify domain-specific semantic concepts described in a text, and then use an appropriate representation language to form the semantic representation for the text. The semantic interpretation module performs these tasks by cooperating with the related domain-knowledge source and the interpretation language. In this section, the TC/CS thesaurus as our domain-knowledge source is described and discussed in detail, the

algorithm of semantic tagging for the terms in TC/CS is followed, and then we introduce the simplified CG as our representation formalism a brief introduction of the main notions of the CG.

### 3.2.1 Domain knowledge Source

The notation of knowledge has been studied in many disciplines ever since the early days of science. Nowadays, the domain knowledge has become popular in the AI community. Recently, the most widely known and used computational forms of domain knowledge are ontology and thesaurus. In this section, we represent the TC/CS thesaurus in detail and show how it is used in our approach.

#### 3.2.1.1 The TC/CS thesaurus

The particular thesaurus we have used in this project is the Canadian Thesaurus of Construction Science and Technology (TC/CS). This thesaurus was developed by the IF Research Group of University of Montreal on construction. It is well suited to our purposes as it is highly customized to our target domain -- construction science. It is important to note that a thesaurus encodes not only the conceptual vocabulary but also semantic relationships between concepts [3]. All the terms in the thesaurus are connected to each other in a network of semantic relationships. The relationships among terms are clearly displayed and identified by standardized relationship indicators. In TC/CS thesaurus, there are over 15,000 construction concepts, with approximately 26,000 links between them, there are three main relationships included in TC/CS thesaurus: 1) equivalence relationship; 2) hierarchical relationship and 3) associative relationship.

1. The *equivalence relationship* exists between a *preferred* term and a set of *lead-in* terms [29]. The lead-in terms are used for pointing to the other terms, called the preferred terms, which may have hierarchical and associative relationships with other preferred terms making up the thesaurus. The equivalence relationship is used to

gather synonyms that refer to the same or closely related meanings. It covers variant spellings, abbreviations, acronyms, popular forms of scientific terms, and so on. A preferred term for a lead-in term is indicated by the US (use) mark in the TC/CS thesaurus. Conversely, a lead-in term is indicated by the mark UF (Use for).

2. The *associative relationship* connects two terms that are conceptually related [29]. This relationship can be used to identify such a associations as between thing and its application, an effect and a cause, an activity and an agent of that activity, a thing and its parts, and so on. The associative relationship is indicated by the RT mark in TC/CS thesaurus. For example, “oxides” is connected to “metallic materials” by this relationship.
3. The *hierarchical relationship* is the primary feature that distinguishes a systematic thesaurus from an unstructured list of terms [29]. It covers three different categories:
  - *generic relationship*: identifies the link between a class and its members or species, i.e., “metallic materials” and “material”. It is the most common hierarchical relationship in the TC/CS thesaurus. It is indicated by the mark BT (broader term) for the class concept, and the NT (narrower term) for the species concept.
  - *whole-part relationship*: covers situations in which one concept is inherently included in another, regardless of context, i.e., “metallic material” and “properties of metal”. It is indicated by the marks: WT (whole term) and PT (part term) in TC/CS thesaurus.
  - *instance relationship*: identifies the link between a general category of things or events, expressed by a common noun, and an individual instance of the category, often a proper name. In TC/CS, it is indicated by the mark GT (General related term).

The most important relationships used in our project are BT/NT (broader/narrower term) and WT/PT (whole/part term). More specifically, BT denotes a subject area that encompasses the original term; WT denotes a composed entity that is made up by the original term. They usually correspond to super-ordination link in the inheritance hierarchy, while NT and PT are the inverses of them, corresponding to sub-ordination

link in the hierarchy. The table below lists all the relationships set up between terms and their abbreviations used in TC/CS.

<b>Relationship</b>		<b>Indicator</b>	<b>Abbreviation</b>
Equivalence		Use	<b>US</b>
		Used for	<b>UF</b>
		French term	<b>FT</b>
Hierarchy	Generic	Broader term	<b>BT</b>
		Narrower term	<b>NT</b>
	Partitive	Whole term	<b>WT</b>
		Part term	<b>PT</b>
	Instance	General related term	<b>GT</b>
Association		Related term	<b>RT</b>
		Associated structured term	<b>AT</b>

**Table 1: an explanation of the relationships and their abbreviations used in TC/CS**

In TC/CS thesaurus, there are 10 levels in this hierarchy, going from general terms such as "Science" and "Action" at the top level down to details such as "Preselector" and "Piston" at the lower level. An example for the descriptor "Metallic Material" is included below.



<b>metallic materials</b>	
LEVEL	5
FT	materiau metallique
UF	ferrous materials
UF	non ferrous metals
BT	materials
NT	alloys
NT	metallurgical products
NT	metals
PT	properties of metal
RT	building materials
RT	metallic elements
RT	oxides

**Table 2: An example of descriptor in TC/CS for concept “metallic materials”**

In all cases, any TC/CS descriptor can point to a BT /WT that is one level above, or a NT/PT that is one level below. For example, in table 2, the descriptor “metallic material” is a level-5 term, its BT relation points to “material”, a level-4 term. It also has three level-6 NTs: “alloy”, “metallurgical product” and “metal”. Since the relationship is symmetrical, if B is a broader term for A, then A is a narrower term for B. Therefore, these three terms must have at least one BT linked to “metallic material”. A tiny fragment of the hierarchy graph around the term “metallic material” is shown in Figure 1.

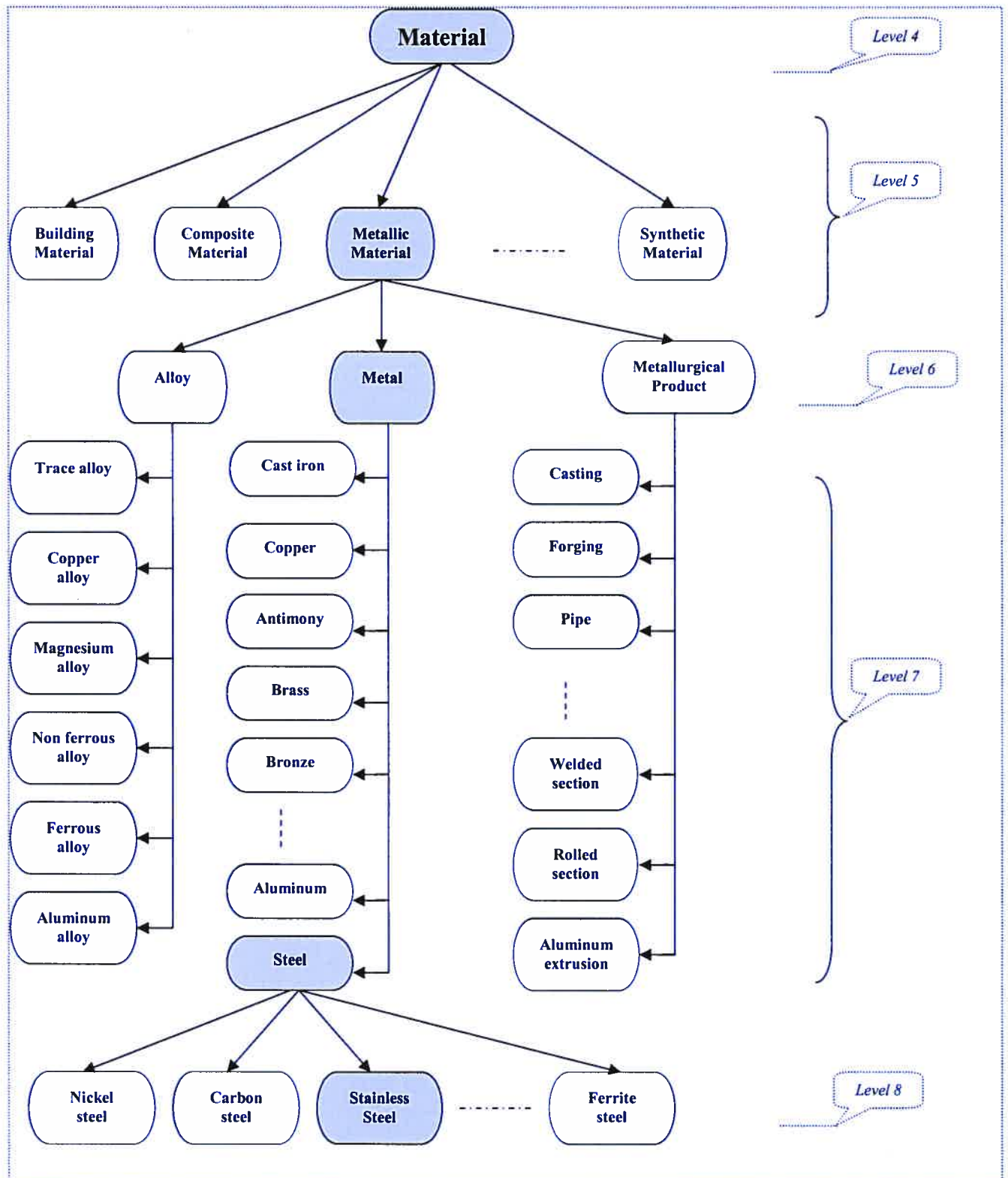


Figure 1: A fragment of the hierarchy tree for “NT” relationship in TC/CS

In the figure, terms are represented by rectangles; directional arrows represent a direction of hierarchy from upper level to lower level linked by the NT relationship. If we choose some upper level terms, for example “Material”, as semantic concept, it is easy to infer that a “Stainless steel” belongs to this concept “Material” by tracing through the graph. (i.e. “Stainless steel” → “Steel” → “Metal” → “Metallic material” → “Material”, shown in gray in the graph). It appears obvious that by using the structure of a thesaurus, we can take advantage of its taxonomic organization to tag the terms with this semantic concept. Based on this observation, we design an algorithm to group all terms in thesaurus into some predefined concepts.

### 3.2.1.2 Definition of semantic concepts

In order to transfer a natural language sentence into a set of simplified CG, we have to recognize semantic concepts of terms in the sentence. That is the need of defining the semantic concepts for the thesaurus terms. The semantic concept represents a conceptual grouping of similar terms. All of the terms in thesaurus are instances of some semantic concepts. The key issue on categorizing the thesaurus terms is the definition of the semantic concepts.

We observed that the level-4 terms in TC/CS thesaurus contain the most general and important concepts of construction domain, i.e., “material”, “building process”, “urban planning”, “construction technology”, etc. Therefore, we chose level-4 terms as candidates for the concepts definition. After a subjective evaluation on all 476 level-4 terms, a subset of them was chosen manually as semantic concepts according to the evaluation result. This subjective evaluation was carried out by the construction experts. It is based on the likely needs of Cibât’s clientele (Centre International du Bâtiment). This center serves the Canadian building industry and the market by improving the use of information about building, building products and building requirements, and by providing a better access to that information. The result of the evaluation distinguishes a certain number of terms from the others by importance. They are probably the most important concepts that capture issues of interest for the construction professionals.

During the subjective evaluation, each term was assigned a numerical value to reflect its relative importance and generalized in construction practice. The value is scaled from 1 to 10 with the higher value indicates the more importance level. The complete evaluation result is provided in Appendix. Here is a fragment of some popular terms.

<b>Descriptor</b>	<b>Importance value</b>
Building process	10
Construction technology	10
Properties of material	10
Experimental method	10
Housing	10
Equipment	10
Dwelling unit	9
Infrastructure	9
Acoustic	9
Manufactured product	9
Physical treatment	9
Property	9
Safety engineering	8
Fluid mechanic	8
Strength of material	8
Natural resource	8
Structural engineering	8
Urban Planning	8
Animal	7
Business	6
Chemical function	5
Authority	5
Civil law	4
Bacteriology	2

**Table 3: A fragment of the result of the subjective evaluation**

The term that has an importance-value exceeding a predetermined threshold was taken as a semantic concept. Considering the construction expert's opinion and the total number of the concepts, 7 is used as the threshold in our experiment. We obtain a list of 86 terms that have importance value 10, 9, and 8 as our semantic concepts.

### 3.2.1.3 Algorithm for semantic tagging

As we described above, given a thesaurus term and a set of predefined semantic concepts, one objective is to classify the terms into some proper semantic concepts by tracing through the links between this term and its upper level terms. In TC/CS thesaurus, both BT and WT relationships can point to a super-ordinate term. The question is: what are the differences between these two relationships and which relationship should we consider in semantic tagging process?

In many thesauri, there is just one sort of hierarchical relationship: BT/NT (Broader terms and narrower terms). It is used to represent all possible hierarchical relations: generic, partitive and instance relationship. The broader term represented the class, whole or general category, the narrower term reciprocally represents the subclass, part or particular instance. Although the disparity between the situations is not evident, it is still possible to distinguish the WT relationship from BT, since the former stating a "component parts of" relationship, the latter stating a "specific types of" relationship. The WT relationship includes several types: geographical, systems and organs of the body, disciplines and fields of knowledge and hierarchical social structures. More substantial hierarchical relationship is generic. The whole-part relationship is not strictly speaking a hierarchical one. Based on this consideration, we consider that the BT relationship has a higher priority level than WT in the semantic tagging process.

According to this criterion, the inference process will consider relationships with different levels of priority. Our algorithm for semantic tagging is shown below:

Relationship = {BT, WT}

For each Relationship

    For each term in a sentence

        Semantic tagging following the Relationship

        Store results in Concepts

If Concepts is empty

    Then semantic tagging continues with the BT and WT mixed

Return Concepts.

### 3.2.2 Semantic representation

In AI, semantic networks are the knowledge representation method of choice in formally describing the relationships among concepts. CG is a well-known formalism for semantic networks. In many applications in IR, CG is used to describe the concepts and relations among them. This section presents the simplified CG as our semantic representation after a brief introduction of CG.

#### 3.2.2.1 CG

CG has been developed as a graphic representation for logic with the full expressive power of first-order logic and based on the semantic networks of AI [16]. Their purpose is to express meaning in a form that is logically precise, humanly readable, and computationally tractable. This kind of formalism knowledge representation incorporates information about both the concepts mentioned in the text and their relationships. The use of CG for knowledge representation in IR has been exploited. It has been the basis of representation language used in our project because it has appropriate properties for representing linguistic concepts and their relationships. However, in our case, much simplification is made.

CG is formally defined by an abstract syntax that is independent of any notation, but the formalism can be represented in either graphical or character-based notations [16]. In graphical notation called *Display Form* (DF), CG is represented as labeled graphs of two kinds of nodes: concept nodes and relation nodes, where concept nodes are connected by relation nodes. With their direct mapping to language, CG can serve as an intermediate language for translating computer-oriented formalisms to and from natural languages. With their graphic representation, they can serve as a readable, but formal design and specification language.

CG also can be represented in several different concrete notations. For example, every CG can be represented in the compact but readable *Linear Form* (LF), or in the formally defined *CG Interchange Form* (CGIF). Any semantic information expressed in any one of these three forms can be translated to the others without loss or distortion. They can also be translated to a logically equivalent representation in predicate calculus and in the *Knowledge Interchange Format* (KIF). The following is an example from [15] for English sentence “John is going to Boston by bus”. It illustrates these three notations of CG.

- **DF** (Display Form): Figure 2 shows the DF of a CG for the English sentence: “John is going to Boston by bus”. In this graph, the concept nodes represent the elements mentioned in the text; they are represented by rectangles: [Go], [Person: John], [City: Boston], and [Bus]. The relation nodes identify the kind of the relation between two concept nodes; they are represented by circles: (Agnt) relates [Go] to the agent John, (Dest) relates [Go] to the destination Boston, and (Inst) relates [Go] to the instrument bus. The arcs that link the relations to the concepts are represented by arrows. For relations with more than two arguments, the arcs are numbered.

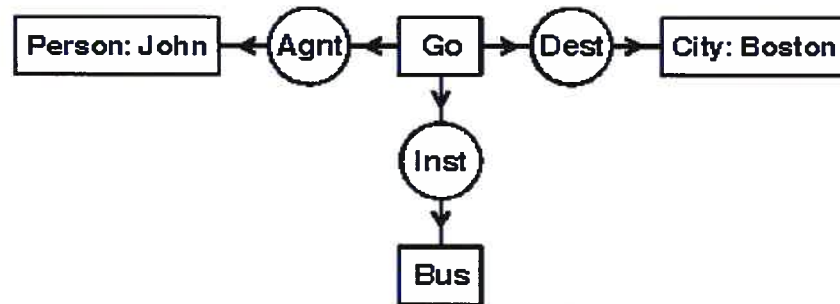


Figure 2: DF of CG for “John is going to Boston by bus”.

- **LF (Linear Form):** LF is intended as a more compact notation than DF, but with good human readability. It is equivalent in expressive power to the abstract syntax and the display form. In this form, the concepts are represented by square brackets instead of boxes, and the conceptual relations are represented by parentheses instead of circles. The hyphen on the first line indicates that the relations attached to [Go] are continued on subsequent lines. Figure 3 is the LF for Figure 2.

```

[Go] ---
  (Agnt) --> [Person: John]
  (Dest) --> [City: Boston]
  (Inst) --> [Bus]
  
```

Figure 3: LF of CG for “John is going to Boston by bus”.

- **CGIF (CG Interchange Form):** The basis of CCIF notations was a syntax developed by Gerard Ellis [16] for a rapid parsing of simple CG. As an official textual notation for CG, CGIF has a simpler syntax and a more restricted character sets. Following is the CGIF for Figure2. Here an asterisk mark and a question mark are used as co-reference labels to three occurrences of the concept [Go] to indicate that they refer to the same instance of going.

```

[Go *x] (Agnt ?x [ Person: John ]) (Dest ?x [City: Boston ]) (Inst ?x [Bus])
  
```

Figure 4: CGIF for “John is going to Boston by bus”.



Both DF and LF are designed for communication with humans or between humans and machines. For communication between machines, we could use the CGIF. CGIF is intended for transfer between IT systems that use CG as their internal representation. It is one of the standard notations for exchanging knowledge permitting fast interchange of CG. It has a minimal abstract syntax and notation that has a proper foundation with translations to KIF and permits extensions to be built upon. CGIF is less readable than LF that is graph-oriented but it is more editable in computer environment since it can be treated as a string. The reason for developing CGIF was to support the interoperability for CG-based applications that needed to communicate with other CG-based applications. In our project, we create a simplified CG based on this formalism to represent the semantic information of domain knowledge.

### 3.2.2.2 Simplified CG representation

The simplest CG is Star graph (SG). In CG standard draft [16], Star graph is defined as a CG that contains a single conceptual relation and the concepts that are attached to each of its arcs. According to this standard, one of the most importance features for a CG is that: any CG  $g$  with  $n$  conceptual relations can be constructed from  $n$  star graphs, one for each conceptual relation in  $g$ .

For example, considering the CG in Figure 2, it contains three conceptual relations: [Agnt], [Dest], [Inst], it could be constructed from three star graphs, which are represented below in CGIF:

( Agnt [Go] [Person : 'John'] )

( Dest [Go] [City: 'Boston'] )

( Inst [Go] [Bus] )

A star graph may be represented as a string of CG that contains exactly one string of Relation and two strings of Concept. Every Concept string in the CG string must

represent one of the concepts attached to the conceptual relation of the star graph. Thus, the star graph is represented by a string as:

Relation [Concept1] [Concept2]

A CG may be constructed from a set of star graphs and a star graph can be represented as a “Relation-Concept1-Concept2” semantic triple, then a sentence may consist of several semantic triples. In our project, we simplified this triple formalism to represent the semantic information of domain knowledge: we use statistical co-occurrence relation to replace the exact semantic relation.

### **Why do we need to simplify CG?**

The simplification is based on several considerations.

- Firstly, it is because of the difficulty of forming a complete CG for a sentence. In order to identify various relationships among the linguistic units, we need to determine the semantic information by analyzing syntax and semantics of natural language text. The analysis usually relies on a NLP. NLP techniques are used to automatically extract facts from plain text; it helps in converting document written in natural language into CG representation. The analysis involves passing the text through various stages of linguistic processing, including word-level, syntactic, and semantic analysis. That is an extremely difficult task, even on a limited domain. It is known that current NLP is not accurate and powerful enough to recognize the contents of unrestricted text. For many cases, it is impossible to form the complete CG for the text. Correspondingly, we can observe the advantages for using the simplified semantic triple replacing the complete CG. The essential of CG is directed bipartite graph, with edges going between concepts and relations. Simplified semantic triples used in our project exactly describe the most basic units making up the CG. The utilization of the simplified semantic triples avoids the complicated linguistic processing. They are easily editable and interpretable.
- Secondly, it is because of the difficulty of the comparison of two CGs for relevance degree measurement. A long phrase may be represented as a set of CG. The method

for the comparison of two CG representations of two texts is more complicated. Some of them are restricted to determine if a graph is completely contained in the other one; in this case, neither description nor measure of their similarity is obtained. Some other methods allow flexible matching of the graphs, measuring the similarity between two CG, but they typically describe this similarity as the set of all their common elements existing in two graphs. In general, the comparison algorithm finds all sub-graph, star graph, of the initial graphs, then the measurement of similarity is applied to each one of them separately, and only the highest values are kept. Using the simplified semantic triple, we still keep the most important parts of a CG for the relevance measure, but avoid the decomposition step. Moreover, since the simplified triple can be treated as a string, the similarity measurement could be performed using the traditional relevance measurement methods to avoid comparison of the graphs. This modification allows simplifying the computation of the similarity as well as constructing a precise description of this similarity. More detailed description about this issue will be given in section 3.2.

The complexity of the generation for CG usually decreases applicability to large documents sets. As the textual information is increasingly available in electronic forms, several approaches have been developed using the simple formalism of the CG as document descriptors in order to deal with the huge document database. In the approach of [18], they use star graph formalism as document descriptors to represent elementary pieces of information. In order to form a set of star graphs, a CG is iteratively split until each concept node has exactly one adjacent edge. Then SGs extracted from the collection are considered as document descriptors. This has been approved to be a feasible and useful way to improve the search result in previous studies [18].

### **How to create the simplified CG representation?**

The simplification we made in the semantic representation is about the relation. We focus on the statistical correlation relationship between concepts. The basic idea behind this simplification is that the correlation information between concepts can be learnt through a training corpus in which only semantic categories have been tagged. Following

this idea, we extract every pair of concepts co-occurring together within a sentence, collect pairs of semantically or contextual associated concepts, then we represent them as a set of simplified CG of the following form:

*Co-occurrence (concept1, concept2)*

For the example we mentioned before, the expression "the corrosion of the stainless steel" contains two domain concepts: "stainless steel" and "corrosion", and a implicit relationship between them. As both concepts occur in the same sentence, we can extract the following simplified relationship between them:

Co-occurrence (corrosion, stainless steel)

### **Why do we choose the co-occurrence relation?**

The basic assumption of co-occurrence in IR is that if two items often co-occur together then the strong association exists between them. The analysis of semantic concepts co-occurrences information will give a measure to determine the strength of an association between concepts in a domain. The strength of the co-occurring concepts can be measured from the number of times two concepts occur together within the domain-specific document collection. The more one concept co-occur with another one, the stronger the association those two concepts have. For example, the concepts "material" and "building" have a strong relation in construction domain, apparently co-occur a large number of times in the domain-specific corpus. These concepts tending to appear together in the collection is taken as evidence of possible relationship between them. So that it is reasonable that we establish a semantic linkage between them. That is the reason why we simplified CG using co-occurrence relation.

A series of recent studies have successfully employed co-occurrence to generate the term association information to help in searching. Early experiments demonstrated the effectiveness of co-occurrence data for improving the performance of IR system. In our project, other domain knowledge bases --- publication collections, is used as a training corpus to obtain the co-occurrence information about domain concepts. Through co-occurrence analysis, we extracted the concepts that are the semantically associated, also

contextually related. It is possible to cover the situation when two concepts are non-semantically related but really context dependent, which could also be useful data for improving the IR performance. Furthermore, by incorporating it within the similarity measurement, co-occurrence information brings more background data about the application domain into relevance measurement. That is another advantage gained from using co-occurrence relation instead of exact relation in simplified CG.

### **3.3. Relevance measurement module**

Another main component of our system is relevance measurement module. After choosing the representation form, we need a similarity measure to evaluate the relevance between the document and query. Several measures have been proposed in the literature. We take the Dice similarity measure as the basis, as it is the one primarily used in IR systems. We extend the basic calculation by adding a decaying factor that decreases the Dice coefficient when the distance between the terms increases.

#### **3.3.1 Training corpus**

As we mentioned in the section above, using simplified CG as representation form brings an advantage for the relevance measurement. We can use other domain knowledge bases --- publication collections as training corpus to estimate the domain concepts and obtain their co-occurrence information; it will provide necessary background data in similarity measurement.

Although in TC/CS thesaurus, the term relationships established by human experts are accurate, and a helpful in identifying domain-specific semantic concepts described in a text, the weakness is that the strength of these relationships is not measured quantitatively, making it difficult to incorporate relations directly into similarity measurement. One solution is to use statistical methods to estimate their connection in a documents collection, which indicates the strength of their association. The correlation

between two concepts can be learnt through training lexical co-occurrence information in a training corpus.

We use the Canadian Building Digest as our training corpus. This corpus is a collection of 250 articles, which take up about 5 Megabytes, published between 1960 and 1990 by (National Research Council) NRC's Institute for Research in Construction and its predecessor, the Division of Building Research. The documents in the collection contain background information and practical guidelines on virtually every aspect of building design and construction in Canada. The correlation information between the semantic concepts obtained from them is mostly representative in the construction field.

### 3.3.2 Reliability assessment

In information retrieval, many different similarity measures are proposed to compare text representations. These three methods are widely used: the Dice coefficient, the Jaccard coefficient, and the Cosine coefficient [31]. For the representation with binary term weights, the Dice coefficient is calculated as follows:

$$Dice(x, y) = \frac{2P(x, y)}{P(x) + P(y)}$$

where  $P(x, y)$  represents the probability that  $x$  and  $y$  occur together in the same sentence, and  $P(x)$  and  $P(y)$  are the probabilities that  $x$ , respectively  $y$ , occurs separately. As a likelihood measure for a semantic triple, we use Dice coefficient, while  $x$  and  $y$  are co-occurring semantic concepts. The probability of the concepts is obtained by the training corpus described above.

We observe that any co-occurrence within a sentence is treated in the same way, no matter how far they are from each other. In reality, closer words usually have stronger relationships. The strength of the underlying relation is stronger when the distance between two concepts is shorter. Therefore, we add a distance factor  $D(x, y)$  in the

calculation. This factor decreases linearly when the distance between two concepts  $x$  and  $y$  increases. The decay factor is calculated as following:

$$D(x, y) = 1 / Dis(x, y)$$

where  $D(x, y)$  is the decay rate,  $Dis(x, y)$  is the number of words between the concepts  $x$  and  $y$  in the sentence.

We extend the previous likelihood measure method by incorporating this decaying factor. Then the reliability of the connection between concept  $x$  and  $y$  in domain field is calculated by:

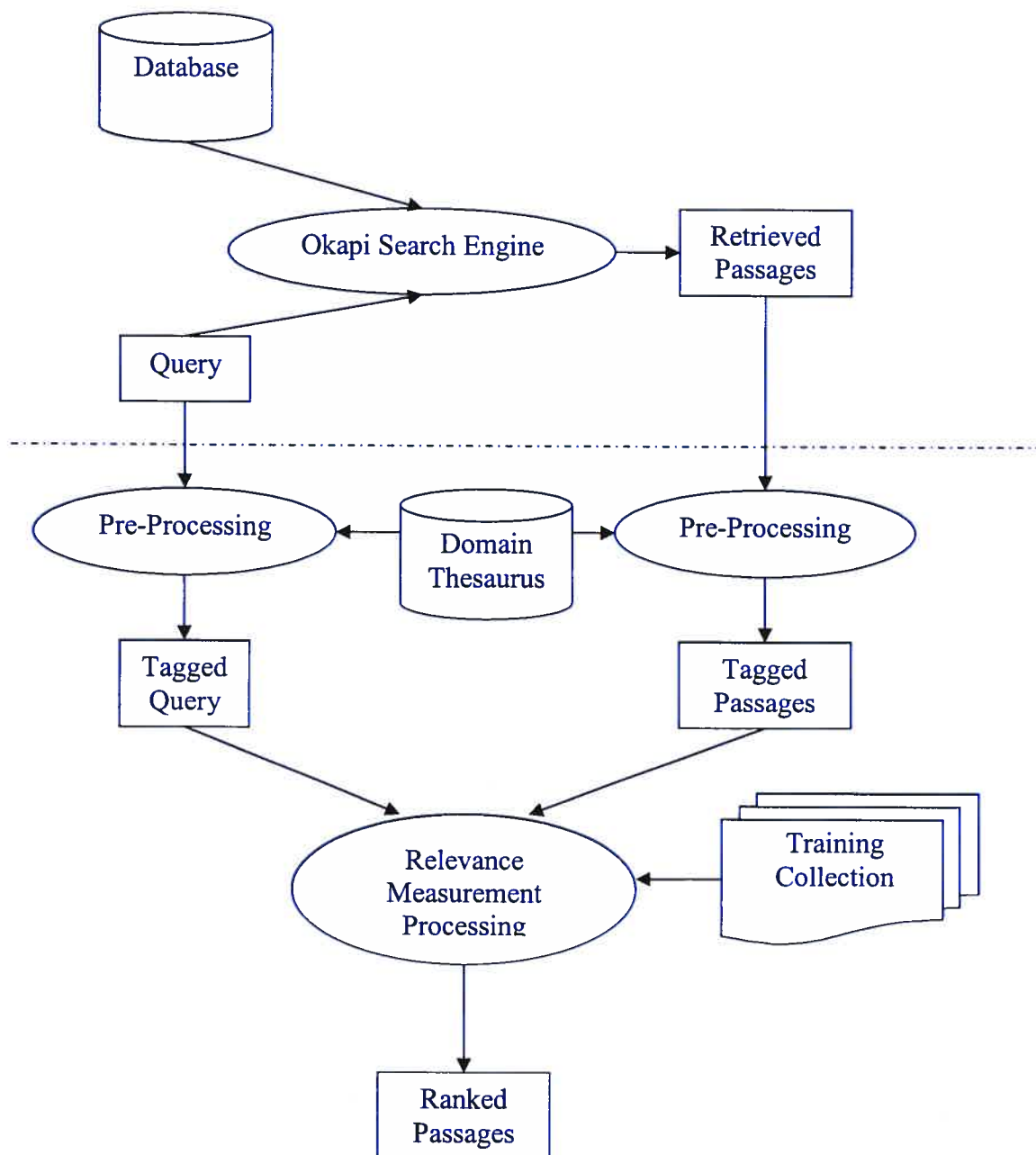
$$Reliability(x, y) = Dice(x, y) * D(x, y)$$

Then the similarity of the query and the document is measured are based on this reliability value. Detailed implementation of system relevance measurement will be given in the next chapter.

### 3.4 Framework of our system

Based on the ideas mentioned above, we developed a new approach for embedding the domain-specific knowledge into IR to improve searching performance. This system performs the IR taking into account two different levels of document representation.

The first level is the traditional keyword document representation. It serves as a filter to select documents potentially related to the user's query. The second level is formed with the semantic triples reflecting some semantic details. This second level supplies additional information about the documents. The following figure illustrates the framework of our approach.



**Figure 5: Framework of the system**

Figure 5 shows the basic architecture of the system. It consists of three main procedures and support domain-knowledge sources. We summarize here the role of the three main procedures. More detailed description of implementation will be given in the next chapter.



### 1. Passage retrieval

We use a search engine Okapi as a filter to retrieval a set of passages that are more relevant to the original query. The top 50 returned passages are taken as candidates for the next step. It allows the subsequent processes to concentrate on the relatively limited quantity of text, and to reduce the time consumption.

### 2. Pre-processing

This processing is used to identify the semantic information of the text. After the following three steps, semantic concepts and semantic triples are extracted based on the algorithm described on section 3.2.1.

- Simple word stemming: removing most common inflections; changing irregular form of nouns to their first form.
- Segmenting passages: splitting each passage into sentences. It helps the semantic analysis focus on the concepts occurring within the sentence boundaries.
- Semantic interpretation: building a semantic representation for the domain knowledge in the text. The result of this phrase is a text enhanced with the semantic representation (i.e. triples).

### 3. Relevance measurement processing

Taking the semantic representation of the query and each sentence of the passage as input, this processing performs a comparison between the sentences and the query; re-weights the degree of relevance for the passage by summing up the reliability value of the sentences. It consists of two steps:

- Entity measurement: by cooperating with the training corpus, the reliability value for each semantic entity is calculated based on the algorithms described in section 3.3.
- Passage measurement: by comparing the semantic representation of the query and the passage, the similarity measurement process re-weights the degree of relevance for each answer passage. Then the passages are re-ranked according to their new weight and displayed to the user as answer to their query.

## Chapter 4

### System implementation

In chapter 3 we have presented the ideas for our approach. This chapter describes their implementation. The implementation of the system can be divided into three parts: passage retrieval, pre-processing, and relevance measurement processing.

#### 4.1 General description of the system

In our system, we adopt different types of document representation in different phases of retrieval. In the first phase, we use the traditional keyword representation; the Okapi passage retrieval system is used to obtain a certain number of passages potentially related to the user's query. A passage can be any fragment of a text, such as a paragraph, as sentence or a window of fixed number of words. In our case, we use paragraph as passage. In the second phase, we analyze the retrieved passages more precisely by using the semantic representation, matching passages with query in semantic level, and re-rank these passages. In this phase, we aim to gain higher precision by exploiting the specific domain knowledge.

Since the passages retrieved by Okapi system include most of the passages relevant to the user's query, we need not apply semantic analysis from scratch. We concentrate on analyzing the results of the first phase, instead of analyzing all the documents in our database. Therefore, in our approach, semantic analysis techniques are used to improve the results of conventional retrieval methods, not as a replacement of conventional methods.

A typical search session involves the three steps, as outline in Figure 5.

- In the first step, the Okapi passage retrieval system serves as a keyword search tool to filter out a set of ranked passages for each query. Only the top ranked passages retrieved for each query were used in the subsequent steps. The objective of using

Okapi system is to narrow down the amount of data for the semantic information analysis. We will describe briefly the Okapi system and its integration with our system in section 4.2.

- In the second step, after the keyword search finds most relevant passages for each query, the pre-processing module constructs the semantic representations for the query and the retrieved passages, according to the text analysis process described in section 4.3.
- The last step is the relevance measurement processing module, which performs comparison between the query and obtained passages at the semantic level. Passages are re-ranked according to the relevant measurement process described in Section 4.4.

## 4.2 Okapi IR system

Okapi is an IR system which provides the platform for implementing the ideas developed in this project. While there are many alternatives in the IR literature, we chose Okapi because it allows for passage retrieval and it has demonstrated good performance in previous experiments. It is a system well suited to our task. This section will give a brief description of Okapi system and some of its general features.

### 4.2.1 Okapi system overview

Okapi is an **experimental** text retrieval system which has been under the continually experiments at City University London for last two decades [21]. Okapi is also an **interactive** text retrieval system. Interaction with system is done via different layers of interfaces built on top of the Basic Search System (BSS), which provides the lowest level of protocols to access the system. Okapi is based on a **probabilistic** retrieval model. The system predicts the probability of a given document being relevant to the user's query by calculating weights based on the Robertson/Sparck-Jones probabilistic model [20].

Okapi system comprises three basic components [21]:

- **Indexing Software** enables users to create Okapi type databases. It accepts raw text documents as input and allows the creation and indexing of databases in a form suitable for Okapi searching. For text databases made up of larger records, it is possible to generate positional information for paragraphs so that a passage search may be implemented.
- **The Basic Search System (BSS)** consists of a set of low-level commands providing efficient functionality for weighting and ranking searches. Term weighting and documents ranking is based on the Robertson/Sparck-Jones probabilistic model [32]. There is a family of built-in weighting scheme functions known as BM25 and its variants. In addition to weighting and ranking facilities, it has the usual Boolean and pseudo-Boolean (proximity) operations and a number of non-standard set operations. BSS also provides functions for blind feedback.
- **The Okapi Interactive Interface** is a configurable interface that calls BSS commands. It allows users to conduct a search on a given query formulation; view full documents and make relevance judgments; conduct relevance feedback searches; incrementally expand the query as relevance judgments are made; modify the current state of the query by adding/removing terms and clearing relevance feedback information; change some interface parameters interactively.

#### 4.2.2 Passages retrieval

One of the most important reasons for choosing Okapi as the platform of our system is that the Okapi is a passage retrieval system. That is, it allows retrieving fragments of documents (called passages) instead of complete documents. This is particularly suited to the situations where the user is interested on obtaining a piece of specific information, which is the case of our application. As mentioned in section 4.2.1, in Okapi search, the position information for paragraphs could be generated during the indexing, so that it could implement a passage search to find out the best fragments within the document.

Passage retrieval has several potential advantages in contrast to whole document retrieval.

- Firstly, Passages are more convenient to the user than long documents. For example, when a long document is retrieved, it is difficult to present it to the user and it is possible that not all parts of the document are relevant. Ideally, users should be guided to the relevant section of the document. This is the motivation of using passage retrieval. A passage could be any fragment of text in a document. In passage retrieval, query evaluation process identifies the passages in the document collection that are most similar to the query. Then the passages are returned to the user together with context information such as the titles of the documents and information about the location of the passages within the documents structures.
- Secondly, document ranking also can benefit from passage retrieval. Experimental evidence suggests that document ranking based on passages may be more effective than ranking of entire documents [22, 14, 25]. Since passages are relatively short, if the query terms occur together in the passage they must be fairly close to each other. A document, which has a short passage containing a high density of words that match a query, is more likely to be relevant than a document with no such passage, even if it contains a reasonable number of matching words across its length and has higher overall similarity. Hearst and Plaunt [22] showed that extracting the best passages from a document and adding scores for several passages produces better ranking than that based on whole-document scores. Salton et al. [14] used passages to filter out documents with low passage scores, showing that, by restricting the retrieval to those documents that have high document and high passage similarity, the retrieval result is improved by up to 22.5% compared with standard ranking. Callan [25] showed that ordering documents based on the score of the best passage may be up to 20% more effective than a standard document ranking.

### 4.2.3 BM25 formula

In Okapi, there is a state-of-the-art term weighting scheme based on the weighting formula of Robertson/Sparck-Jones. The BM25 formula used by Okapi has produced very good results, and is regarded equally with cosine correlation as the standard 'matching' function for ranked retrieval. It is currently the best performing "classical" probabilistic ranking algorithm.

The basic Robertson/Sparck-Jones weight for a term is as follows: the probability of a document indexed by the term  $t$  being relevant to a given query is calculated by the following formula:

$$W_t = \log \frac{(r + 0.5) / (R - r + 0.5)}{(n - r + 0.5) / (N - n - R + r + 0.5)}$$

where  $N$  is the number of items (documents) in the collection;  $n$  is the number of documents containing the term;  $R$  is the number of documents known to be relevant to a specific topic;  $r$  is the number of relevant documents containing the term.

In the above equation, 0.5 is added for each of the components in order to avoid indeterminate values when  $r$  and  $R$  are 0 and increase accuracy when there is little relevance information. In the absence of relevance information ( $R = r = 0$ ), as it is the case at the beginning of a search session when the user enters new search terms, the formula reduces to:

$$W_t = \log \frac{(N - n + 0.5)}{(n + 0.5)}$$

It turns out to be similar to the collection-frequency weight (idf). If the user judges some documents to be relevant, this relevance information can be fed into the formula. It may also make use of "blind" or "pseudo-relevance" feedback, where no real relevance

information is available, but an initial search is conducted and the top few documents are assumed to be relevant.

After the term weights are calculated using the above formula, the probability of a given passage being relevant to the user's query is calculated simply adding up the weights of individual query terms that index it. After the relevant score is calculated for each passage, passages are presented to user in descending order of their scores. Passages with the same weights are ordered chronologically and within that in alphabetical author order. The top-ranked passages and their relevant score are taken as input of the subsequent processing. The example below shows a fragment of the answer passages given by the Okapi system. We present the top two passages to illustrate the form of the results.

**Query:**

*How to reduce the corrosion of the reinforcing steel in garages?*

**Answer passages:**

**1. Weight 24.308**

The cause of the deterioration of parking garages is usually corrosion of the reinforcing steel due to the action of de-icing salts carried in by vehicles. Frost action seldom occurs because the temperature, even in unheated garages, usually remains above the freezing point.

**2. Weight 24.059**

If costly repairs caused by the corrosion of the reinforcing steel are to be avoided, garage decks cannot be designed and built like ordinary office building floors. Corrosion can be prevented or at least minimized by using epoxy-coated steel, low water-cement ratio concrete, waterproofing, and good drainage. Whereas each of these measures is valuable in it and will result in reduction of the rate of corrosion, several protective measures should be used simultaneously. To apply all available protective measures is unnecessary and prohibitively expensive.

## 4.3 Pre-processing

In this process, the possible semantic concepts and their relationships are extracted from the given natural language text. The system performs it by segmenting passages, stemming the words, sub-strings matching on concepts in the thesaurus, browsing the thesaurus iteratively to categorizing terms according to the algorithm described in section 3.2.1. Then the system constructs semantic triples for the text using the form as described in section 3.2.2.

### 4.3.1 Segmenting passages

Both the query and top-ranked answer passages retrieved from the Okapi system are passed to this step first. The goal of this step is to segment a passage into a sequence of sentences, allowing the semantic interpretation analysis to be performed within sentence boundaries.

We build a sentence splitter to identify sentence boundaries in the text body. Given a string of text, the sentence splitter returns a list of strings, where each is a sentence. Usually, sentence starts with a capitalized letter and finish with a full stop or other sentence delimiters. By default, the sentence splitter treats occurrences of '.', '?' and '!' as sentence delimiters, but we still pay attention to the exceptions when an occurrence of '.' does not have this role, for example, in abbreviations (Mr., Dr., etc), URLs, numbers (i.e. 10.000), etc., to make sure the splitter can correctly segment a text.

### 4.3.2 Simple word stemming

Following the passages segmentation, we use a simple word stemming. Stemming is a process for removing the common morphological and inflectional endings from words. Its main use is as part of a term normalization process. Our intention is to parse and stem input terms to convert them to the standard form used in TC/CS thesaurus, which is the domain knowledge source for semantic tagging.



The sentences resulting from the segmenting passages process are taken as input to this step. After tokenisation, the tokens are parsed to remove capitals, hyphens, punctuation and similar linguistic devices. The remaining terms are stemmed. Here we use a simple stemming, which tries to transform words (nouns) to their singular forms. For irregular words, we use an exception list consists of irregular plural nouns and their singular form to make sure that they are stemmed correctly (i.e., "wolves" to "wolf"). For regular words, we use a stemmer to convert it into its singular form. The stemmer starts with finding a terminal sub-string of the input word that is in the list of inflectional suffixes. This suffix list was prepared by hand. It includes most plural noun suffixes. After the suffix is determined, the stemmer transforms the word to its singular form by taking the stem and adding appropriate characters if necessary.

As the last step of our simple word stemming, in order to ensure the result of the stemming process, a dictionary is implemented. The stemmer checks the result against the dictionary after deduction step. After a look-up in the dictionary, this step will prevent "calories" from being converted to "calory" and many other possible mistakes.

### **4.3.3 Semantic interpretation**

The semantic interpretation module is an important part of our system. The principle of the module is already represented in section 3.2. It analyses the text and extracts the semantic information using the domain knowledge by cooperating with TC/CS thesaurus. The output of this module is a text enhanced with corresponding semantic concepts and semantic triples. In the following sub-sections, we will explain some key issues considered in the implementation of this module.

#### **Selection of the longest compound term as candidate**

During the semantic tagging, we need to compare the words within the sentence with the TC/CS thesaurus terms to identify the candidate to be tagged. We decided to select the

longest compound term as candidate. This is because the longest term is also the most specific in the construction area. The selection of the longest compound term makes it possible to identify the most appropriate thesaurus term in a sentence.

In TC/CS thesaurus terms, many of them are compound terms, which usually are built by combining two or more simple terms. If two terms may be combined into a longer term, and this longer term is also stored in the thesaurus, it is generally the case that the longer term denotes a specific meaning in the domain. The meaning of a compound term can be a combination (or not) of the meanings of the simple terms comprising it. For example, the term “cotton bag” actually refers to a bag made of cotton, whereas the term “fire wall” should not be understood literally as a wall, which protects against fire but as a computer device, which metaphorically acts as such. Regarding our application domain, it would make no sense to retrieve descriptions containing “mild” and “steel” separately in response to a query, which contains “mild steel”. In such a case, the compound term has, therefore, to be marked as acting as a single term. Based on this consideration, we use the longest-matching method in the candidate selection. For example, the “mild steel” would be identified as a concept rather than two individual concepts “mild” and “steel”, since these two concepts are also appeared in the thesaurus.

The longest matching method is well-known method to do morphological analysis and commonly applied to segment the sentence. It basically tries to get the longest dictionary entry that matches the input sentence. We use this method to scan an input sentence from the beginning, and select the longest match with the thesaurus entry. The matched terms will then be removed from the input sentence and the procedure will be repeated for the remaining terms, until nothing is left in the input sentence. Since the scanning starts from the head of the sentence, this method is also called the forward maximal matching method. It runs in a time proportional to the length of the input sentence.

### Using MySQL database as TC/CS thesaurus storage

One important component used in semantic tagging of terms in this module is TC/CS thesaurus. We need to consult the thesaurus frequently during the semantic tagging. In order to keep this step efficient, we use a MySQL database management system to store and maintain the TC/CS thesaurus. Internally, this thesaurus database consists of two MySQL tables, containing the following fields:

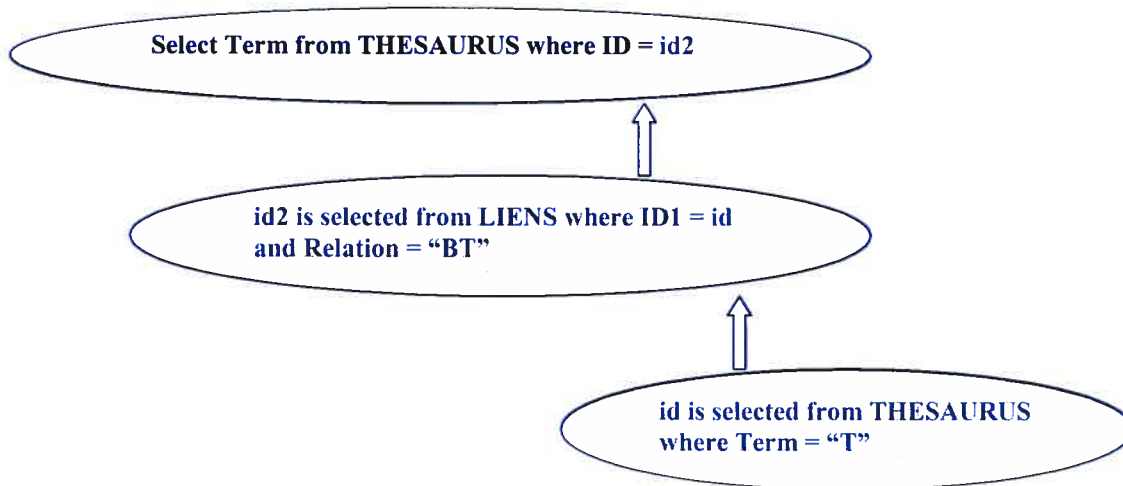
Table	Field	Description
<b>THESAURUS</b>	ID	Unique identifier of term
	Term	English form
	French term	French form of term
	Level	Hierarchical level of term
<b>LIENS</b>	ID1	Unique identifier of term1
	ID2	Unique identifier of term2
	Relation	Relationship of term1 and term2

**Table 4: Structure of the database**

- The **THESAURUS** table keeps information about each term as their identifier number, level and corresponding French term, where each term has a single entry since the table is a complete list of thesaurus terms. The ID field is the primary key for this table. It is used to uniquely refer to the term in the database.
- The **LIENS** table lists connections between terms in the thesaurus, which consists of a pair of IDs and the type of relationship between them. A term may have multiple entries, each with the different target term or the different relationship.

Using the contents of the database with SQL (Structured Query Language) scripts, we can obtain related terms using two tables. For example, finding all relationships for term T is a matter of selecting all rows in the LIENS table where ID1 equals T's ID or ID2 equals T's ID. If we are looking for a specific kind of reference, we only need to look at

one of the two. For example, to find all Broader Terms for T, we could carry out with SQL statement like the following:



### Using recursive function in semantic tagging

In order to manage all the possibilities of semantic tagging via different relationship link, the semantic interpretation module uses a recursive function to easily handle the changes of context.

In our case, the problem of semantic tagging could be broken down into a set of small, easily solvable base cases, which is to find out the related term according to the expected relationships. For example, as illustrated in Figure 1, we tag the term “stainless steel” with the semantic concept “material” following the relationship NT: “Stainless steel” → “Steel” → “Metal” → “Metallic material” → “Material”. The base case here is to find out the BT for a given term, defined as **RelatedTerm** (term T). Each time we tag a term, the system just handle with the base case, the base case calling itself recursively to handle the rest. Because different initial terms are passed to the base case each time when it is called, it searches for a different related term each time it is called. The following algorithm is the recursive function of term semantic tagging implemented in our system.

**Base case:**

**RelatedTerm**(term **T**, relationship **R**, list of predefined concepts **C**).

**Input:** A term **T**, a list of predefined concepts **C**, the relationship **R** that the inference followed.

**Output:** All possible concepts for term **T**, else **Empty**

**Algorithm:**

1. Searching all possible related term **RT** for **T** following **R**
2. For each possible related term **RT** {
  - If **RT** is a element of **C**, then return **RT**
  - Else {
    - If **RT** is not a top-level term
    - Then return **RelatedTerm**(**RT**, **R**,**C**)
    - Else return **Empty**

**Figure 6: Algorithm of the recursive function for semantic tagging**

In the first step, we use SQL language in cooperating with the MYSQL database management system, as described in the previous section, to obtain all the possible related terms for the candidate term **T**. Step 2 is the main part of the function corresponding to the basis clause of the recursive definition. It loops through each possible related term; performs some test on its arguments to check some necessary condition for recursively calling the base case.

**Construct the semantic triple**

After the semantic concepts have been tagged, the system constructs semantic triples for the sentences using the form as described in section 3.2.2, which is reproduced below:

Co-occurrence (concept1, concept2)

We take each pair of concepts within a sentence to form this triple. In order to calculate the reliability value, which is related with the distance between two concepts, we also preserve the position information for each concept. With all these necessary data, the subsequent processing performs the relevance measurement for the system.

## 4.4 Relevance measurement processing

Taking both query and candidate answer passages with their pre-processed results as input, this process performs a comparison between the sentences and the query; calculates the reliability scores of the concepts and semantic triples according to the formula described in section 3.3.2; sums up entity scores for each sentence; re-weights the degree of relevance for each passage, then the final step returns the re-ranked passages to be used as the answer.

Once the previous processing has extracted the semantic information for query and candidate answer passages, this module evaluates the relevance between them in semantic level. As we mentioned above, our approach tries to use semantic information to improve the performance of conventional retrieval, not as a replacement. From this perspective, it is necessary to have the appropriate methods to adjust the conventional retrieval result after the comparison of two texts in their new representation.

### 4.4.1 Entity measurement

We have two types of semantic entities: semantic concepts and semantic triple. As described in section 3.3.2, we compute the reliability weight of each semantic concept in the training corpus that is described in section 3.3.1, and calculate the reliability weight for semantic triples using the formula:

$$\text{Reliability}(x, y) = \text{Dice}(x, y) * D(x, y)$$

where  $Dice(x, y)$  is the Dice coefficient for semantic concepts  $x$  and  $y$  and  $D(x, y)$  is the decay rate that is related with the distance between these two concepts, i.e., number of words between them.

Each entity in a candidate answer sentence (a semantic concept or a semantic triple) receives a score by comparing the semantic similarity to the query. Each entity in the query is a “constraint”: the entity in the answer sentence that is different from the constraint will be assigned zero as its score. Otherwise, the entity gets the reliability weight as its entity score. The candidate answer sentence gets a sentence score from each entity it contains.

#### 4.4.2 Passage measurement

When the sentence score calculation is finished, each passage that consists of several sentences is assigned a passage score by summing up the sentence scores of all the sentences in the passage.

Since there are two type of the entity: semantic concept and semantic triple, in the semantic similarity measurement, we have two choices: similarity measure score for the semantic concepts ( $S_c$ ) and similarity measure score for the semantic triples relation ( $S_r$ ). Both are the sum of the corresponding similarities of the elements contained in the passage. These two measures are then combined into a cumulative semantic similarity measure score  $S$ .

Our first combination is multiplicative, i.e.,  $S = S_c * S_r$ . However, we note that the semantic triple similarity has a secondary importance, because its existence depends on the existence of semantic concepts. In other words, a semantic triple similarity implies semantic concept similarity of its components. Under this consideration, the cumulative score is proportional to  $S_c$ . However,  $S$  still should not be zero when  $S_r = 0$ . So we smooth the effect of  $S_r$  using the following formula:

$$S = S_c * (1 + \beta * S_r)$$

With this definition, if no triple relational similarity exists in a sentence ( $S_r = 0$ ) then the semantic similarity only depends on the value of the conceptual similarity. In this situation, the semantic similarity is exactly the conceptual similarity. Otherwise, the semantic similarity is the conceptual similarity plus a fraction of the relational similarity, where the coefficient  $\beta$  indicates the value of this fraction.

After obtaining the semantic similarity measure score, we can use it to adjust the original relevance weight obtained from Okapi system for each answer passage. The new passage weight is computed by the following formula:

$$\begin{aligned} \text{New weight} &= \text{Okapi-weight} * (1 + \alpha * S) \\ &= \text{Okapi-weight} * (1 + \alpha * S_c * (1 + \beta * S_r)) \end{aligned}$$

The coefficients  $\alpha$  and  $\beta$  reflect user-specified balance. Their values range from 0 to 1. The coefficient  $\alpha$  indicated the importance of the part of the similarity exclusively dependent on the common concepts, and the coefficient  $\beta$  expressed the importance of the part of the similarity related with the connection of these common concepts. The choice of the coefficients allows adjusting the similarity measure to the different applications and interests. For instance, when  $\alpha > \beta$ , the conceptual similarities are emphasized, while when  $\alpha < \beta$ , it stresses structural similarities.

These values of  $\alpha$  and  $\beta$  have been estimated empirically. In the current implementation, the coefficients are static values; we compare the results of system performance of different coefficients; then choose the best solution from the best result. It turns out that the best value for both  $\alpha$  and  $\beta$  is 0.2. A major step in the estimation of these two coefficients therefore is the determination of test collection used in estimation. In section 5.1.3 of this thesis we describe the data used in estimation.

Finally, all candidate answer passages retrieved by Okapi system are re-weighted using the formula just mentioned, passages are sorted by the new passage weight and the



system outputs the top 50 re-ranked passages as final result. The following example illustrates the results of the pre-processing and relevance measurement processing for the potential answer passages for the query we mentioned in section 4.2.

**Query:** *How to reduce the corrosion of the reinforcing steel in garages?*

**Semantic interpretation:**

- Semantic concepts:
  - “corrosion” → [degradation of material]
  - “steel” → [material]
- Semantic triple:
  - Co-occurrence (material, degradation of material)

**Relevance measurement:**

- Passage 1:
  - Co-occurrence (material, degradation of material) Probability=0.019273838
  - weight = 24.308
  - new weight =  $\text{weight} \cdot (1+P) = 24.77651$
- Passage 2:
  - Co-occurrence (material, degradation of material) Probability=0.09198877
  - Weight =24.059
  - new weight =  $\text{weight} \cdot (1+P) = 26.272158$

We can see here that once the semantic aspects are considered, the second passage, which was ranked lower than the first passage by Okapi, is now ranked higher.

## Chapter 5

### System evaluation

In this chapter, we present an evaluation of our system. Firstly, we briefly review the experimental methods in IR, and then introduce the test data used in the system evaluation. We present also the general discussion of the various parameters and decisions involved in the system evaluation, analysis of the results of the experiments performed.

#### 5.1 Evaluation in IR

The major criterion of quality of an IR system is retrieval effectiveness. This should reflect the ability to which a system is able to retrieve relevant documents and to reject non-relevant ones.

##### 5.1.1 Traditional evaluation methods

The standard measures of retrieval effectiveness are *precision* and *recall*. When taken together recall and precision provide a useful measure of the system's performance. Assuming that:

- RET is the set of all texts the system has retrieved for a specific inquiry;
- REL is the set of relevant texts for a specific inquiry;
- RETREL is the set of the retrieved relevant texts, i.e.  $RETREL = RET \cap REL$ .

then precision and recall measures are obtained as follows:

$$precision = RETREL / RET$$

$$recall = RETREL / REL$$

In words, recall is the proportion of relevant documents in the collection that the system assigns to the query; precision is the proportion of the documents assigned to the query by the system correctly. An ideal system would have 1 for both recall and precision.

In all retrieval systems, precision and recall generally vary inversely with each other. With limited document representations, it does not seem possible to search the document collection for relevant documents without retrieving increasingly larger proportions of non-relevant documents. This inescapable fact will affect the way in which the user deals with the ranked retrieval output. A high-precision search will typically involve the user assessing the relevance of the top few retrieved documents and being satisfied with the one or two most relevant items in the collections. Alternatively, a high-recall search may involve the user assessing a larger number of the initially retrieved documents, then reformulating the initial query using relevance feedback, and searching deeper in the collection for relevant items. Precision and recall are popular and useful measures, because they give a direct indication of the retrieval system parameters that is likely to be of interest to the user. The advantages of precision and recall as measures of retrieval effectiveness are that they are highly intuitive and easy to calculate. However, there are several disadvantages to their use. One of the most important problems is that for any set of retrieved documents, retrieval effectiveness must be expressed as a precision-recall pair.

### 5.1.2 Average precision and recall

For evaluating an IR system performance, it is a common practice to perform retrieval for a number of queries and then to pool the results obtained on each query to obtain some average indicator of performance over the set of queries.

Van Rijsbergen [24] identified two different methods for cross-query averaging: *Predictive* and *Descriptive*. In the predictive method, precision values are pooled and averaged for fixed recall levels irrespective of the real precision-recall pairs produced by

each query. Conversely, in the descriptive method, cross-query correspondence is based on some variable underlying parameter common to both queries, such as the number of documents retrieved. In the TREC scoring software, both methods are employed: the predictive method generates precision levels for fixed recall values between 0 and 1; the descriptive method produces precision scores after the retrieval of  $n$  documents.

To pool precision-recall curves over the query set to obtain a predictive average performance curve, two methods could be used: *Micro-averaging* and *Macro-averaging* [23]. Micro-averaging considers all queries as single group and calculates recall, precision as defined above. On the other hand, Macro-averaging, computes these measures separately for all documents associated with each single query, and then computes the mean of the resulting effectiveness values [10].

### 5.1.3 Evaluation with test collections

To establish the retrieval performance of an IR system, it is necessary to use a test collection. The existence of test collections brings the advantage of repeatability and controllability, which makes it possible to compare the results across different systems or retrieval methods.

The requirement for a suitable test collection for text retrieval was recognized early. It is specifically created for evaluating experimental IR systems. Such a collection consists of: a set of documents; a set of standard queries; and for each query, a list of the documents relevant to that query. These relevant document lists are manually identified, a process that involves significant human effort. In recent years a set of large test collections have been created which are approximately 4GB in size. These collections are collectively known as the TREC collection. TREC test collections are increasingly being used for different investigations. The popularity of TREC has been demonstrated by a number of recent conferences.

In our project, a small test collection is used in the evaluation exercises. This test collection was created specifically for the construction domain to assist the IR performance evaluation processing for our system. It is a construction-oriented subset, consisting of 1) document collection of Canadian Building Digest, published between 1960 and 1990 by NRC's Institute which take up about 5 Megabytes, 2) 50 queries generated by construction experts with a examination on the contents of documents collections, and 3) each query is associated with a short passage that is judged as the correct answer for the query. The particularity of this test collection is these queries are very specific questions about the professional building design and construction technique, covering the essential aspects of construction area, so that the experts give one answer for each query.

The evaluation of the system performance is executed by comparing the system's output with standard answer passage.

## 5.2 Evaluation experiment

Usually in IR, system effectiveness is reflected by a single value, the average precision across the 11-point recall levels. Typical standard recall levels, referred to as 11-point levels, are 0%, 10%, ... 90%, and 100%. In our approach, since we just have one relevant answer, we use 100% point precision as system precision. According to their definition, we obtained:

**Recall** = 100 %;

**Precision** = Number of relevant passages retrieved/ Number of total passages retrieved  
= 1/ Position of the correct answer passage in the answer list

**System Performance** = sum of precisions for all queries

To evaluate the retrieval effectiveness of our system, the test collection contains 50 queries with one correct answer passage for each query. We calculate precision for each query, then sum up for a total value as the system performance. The evaluation exercise consists of two parts.

The first part of the evaluation exercise aims to determine the coefficients  $\alpha$  and  $\beta$  in the passage re-weighting formula. Generally, there is no simple and straightforward way of doing this that guarantees the accuracy of the estimates. In our application, the coefficients have been estimated empirically. We compare the results of system performance of different coefficients. Experimental results show that two coefficients be assigned the value 0.2, the search performance obtained the best result.

The second part of the evaluation exercise consists of three tests. The purpose of tests is to have an in-depth analysis of the functionality of the semantic information in IR search. We perform the tests by selecting different semantic representation forms in each test. Analyzing the results obtained the efficiency of different semantic information.

- Test 0: the original search result data from the Okapi search in order to compare with the other tests.
- Test 1: using semantic triples in semantic retrieval, evaluate the efficiency of the semantic triple in search.
- Test 2: using concepts in semantic retrieval, evaluate the efficiency of the semantic concepts in search.
- Test 3: using all the semantic information, including the concepts and semantic triples together, examine the functionality of the semantic information.

The table below shows the results obtained from the tests. The numbers with star signal indicate that there is no added semantic information available for this query, so that the result is the same as the keyword-search.

Query No.	Position of the correct answer			
	Test 0	Test 1	Test 2	Test 3
1	2	5	6	7
2	15	15*	13	13
3	1	1	1	1
4	3	3	3	3
5				
6	1	1	1	1
7	1	1*	1*	1*
8	9	7	17	17
9	1	1	1	1
10	20	20	18	18
11	1	1*	1	1
12	44	44	43	43
13	3	2	1	1
14	1	1	1	1
15				
16	5	4	5	5
17	1	1*	1	1
18	3	3	5	5
19	1	1	1	1
20	1	1	1	1
21				
22	14	14*	14	14
23	1	1	1	1
24	1	1*	1	1
25	5	2	2	3
26	3	3*	9	9
27	1	1*	1*	1*
28	2	2	2	2
29	5	5*	1	1
30	1	1	1	1
31	2	1	1	1
32	1	1*	1	1
33	1	1*	1	1
34	3	3	3	3
35	7	7	12	12
36	2	2*	2	2
37	3	1	2	1
38				
39	44	44*	41	41
40	2	2*	3	3
41	9	9*	6	6
42	3	3*	1	1
43	27	27*	24	24
44	1	1	1	1
45	1	1	1	1
46	3	3*	3	3
47	4	4*	4*	4*
48	20	13	21	16
49	1	1	1	1
50	10	10	10	10

Table 5: Results-1 of system evaluation

### 5.3 Discussion

Table 6 summarizes the results of the system performance values obtained and the improvements of the system performance achieved in different tests.

	Test 0	Test 1	Test 2	Test 3
System Performance	Sum = 24.80	Sum = 26.24	Sum = 27.01	Sum = 27.33
Improvement		5.8 %	8.9 %	10.2 %

**Table 6: Results-2 of the system evaluation**

1. The test 1 employed semantic triples directly. The increase of the system performance is not significant. This is mainly due to the fact that there is only one semantic concept extracted from the sentences in a number of queries. Therefore, no semantic triple is formed in these cases. This situation occurs in approximately 40% of all queries.
2. The second test employed semantic concepts in semantic retrieval. The system performance is increased by 8.9 %. The use of the semantic concepts instead of the semantic triple has solved the problem mentioned above.
3. The third test has the best result in increasing the IR performance because we use the semantic triples and semantic concepts together in semantic retrieval. The system performance brings an improvement of 10.2 %.

Table 7 reports the detailed analysis results performed on the tests.

		Test 1	Test2	Test3
1	No answer	4	4	4
2	No semantic information	19	3	3
3	Up	7	13	13
4	Down	1	7	7
5	No change	8	9	9

**Table 7: Results-3 of the system evaluation**



Here are the explanations for the table 7:

1. The first line indicates that there are 4 cases that Okapi system could not find the correct answer passage for the query.
2. The second line of the table 7 shows the number of the queries that semantic information could not be extracted from query text, meaning that there is not semantic triple or semantic concepts in query text. This problem is primarily caused by the insufficient coverage of the TC/CS thesaurus.
3. The third, fourth and fifth lines of the table 7 show the number of the queries that the position of correct answer passage has been changed in different directions. The numbers for test 1 show that even the simplified semantic information could help the search: 65% of them changes toward the expected direction, while there are less negative effect or no effect on the other queries.

After analyzing the failure cases, we observed several weaknesses which might be the causes.

- There is an insufficient coverage of the TC/CS thesaurus because there are still concepts missing in the thesaurus.
- Another reason is that although the TC/CS thesaurus is highly connected, it is often the case that desirable links, at least for our purposes, were missing. In fact, among the concepts in the thesaurus, many concepts are not connected with any other concept through the hierarchical relationships, meaning that knowledge of concept associations could not be applied in those cases. Thus leads the semantic tagging to failure.
- The semantic tagging may be imprecise. The accuracy of the determination of the semantic concepts can have an important impact on the semantic information expression. In our project, the definition of the concepts is subjective: it has been set up by the experts.
- The ambiguity of the semantic triple: co-occurrence relationships used in semantic triples could not embody the genuine relationships between semantic concepts. More genuine relationships that really express the way that concepts are interrelated and more sophisticated method for identifying the concepts are required.

## Chapter 6

### Conclusion

In this final chapter, we summarize the conclusion drawn from this study, and points to some future research directions and questions.

#### 6.1 Project

Our goal is to find a flexible way in which domain knowledge can easily be incorporated into semantic information determination, and to explore the possibility of extending traditional IR systems with knowledge-based approaches to improve the retrieval performance. In our approach, we addressed the issues concerning the application of the domain knowledge and IR at a semantic level. We developed a knowledge-based application, which exploits the domain knowledge by using a large, pre-build, technical thesaurus. Combining with simple AI techniques, our approach can conduct search at a semantic level and improve the system precision.

#### 6.2 Basic approach

In order to exploit domain knowledge into information retrieval, in our project, we used TC/CS thesaurus as conceptual vocabulary to identify the underlying domain concepts. We simplified CG to construct semantic triples, which are employed in our project as the representation language to describe semantic information of a text. A collection of articles published in the construction field, Canadian Building Digest, is also used as a training corpus to estimate the co-occurrence information among the domain concepts. A set of construction-oriented queries associated with correct answer passages that generated by construction experts is utilized for evaluating the system performance. By incorporating several domain-specified knowledge sources, our approach has found a flexible way in which domain knowledge may be embedded into information search.

Experimental results showed that an increase in retrieval performance can be obtained by using this approach.

### 6.3 Results

The evaluation experiments performed aimed at finding out whether the semantic information was actually effective in improving the IR performance and whether the domain-specific knowledge source was useful in semantic information extraction. The following observations are the main result of the experiments:

- An increase in retrieval performance can be obtained by merging semantic information into search.
- Domain-specific knowledge is useful in extracting the semantic information.
- The knowledge-based system retrieval results in higher effectiveness in term of precision than the basic Okapi system.

Our results show that even the simplified semantic information could improve the IR performance. The experimental results suggest that the simplified semantic information has had only minimal negative effect on some cases, while significantly increasing precision of the whole system. In other words, the added semantic information is apparently of satisfactory quality, and allows a significantly improvement of IR performance during the search.

The success of this application relies on several factors: the quality of the underlying knowledge bases (the TC/CS thesaurus and other domain-knowledge sources), the semantic tagging algorithm for domain-specific concepts, the semantic expressing of the domain knowledge and relevance measurement for the semantic information.

### 6.2 Future directions

This project is very preliminary and as such there are many ways to improve it. Here we will discuss briefly some of the ideas that may deserve future research attention.

- **Semantic information extraction**

The simplified CG representation is used to encode semantic concepts and their relationships in our approach. The choice of simplified CG is essentially due to its simplicity, easy implementation. When compared to keyword techniques, the domain knowledge semantic information about text is relatively rich. However, this representation still has some problems.

Firstly, although in our project, we use the co-occurrence relationship to replace all other kind of relations, co-occurrence relationship may not embody the genuine relationships between semantic concepts.

- One possibility to solve this problem is to extend the co-occurrence relationship to include some domain-specific semantic relations. For example, by defining the relationship: *made-of* between two semantic concepts: *material* and *building*, and *attribute* between the *material* and *degradation of material*, to distinguish the different characteristics of construction material. This could contribute in eliminating some ambiguity of the semantic triple.
- Another possibility to address this issue would be to develop a new mechanism to find out and represent the real relationship. This new mechanism could be either an application of NLP procedure or other cognitive tools. Then we could use more elements of the CG formalism in forming the semantic representation.

Secondly, in relevant measurement module, while we evaluate the semantic relatedness between the query and candidate passages, the uncertainty of the original term belongs to its semantic concepts have not been taken into account. A possible solution to estimate this uncertainty is to measure the semantic distance between the original concept and its concept. It might be computed by counting the links between them in the thesaurus graph. Intuitively, a short path between concepts in the thesaurus graph might be expected to correspond to some loose notion of relevance between those concepts.

- **Domain specific ontology**

The TC/CS thesaurus plays an important role in our study; it served as the main domain knowledge source in the semantic information extraction. However, this thesaurus is not sufficient. The thesaurus can only represent partial semantic relationships, i.e. simple hierarchical, equivalent, and associative relationships. As we discussed in chapter 2, ontology provides a more detailed, formal knowledge representation language that provides a better representation of word meaning. The relationships in ontology are more complete and useful. They are formally defined and are unambiguous. They are supposed to cover all the ways in which words can be interrelated. With the rapid growth in the development of the domain ontology, it is possible to enhance the TC/CS thesaurus with construction domain ontology.

- **Context and documents meaning**

In this approach, we processed the semantic information extraction within the sentence boundary. For a better result, this approach may need to be integrated into further level of processing that take context into account and encode document meanings instead of sentence meaning. This means that some relationships across sentences should also be considered.

## References

- [1] Jadranka Lasic-Lazic and Sanja Seljan and Hrvoje Stancic, *Information Retrieval Techniques*, Faculty of Philosophy, Department of Information Science, 2000
- [2] Hele-Mai Haav, Tanel-Lauri Lubi, *A Survey of Concept-based Information Retrieval Tools on the Web*, Institute of Cybernetics at Tallinn Technical University, 2001
- [3] P. Clarke and J. Thompson and H. Holmback and L. Duncan, *Exploiting a Thesaurus-Based Semantic Net for Knowledge-Based Search*. In Proceedings of the 12th Conference on Innovative Applications of AI, pages 988-995, 2000
- [4] John F. Sowa, *Building, Sharing, and Merging Ontologies*, 2003, <http://www.jfsowa.com/ontology/ontoshar.htm>
- [5] K. Sparck-Jones, *Notes and references on early automatic classification work*. SIGIR Forum, 25(1): 10-17, 1991
- [6] Manuel Montes y Gomez and Aurelio Lopez and Alexander Gelbukh, *Information Retrieval with Conceptual Graph Matching*, In Database and Expert System Applications, Greenwich, England, pp 312-321, 2000
- [7] Howard Beck and Helena Sofia Pinto, *Overview of Approach, Methodologies, Standards, and tools for Ontologies*, The Agricultural Ontology Service, 2002
- [8] J.Y. Nie, *A General Logical Approach to Inferential Information Retrieval*, Encyclopedia of Computer Science and Technology, Ed. A. Kent and J.G. Williams, Vol. 44: 203-226, 2001

- [9] J.Y. Nie and M. Brisebois, *An inferential approach to information retrieval and its implementation using a manual thesaurus*, *Artificial Intelligence Review*, 10: 409-439, 1996
- [10] Davis D. Lewis, *Evaluating text categorization*, *Proceedings of the Speech and Natural Language Workshop, Asilomar*, pp 312-318, 1991
- [11] H. P. Luhn, *The automatic creation of literature abstracts*, *I. B. M. Journal of research and Development* 2(2), pp 159--165, 1958
- [12] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill New York [etc.] 1983
- [13] G. Salton, ed. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*, Addison Wesley Publishing, 1989.
- [14] G. Salton and J. Allan, and C. Buckley, *Approaches to passage retrieval in full text information systems*. *Proceeding of the 16<sup>th</sup> Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp 49-58, 1993
- [15] T. M. Mitchell, *Machine Learning*, McGraw-Hill International Editions, 1997
- [16] John F. Sowa, *Conceptual Graphs: Drafts of the proposed ISO standard*, From ISO/JTC1/SC 32/WG 2. Reference: ISO/JTC1/SC 32/WG2 N 000. 2001.
- [17] John F. Sowa, *Knowledge Representation: Logical, Philosophical and Computational Foundations*, Brooks Cole Publishing Co, Pacific Grove, CA, 2000
- [18] J. Martinet and Y. Chiamella and P. Mulhem, *Un modèle vectoriel étendu de recherche d'information adapté aux images*, 20ème Congrès INFORSID'02

(Informatique des Organisations et Systèmes d'Information et de Décision), Nantes, France, pp337-348, 2002.

[19] David A. Smith MA, *Use of a thesaurus in two-stage information retrieval of electronic records*, <http://europa.eu.int/ISPO/dlm/dlm96/proceed-en4.pdf>

[20] S. Robertson and K. Sparck Jones, *Relevance Weighting of Search Terms*, Journal of the American Society for Information Science, 27(3): 129-146, 1976

[21] S. E. Robertson, *Overview of the Okapi projects*. Journal of Documentation, 53(1): 3--7, 1997

[22] M. A. Hearst and C. Plaunt, *Subtopic structuring for full-length document access*, Proceedings of the 16<sup>th</sup> Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, pp 59-68, 1993

[23] Y. Yang, and X. Liu, *A Re-Examination of Text Categorization Methods*. In Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval, pp 42-49, 1999

[24] CJ van Rijsbergen, *Information Retrieval*, Butter Worths, 1997

[25] J. P. Callan, *Passage-retrieval evidence in document retrieval*, Proceedings of the 17<sup>th</sup> Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, pp 302-310, 1994

[26] Gail Hodge, *Systems of Knowledge Organization for Digital Libraries: Beyond Traditional*, Authority Files, CLIC, April 2000

[27] Howard Beck, Helena Sofia Pinto, *Overview of Approach, Methodologies, Standards, and tools for Ontologies*, The Agricultural Ontology Service, 2002



[28] Sowa, John F. *Conceptual Structures: Information Processing in Mind and Machine*. Ed. Addison-Wesley, 1984

[29] NISO, The National Information Standards Organization, 1994, <http://www.niso.org/>

[30] Kilgarriff, Adam and Colin Yallop, *What's in a thesaurus?* In Proceedings of LREC, pp 1371-1379, 2000

[31] Gerard Salton, *Automatic text processing - the transformation, analysis, and retrieval of information by computer*, Addison Wesley, 1989.

[32] S.E. Robertson and Sparck Jones K. *Relevance Weighting of Search Terms*, Journal of the American Society for Information Science, 27(3), 129-146, 1976.

## Appendix

Descriptors (level 4)	NO OF <sup>1</sup>		subj Rank <sup>2</sup>
	NT/PT	RT	
ACCOUNTING	10	1	5
ACOUSTICS	8	1	9
AD HOC APPROACH	0	0	3
ADMINISTRATIVE LAW	6	1	4
AERONAUTICS	0	1	3
AFFINE GEOMETRY	0	0	2
ALGEBRAIC GEOMETRY	4	1	3
AMBIENT ENVIRONMENT	15	8	10
ANALOGY LAW	0	2	3
ANIMAL COMMUNITIES	0	2	4
ANIMAL ECOLOGY	0	3	5
ANIMAL HUSBANDRY	0	0	4
ANIMALS	9	3	7
APPLIED ACOUSTICS	8	1	9
APPLIED ARTS	1	0	5
APPROPRIATE TECHNOLOGY	0	1	8
ARBORICULTURE	0	0	4
ARCHEOLOGY	0	1	4
ARCHITECTURAL HISTORY	2	5	6
ARCHITECTURE(ACTIVITY)	3	6	9
ARITHMETIC OPERATIONS	0	0	2
ART HISTORY	0	3	3
ARTISTIC CEATION	1	2	2
ASSOCIATIONS	7	1	7
ASTRONOMY	0	0	1
ATMOSPHERE	7	3	7
ATMOSPHERIC PHENOMENA	20	4	8
ATMOSPHERIC PHYSICS	8	5	7
AUTHORITY	4	4	5
BACTERIOLOGY	2	0	2
BARTERING	0	0	1
BEEKEEPING	0	0	1
BEHAVIOUR	5	3	7
BIOCENOSIS	0	1	2
BIOCHEMICAL COMPOUNDS	0	1	2
BIOCHEMICAL CYCLES	1	1	2
BIOCHEMICAL REACTIONS	2	1	4
BIOELECTRICITY	0	1	1
BIOMECHANICS	0	1	1
BIOSIS	0	1	1
BOOLEAN ALGEBRA	3	1	2
BUILDING ECONOMICS	6	6	9
BUILDING MECHANICAL ENGI	4	2	6
BUILDING PROCESS	5	3	10

<sup>1</sup> These numbers are based on a count in the TC/CS

<sup>2</sup> Subjective evaluation based on the likely needs of Cibât's clientele (10=high)

BUILDINGS	24	5	10
BUILT UP AREAS	1	6	6
BUSINESS	3	3	6
BUSINESS MANAGEMENT	8	5	8
BYLAWS	2	0	8
CAREERS	39	2	10
CASE LAW	0	0	1
CELLS(ORGANISMS)	2	0	1
CHEMICAL ANALISIS	4	4	4
CHEMICAL COMPOUNDS	8	4	6
CHEMICAL ELEMENTS	2	2	3
CHEMICAL ENGINEERING	0	1	4
CHEMICAL FUNCTIONS	9	3	5
CHEMICAL INDUSTRY	1	1	2
CHEMICAL REACTIONS	21	3	8
CIRCULATORY SYSTEM	1	0	1
CITIES	8	7	9
CITIZENS	0	0	3
CIVIL DEFENCE	2	1	8
CIVIL ENGINEERING	11	7	10
CIVIL ENGINNEERING WORKS	10	5	10
CIVIL LAW	4	1	4
CLIMATOLOGY	6	3	7
COMBINATORIAL ANALYSIS	4	1	3
COMMERCIAL LAW	2	1	5
COMMUNICATION THEORY	3	0	2
COMPANIES	18	3	9
COMPUTER SCIENCE	4	1	3
CONSTITUTIONAL LAW	0	1	1
CONSTRUCTION INDUSTRY	5	4	10
CONSTRUCTION TECHNOLOG	11	5	10
CONSTRUCTIONS	32	5	10
CONSUMPTION	2	2	4
CONTEMPORARY ART	0	0	1
CONTINENTAL MASSES	0	2	1
CONTINUOUS FIELD MECHAN	3	2	3
CONTRACT LAW	0	2	6
CONTRACTS	16	3	9
CONTROL THEORY	4	1	3
CONURBATIONS	0	2	5
COPYING PROCESS	4	0	2
CORRELATION(STATISTICS)	0	2	2
CRAFT PRODUCTS	2	0	2
CRIMINAL LAW	2	1	1
DECISION THEORY	1	5	4
DEGRADATION OF MATERIAL	5	2	7
DEMOGRAPHY	3	2	4
DESCRIPTGIVE GEOMETRY	0	0	1
DESIGN	12	7	10
DETERMINISTIC APPROACH	0	0	1
DIFFERENTIAL GEOMETRY	1	0	1
DIGESTIVE SYSTEM	0	0	1
DISTRIBUTION	5	4	7
DISTRIBUTION LAW (STATIST	4	1	4
DWELLING UNITS	0	0	7
EARTH CORE	0	0	1

EARTH CRUST	2	0	1
EARTH SURFACE	3	3	4
ECOLOGICAL LIFE CYCLES	0	2	3
ECONOMIC ANALYSIS	7	4	7
ECONOMIC CONDITIONS	6	2	7
ECONOMIC GEORGAPHY	2	4	3
ECONOMIC GEOLOGY	3	2	3
ECONOMIC POLICIES	10	6	8
ECONOMIC THEORY	3	1	3
ECONOMIC VALUE	7	2	6
ECOSYSTEMS	3	4	8
EDUCATIONAL INSTITUTIONS	9	0	8
ELASTIC SOIL MECHANICS	4	3	5
ELECTRICAL ENGINEERING	3	1	8
ELECTRICITY(THEORY)	7	4	7
ELECTROCHEMISTRY	3	1	4
EMPLOYMENT	19	3	9
ENERGY CONSERVATION	0	0	8
ENVIRONMENTAL CONTROL	7	3	9
EQUAL REPRESENTATION IN	3	0	2
EQUATIONS(MATHS)	9	0	3
EQUIPMENT	20	4	10
ERROR ANALYSIS	10	5	4
ETHNOLOGY	2	1	1
EVERY DAY LIFE	0	0	1
EXPERIMENTAL METHODS	31	4	10
EXPERIMENTAL PROCEDURE	1	0	6
EXPONENTS	0	0	1
FACILITIES	22	6	10
FACTORIAL ANALYSIS	0	0	1
FAMILIES	2	3	5
FAMILY SOCIOLOGY	0	1	2
FINANCES	7	5	7
FINANCIAL INSTITUTIONS	6	0	6
FINANCING	8	2	8
FINITE DIFFERENCE THEORY	2	0	1
FISCAL POLICY	3	5	7
FISH FARMING	0	0	2
FLUID MECHANICS	7	2	8
FOOD PROCESSING INDUSTRY	1	0	3
FOREIGN TRADE	4	2	6
FORM PSYCHOLOGY	2	1	3
FUNCTIONAL ANALYSIS(MATHS)	1	1	1
FUNCTIONS(MATHS)	18	3	3
GADGETS	0	2	1
GAME THEORY	3	5	3
GENERAL PSYCHOLOGY	0	0	1
GEOCHRONOLOGY	0	0	1
GEOELECTRICITY	2	1	2
GEOGRAPHICAL COORDINATES	0	0	1
GEOGRAPHICAL DIVISIONS	8	1	4
GEOMAGNETISM	2	1	2
GEOMETRIC SOLIDS	0	2	1
GEOMETRIC SURFACES	1	1	1
GEOMORPHOLOGY	9	3	8
GEOTECHNICS	5	2	7

GEO THERMICS	1	0	2
GERIATRICS	1	0	2
GOVERNMENT POLICIES	2	7	5
GOVERNMENTS	9	4	7
GRAIN FARMING	1	0	3
GRAMMAR	0	0	1
GRAPHIC SEMIOLOGY	1	1	1
GRAPHICAL ANALYSIS	1	1	1
GROUP THEORY(MATHS)	3	1	1
HABITATS(ECOLOGY)	1	5	4
HEALTH	4	1	4
HEURISTIC APPROACH	0	1	1
HIGH TECHNOLOGY	0	0	1
HISTORIC PRESERVATION	2	6	8
HISTORICAL HERITAGE	3	1	8
HOUSING	10	2	10
HOUSING ECONOMICS	1	2	7
HUMAN BEINGS	3	7	7
HUMAN COMMUNICATION	2	0	3
HUMAN COMMUNITIES	2	3	5
HUMAN ECOLOGY	2	3	4
HUMAN RESOURCES	1	2	4
HYDRAULICS	5	2	6
HYDROGEOLOGY	2	1	3
HYDROSPHERE	1	1	1
INCOME	4	3	5
INDUSTRIAL ECONOMICS	5	4	5
INDUSTRIAL SOCIOLOGY	2	3	3
INDUSTRIALIZATION	5	8	8
INFRASTRUCTURE	3	3	7
INTERIOR DESIGN	3	3	6
INTERNAL TRADE	2	1	3
INTERNATIONAL LAW	0	2	2
INVENTORIES	0	0	2
JURISDICTION	1	2	2
LABOUR LAW	5	1	5
LAND DEVELOPMENT	7	2	7
LAND ECONOMICS	4	5	7
LANDSCAPE DESIGN	1	6	7
LANDSCAPE FEATURES	3	4	6
LANGUAGES	6	2	3
LEGAL CAUSE	3	1	2
LEGISLATIVE ACTS	1	1	2
LEISURE	2	3	3
LEXICOLOGY	1	1	1
LIGHTING TECHNOLOGY	5	1	6
LIMITS(MATHS)	0	2	1
LINEAR ALGEBRA	8	2	2
LINES(GEOMETRY)	1	0	1
LINGUISTIC SEMIOLOGY	0	2	1
LOW TECHNOLOGY	0	0	2
MAGNETISM(THEORY)	3	1	2
MANAGEMENT STRUCTURES	6	2	7
MANUFACTURED PRODUCTS	10	5	9
MANUFACTURING PROCESS	1	2	8
MARKET ECONOMIES	5	3	6

MARKETING	7	5	7
MECHANICS(THEORY)	8	2	6
MEGALOPOLISES	0	2	4
MERCHANDIZING	10	6	6
METABOLISM	1	0	1
METALLURGICAL INDUSTRY	0	0	1
MATHEMATICS	0	0	1
MECHANICAL ENGINEERING	5	1	8
METEOROLOGICAL CONDITIO9	4	4	6
METROLOGY	8	1	6
METROPOLITAN AREAS	0	2	5
MINING ENGINEERING	4	2	7
MINING INDUSTRY	0	1	3
MINORITIES	0	2	1
NATURAL RESOURCES	4	8	8
NERVOUS SYSTEM	2	0	1
NETWORK ANALYSIS	4	1	3
NUMBERING SYSTEMS	3	1	2
NUMBERS	7	1	3
NUMERICAL ANALYSIS	3	1	2
NUMERICAL CALCULATION	4	2	2
NURSING	2	0	1
OPTICS	7	2	4
OWNERSHIP	8	1	7
PARAFISCALITY	1	1	2
PEDIATRICS	1	0	1
PERSONNEL MANAGEMENT	11	2	8
PETROGRAPHY	3	1	1
PHONETICS	0	0	1
PHOTOCHEMISTRY	4	1	2
PHYSICAL ANTHROPOLOGY	4	4	4
PHYSICAL CHEMISTRY	15	6	8
PHYSICAL PHENOMENA	26	3	7
PHYSICAL TREATMENT	18	4	9
PHYSIOLOGICAL EFFECT	4	3	5
PHYSIOLOGICAL PROCESSES1	0	0	1
PLANE GEOMETRY	0	0	1
PLANNING POLICIES	3	7	8
PLANT ECOLOGY	1	1	2
PLASTIC ARTS	0	0	2
POINT(GEOMETRY)	0	0	1
POLITICAL ACTIVITIES	1	1	1
POLITICAL GEOGRAPHY	2	1	2
POLITICAL INSTITUTIONS	2	2	2
POLUTION	8	4	8
POPULATION	7	2	6
PREVENTIVE MEDICINE	2	1	1
PRIMARY SECTOR	2	2	2
PRIMITIVE HOUSING	0	1	3
PRIVATE LAW	1	6	4
PRIVATE LIFE	1	0	1
PROBABLISTIC APPROACH	0	1	1
PROBLEM SOLVING	4	3	4
PROCEDURES(METHODS)	1	0	3
PRODUCTION ACTIVITIES	14	5	10
PROFESSIONAL PRACTICE	9	4	10

PROGRESSION(MATHS)	3	0	1
PROPERTIES OF MATERIALS	15	3	10
PROPERTY	8	4	9
PROPERTY DEVELOPMENT	0	0	9
PROVISION OF SERVICES	12	2	9
PSYCHIATRY	2	1	1
PSYCHICAL PROCESS	7	1	2
PSYCHOLOGICAL STRESS	0	4	5
PSYCHOPATHOLOGY	0	2	1
PSYCHOPHYSICS	3	1	1
PSYCHOPHYSIOLOGY	1	1	1
PUBLIC ADMINISTRATION	5	8	8
PUBLIC HEALTH	2	4	7
PUBLIC INSTITUTIONS	5	1	6
PUBLIC LAW	1	5	4
PUBLIC RELATONS	0	3	2
PUBLIC SERVICES	3	2	5
PUEBLOS	0	0	1
QUALITY OF LIFE	0	4	4
RECYCLING	2	1	6
REGIONAL GEOGRAPHY	2	2	2
REGIONAL PLANNING	3	7	8
REGRESSION ANALYSIS	0	2	4
RESEARCH AND DEVELOPME	0	1	6
RESOURCE ALLOCATION	0	5	7
RESOURCE INDUSTRIES	2	0	6
RESPIRATORY SYSTEM	1	0	1
REST	1	0	1
RING THEORY	0	1	1
RURAL COMMUNITIES	0	3	4
RURAL ENVIRONMENT	3	5	5
RURAL GEOGRAPHY	0	2	2
RURAL SOCIOLOGY	0	4	3
SAFETY ENGINEERING	6	3	8
SALTS	14	3	5
SALVAGING	0	1	3
SAMPLE THEORY	1	2	2
SECONDARY SECTOR	2	1	2
SEDIMENTOLOGY	0	3	1
SEISMOLOGY	2	0	5
SEMANTICS	0	2	1
SERIES(MATHS)	0	1	1
SERVICE INDUSTRIES	1	3	5
SET THEORY	1	2	1
SEX	0	2	1
SHORTAGES	0	0	1
SIMULATION	0	2	2
SINGLE PERSONS	2	2	4
SOCIAL ANALYSIS	0	6	2
SOCIAL CHANGE	2	5	5
SOCIAL CLASSES	4	3	4
SOCIAL INDICATORS	0	0	1
SOCIAL INSTITUTIONS	2	0	5
SOCIAL PARTICIPATION	2	1	5
SOCIAL PERCEPTION	1	2	3
SOCIAL POLICIES	1	4	5

SOCIAL PSYCHOLOGY	5	5	3
SOCIAL RELATIONS	2	4	4
SOCIOECONOMIC SYSTEMS	7	5	3
SOCIOMETRICS	0	0	1
SOCIOPROFESSIONAL CLAS	16	2	8
SOLID GEOMETRY	0	0	3
SPATIAL ENCLOSURES	1	1	5
SPECIES	0	3	1
SPECULATIVE THOUGHT	0	1	1
STANDARD OF LIVING	3	5	5
STATISTICAL ANALYSIS	7	4	3
STATISTICAL DISTRIBUTION	0	2	1
STRATEGIES	0	5	5
STRATIGRAPHY	0	1	1
STRENGTH OF MATERIALS	6	4	8
STRUCTURAL ENGINEERING	7	3	8
STUDY OF BUILDING DEFEC	2	2	2
SUBURBAN ENVIRONMENT	1	0	5
SUFFERING	0	1	1
SYMBOLIC ANALYSIS	3	0	1
SYMBOLS	0	2	1
SYSTEMS ANALYSIS	6	2	5
SYSTEMS APPROACH	0	2	5
SYSTEMS THEORY	3	3	5
TACTICS	0	2	1
TASK PHYSIOLOGY	6	5	5
TEACHING	7	1	3
TECHNICAL EDUCATION	6	1	4
TECHNICAL CHANGE	1	1	2
TECHNOLOGY TRANSFER	0	4	7
TERRESTRIAL RADIATION	0	0	2
TERRESTRIAL ROTATION	4	1	1
TESTING THEORY	1	3	4
THEORY OF FIELDS	3	4	1
THERAPEUTICS	1	0	1
THERMAL ENGINEERING	5	2	8
THERMAL DYNAMICS	8	3	7
TOPOGRAPHY(SURVEYING)	2	0	6
TOPOLOGY	6	1	3
TOXICOLOGY	0	0	1
TRAFFIC ENGINEERING	4	3	8
TRANSPORTATION ENGINEE	2	3	8
TRIGONOMETRY	1	0	1
URBAN ECONOMICS	3	1	5
URBAN ENVIRONMENT	3	3	5
URBAN GEOGRAPHY	0	3	2
URBAN HISTORY	0	0	1
URBAN PLANNING	5	10	8
URBAN SOCIOLOGY	1	4	3
USE(UTILIZATION)	9	3	8
USERS	7	4	8
UTILITARIAN PRODUCTS	1	2	3
VAUE THEORY	0	2	2
VARIANCE ANALYSIS	0	0	1
VARIATION ANALYSIS	0	2	1
VEGETATION	15	2	6



VILLAGES	0	3	4
VITICULTURE	0	0	1
WASTAGE	0	0	3
WAVE MECHANICS	3	0	1
WEATHER FORECASTING	1	4	5
WORK ENVIRONMENT	1	1	5
WORK ORGANIZATION	14	5	9
WORK SOCIOLOGY	0	0	2
WORKS OF ART	6	2	5

**Statistics :**

Rank	number
10	17
9	15
8	36

