Theories of Information and Uncertainty for the modelling of Information Retrieval: an application of Situation Theory and Dempster-Shafer's Theory of Evidence

Mounia Lalmas

Department of Computing Science
University of Glasgow

April 19, 1996

Thesis submitted for the Degree of Doctor of Philosophy at the University of Glasgow

Declaration of Originality

The material in this thesis is entirely the result of my own independent research under the supervision of Professor C.J. van Rijsbergen, and is not the outcome of any collaborative work. All published or unpublished material used in this thesis has been given full acknowledgment.

I further state that no part of my dissertation has already been, or is currently being, submitted for any such degree, diploma or other qualification.

Permission to Copy

Acknowledgments

Although this thesis represents my individual work, many people contributed to it indirectly through their discussion and support.

First, my thanks go to my supervisor Keith van Rijsbergen. Keith, it was great working with you. I enjoyed our discussions and arguments. Thanks for allowing me much freedom in my work, while at the same time making sure I did not lose perspective. Thanks for believing in me when I was having serious doubt about my work, and for your advice both professional, and about life, films, and wines.

I would also like to thank Theo Huibers who read many versions of this thesis. Theo, we do share the same passion in our work, and I hope we will do so for a long time.

The people in the IR group at the Department of Computing Science, University of Glasgow, have been very supportive. They have shared both my anxiety and happiness. I would particularly like to thank Mark Sanderson and Iain Campbell for their help during my experimental work (I now know how to build a "slow" IR system, and I can also evaluate it!).

I am very much in debt to Chris Ledgebow and Kevin O'Neil for reading earlier version of this thesis, and for teaching me how to write English that people can understand.

I would like to thank the staff and students of the School of Computer Science at the University of Windsor, Ontario, Canada for their support while writing the first part of this thesis and, in particular, Steve Karamatos, for explaining the mystery of Publisher.

I am indebted to the staff and students of the Department of Computing Science, University of Glasgow for their support throughout this work. In particular, Anne Sinclair who has always been there to answer my queries, and the students of Room F101; Aran Lunzer, Brian Matthews, Daniel Chan, Jackie Moyes and Sharon Flynn, in whose warm company I spent the first three years of this work.

I would like to thank all the people working in logic and information retrieval, and the FERMI group. It was fun meeting you, and talking about our work. I hope that this will continue.

And finally, I would like to thank my husband Steve McGowan for proof-reading this thesis many times, and his constant and great help and companionship. Steve, I owe you lots of bottles of malt whisky.

Funding for this research was provided by a Glasgow University Postgraduate Scholarship (Science) and an Overseas Research Students (ORS) Scholarship ORS/9017022.

Abstract

Current information retrieval models only offer simplistic and specific representations of information. Therefore, there is a need for the development of a new formalism able to model information retrieval systems in a more generic manner. In 1986, Van Rijsbergen suggested that such formalisms can be both appropriately and powerfully defined within a logic. The resulting formalism should capture information as it appears in an information retrieval system, and also in any of its inherent forms. The aim of this thesis is to understand the nature of information in information retrieval, and to propose a logic-based model of an information retrieval system that reflects this nature.

The first objective of this thesis is to identify essential features of information in an information retrieval system. These are:

- flow,
- intensionality,
- partiality,
- structure,
- significance, and
- uncertainty.

It is shown that the first four features are qualitative, whereas the last two are quantitative, and that their modelling requires different frameworks: a theory of information, and a theory of uncertainty, respectively.

The second objective of this thesis is to determine the appropriate framework for each type of feature, and to develop a method to combine them in a consistent fashion. The combination is based on the Transformation Principle.

Many specific attempts have been made to derive an adequate definition of information. The one adopted in this thesis is based on that of Dretske, Barwise, and Devlin who claimed that there is a primitive notion of information in terms of which a logic can be defined, and subsequently developed a theory of information, namely Situation Theory. Their approach was in accordance with Van Rijsbergen's suggestion of a logic-based formalism for modelling an information retrieval system. This thesis shows that Situation Theory is best at representing all the qualitative features.

Regarding the modelling of the quantitative features of information, this thesis shows that the framework that models them best is the Dempster-Shafer Theory of Evidence, together with the notion of refinement, later introduced by Shafer.

The third objective of this thesis is to develop a model of an information retrieval system based on Situation Theory and the Dempster-Shafer Theory of Evidence. This is done in two steps. First, the unstructured model is defined in which the structure and the significance of information are not accounted for. Second, the unstructured model is extended into the structured model, which

incorporates the structure and the significance of information. This strategy is adopted because it enables the careful representation of the flow of information to be performed first.

The final objective of the thesis is to implement the model and to perform empirical evaluation to assess its validity. The unstructured and the structured models are implemented based on an existing on-line thesaurus, known as WordNet. The experiments performed to evaluate the two models use the National Physical Laboratory standard test collection.

The experimental performance obtained was poor, because it was difficult to extract the flow of information from the document set. This was mainly due to the data used in the experimentation which was inappropriate for the test collection. However, this thesis shows that if more appropriate data, for example, indexing tools and thesauri, were available, better performances would be obtained.

The conclusion of this work was that Situation Theory, combined with the Dempster-Shafer Theory of Evidence, allows the appropriate and powerful representation of several essential features of information in an information retrieval system. Although its implementation presents some difficulties, the model is the first of its kind to capture, in a general manner, these features within a uniform framework. As a result, it can be easily generalized to many types of information retrieval systems (e.g., interactive, multimedia systems), or many aspects of the retrieval process (e.g., user modelling).

Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 What is an Information Retrieval System?

An *information retrieval* (IR) *system* [vR79, Bla90, Doy75] is a tool used to store information for the later retrieval and use of interested parties. Information can be stored and relayed in different forms including texts, images, audio and video tapes. This thesis deals principally with textual information (e.g., articles, books, news, diagnoses, etc.) stored under the form of *documents*, the set of which constitutes a *corpus* or *collection*. However, the work carried out in this thesis is pertinent to all forms of information. In addition, this work is concerned with computer-based automatic IR systems. Manual systems are excluded because they are inadequate in dealing with large amounts of information. They have become too prohibitive and time consuming. In this thesis, when referral is made to an IR system, the system will consist of a collection of documents, the textual *content* of which contains *information* a user may consult to satisfy an information need.

In most cases, an IR system does not, or cannot, incorporate the entire *information content* of a document due to factors of length (e.g., books) and complexity. Furthermore, a limited storage capacity is often the case and a fast access is essential. Hence, an IR system handles a manipulable representation of the document information content. This *internal representation* aims to model as faithfully as possible the document's information content. The creation of the internal representation of a document from its textual information content is a prominent function of an IR system. The output of the internal representation can influence considerably the effectiveness of the IR system. In conventional IR systems, the creation of internal representation is often referred to as *indexing*. The outcome of indexing is a set of *indexing items* that supposedly summarize the information content of a document. The indexing items can be keywords, phrases, parse trees, semantic structures, and, in extreme cases, full texts.

A *user* in need of information submits a *query* to the IR system that expresses the information need. The query is then *evaluated* by the system and transformed into an *internal representation* that is manageable by the system. The transformation of a query involves a process often similar to that used to represent a document's information content. The system *compares* the query representation with *all* the document representations and determines by some *matched-based computational techniques* the document representations which may satisfy the user request. These then become the *retrieved documents*. The comparison process depends primarily on the type of IR system being used (more about this in the next section).

Depending on how efficient and adequate the system is, the retrieved documents correspond variably

to the *relevant* documents that satisfy the original information need. The positive correspondence between retrieved and relevant documents is the main objective of an IR system: a good IR system should retrieve as *many* relevant documents as possible, but *only* the relevant documents.

In some IR systems, the comparison of a document and a query representation results in a numerical value that expresses to what extent the information content of that document satisfies the information need as specified in the query. The resultant numerical value is often referred to as a *degree of relevance*. Consequently, in those systems, the retrieved documents are ordered according to a degree of relevance. The ordering displays to the user which documents, according to the system, satisfy his or her query, the best.

In this thesis, the term "relevance" is used in both contexts, with respect to the user (which is the correct use in the IR world) and with respect to the system (indicating good satisfaction).

Both the comparison and the representation processes of documents and queries sometimes use additional semantic *knowledge* generally stored in a *thesaurus*. An example is that of synonymous relationships; a document indexed by an item is also indexed, eventually implicitly, by all the synonyms of that item that are stored in the thesaurus.

In some IR systems, upon delivery of the documents, the user can *specify* which, among the retrieved documents, are particularly *relevant*. This information can be taken into account by the IR system in a manner that depends on the type of IR system, which then, sometimes together with the user, constructs a second query which is submitted to the system. This process is called *relevance feedback*. This thesis is not concerned with this feature.

When building an IR system, an evaluation method is required to test the system performance. The most commonly adopted is the *precision — recall* method:

$$Precision = \frac{number\ of\ retrieved\ and\ relevant\ documents}{number\ of\ retrieved\ documents}$$

$$Recall = \frac{number\ of\ retrieved\ and\ relevant\ documents}{number\ of\ relevant\ documents}$$

A general schema for the overall functionality of an IR system is in Figure 1.1. The manner in which documents and queries are represented and the comparison process utilized depends on the model of the IR system.

1.2 Models for Information Retrieval

There are various models of IR systems. The most publicized are the *Boolean*, *Vector Space* [Sal71, SM80], *Probabilistic* [vR79, Rob77, CGD92, Fuh92, vR92], and more recently, the *Logical* [vR86a, Nie90, vRL96, CC92] models (for a survey, see [Lal96b]). The logical models were advanced because it has been observed (the details can be found in [Nie90, LvR93, vRL96]) that the other models appear to have reached their maximum potential. Although many extensions of the Boolean Model, Vector Space Model or Probabilistic Model are claimed to be more advanced, the extended models tend to differ from previous models because

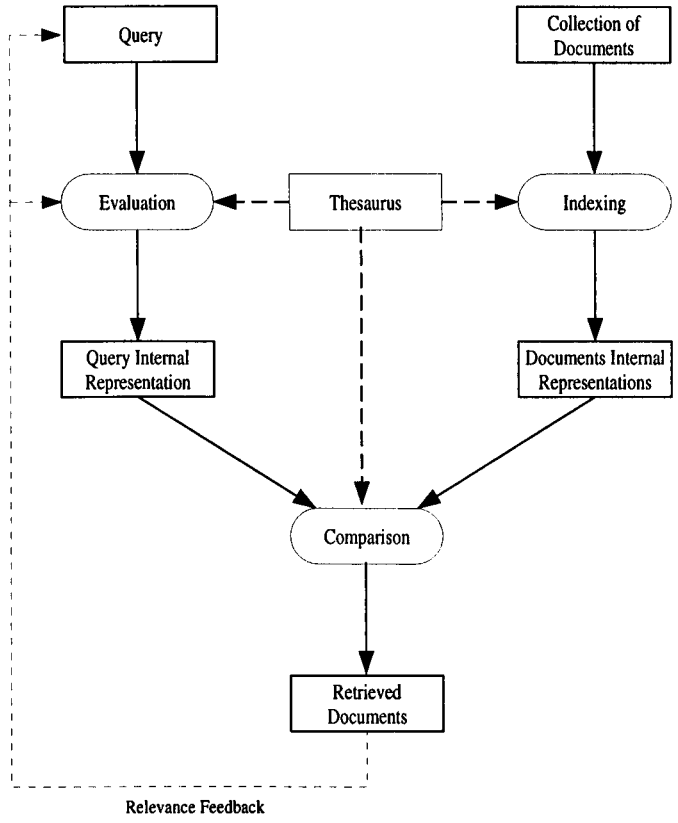


Figure 1.1: The different components of an IR system

(i) of their implementation of new techniques or algorithms based on more advanced technologies, or

(ii) the setting of parameters which provide different variants of the same model, some variants being more effective than others.

Regardless, very significant improvements have not really been observed.

Most IR systems to date have proposed a very simplistic representation of textual information. For example, in the Vector Space model [SM80, Sal71] the information contained in a document is represented by independent index terms, called *stems* [Por80]. An initial selection of words that appear in the document removes common words (like "is", "the", "every", etc.). The remaining words are stemmed. This stemming ensures that words such as "connections", "connection" and "connected" are represented by the unique stem "connect". A weight is assigned to each stem, which very often corresponds to its occurrence frequency within the document. A document is then represented by a vector, in which the components are the weights associated with the stems. In most cases, the original semantic relationships between words are ignored. It is obvious that such a representation does not capture the complexity of textual information very well.

In 1986, Van Rijsbergen [vR86a] suggested a model of an IR system based on logic because the use of an adequate logic can provide all the necessary tools to model the different functions of an IR system, and in addition seem to be a more accurate model of information. Indeed, most logics consist of sentences which can be joined by connectors to construct complex sentences. A particular connector is the implication $\rightarrow$ which is used to model inference. Given two sentences, ϕ

and ψ, the truth of $\phi \rightarrow \psi$ (more about this notion of truth later) means that the sentence ϕ implies the sentence ψ whenever ϕ is true. In other words, ψ can be inferred from ϕ. Suppose there is a way to represent the information content of a document by a sentence d and the information need as phrased in the query by a sentence q. The truth of $d \rightarrow q$ would mean that the query sentence can be inferred from the document sentence. To put it another way, the information captured by d is sufficient to infer the information represented by q. In the IR world, this could be viewed as the document *satisfying* (or to be relevant to) the query.

1.3 What is Logic?

The nature of information in IR is complex. For example, if the information concerned is textual then natural language is involved; the study of such is delicate, complicated and enigmatic. The Science American Journal states:

> "The grammar of languages includes rules of phonology, which describe how to put sounds together to form words; rules of syntax, which describe how to put words together to form sentences; rules of semantics, which describe how to interpret the meaning of words and sentences; and rules of pragmatics, which describe how to participate in a conversation, how to sequence and how to anticipate the information needed by an interlocutor."

However difficult the essence of information is to seize, a model of an IR system should be principally concerned with the incorporation of information, and this is possible with logic. Indeed, in the Oxford English Dictionary, *logic* is defined as:

> "The branch of philosophy that treats of the form of thinking in general, and more especially of inference and scientific method."

That is, logic is a formalization of the way we use information in our everyday life to think, infer, conclude, acquire knowledge, make decisions and so forth. In this sense, logic undertakes to model *information and its flow.*

The use of logic for modelling IR is not a particularly new idea. It was first suggested by Van Rijsbergen [vR86a] and later followed by authors such as Nie [Nie90, Nie88, Nie89], Bruza [BvdW91, BvdW92, Bru93] and Sebastiani & al [MSST93, Seb94] all of whom proposed interesting frameworks. My main objection is that these authors have all adopted a truth-based logic, which corresponds to the second view of logic in the Oxford English Dictionary:

> "Also since the work of Frege (1848–1925), [logic is] a formal system using symbolic techniques and mathematical methods to establish truth-values in the physical sciences, in language, and in philosophical argument."

In many domains that relate to information, such as artificial intelligence, databases, linguistics and even philosophy, information is represented by some structure or calculus that is built on the concept of *truth.* I object to this representation following the line of Drestke [Dre81], Landman [Lan86], Barwise & al [Bar89, Bar91, Bar92, BE87, BE90, BP83] and Devlin [Dev91], and so forth, and advances a logic-based model of IR using a *logic of information.* This thesis adopts the first above definition of logic and puts forward a logic-based model of IR systems in which the

appropriate representation of information is a crucial factor.

In order for the reader to understand a logic of information, and to see the difference between this version and the truth-based logics, he or she must understand the components of truth-based logics. For that purpose, one framework that belongs to this category, namely *Classical Logic* [Ram88, Tur84, Gal87], is described next.

1.4 What is Classical Logic?

A description of Classical Logic is necessary for two reasons. First, the desired attributes for a logic to model an IR system can be identified. Second, Chapter 2 and Chapter 3, which contain surveys of possible frameworks for modelling an IR system, use many concepts from Classical Logic, which are important to define correctly. Although Classical Logic is mentioned, only a subclass of Classical Logic, namely *Propositional Calculus*, is described. Variables, quantifications, and assignment functions of the *Predicate Calculus* are irrelevant in the arguments made in this chapter.

Let L be a logic. A *vocabulary* is defined, composed of a set of *propositions* $\{p, q, r, s, \ldots\}$, as well as *logical connectors* $\wedge, \vee, \neg, \rightarrow$ and $\leftrightarrow$. The logic L defines a formal language by *syntax* and *semantics*.

1.4.1 Syntax

The syntax of Classical Logic specifies formally the set of well-formed formulae (wff) or *sentences* as follows:

(i) if p is a proposition, then $p \in L$,
(ii) if $\phi \in L$ and $\psi \in L$, then $\phi \wedge \psi \in L$, $\phi \vee \psi \in L$, $\neg\phi \in L$, $\phi \rightarrow \psi \in L$ and $\phi \leftrightarrow \psi \in L$.

Examples of wffs are $p \vee q$ and $\neg p \rightarrow q \wedge r$.

1.4.2 Semantics

Any non-logical symbol in L has an intended meaning called a *semantic value*. The set of these constitutes the semantics of L. In Propositional Calculus, semantic values are the set $\{0, 1\}$ of truth values *false* and *true*, respectively. The semantics of a well-formed formula (formula from now on) are defined by the semantics of the formulae that constitute it and the semantics attached to the different logical connectors. This is known as the *Principle of Compositionality*.

In Classical Logic, the semantic value attached to *conjunction* $\wedge$, *disjunction* $\vee$, *negation* $\neg$, *implication* $\rightarrow$ and *equivalence* $\leftrightarrow$ are described in the following *truth table*:

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Table 1.1: Semantics of negation, conjunction, disjunction, implication and equivalence

The semantic value of a sentence $\phi \in L$ is denoted $||\phi||$, whose value depends on the propositions and the connectors that appear in ϕ. For example, if $\phi = \neg p \vee \neg q$ and $||p|| = ||q|| = 1$ then $||\phi|| = 0$.

If the set of non-logical symbols in L is $\{p, q\}$, four *interpretations* are obtained, one for each line of the above truth table. For example, in the second interpretation, $||p \wedge q|| = 0$ and $||p \vee q|| = 1$. More formally, an interpretation is a structure $I = \langle \{0,1\}, F \rangle$ where F is a function that assigns semantic values to the propositions. It then becomes necessary to say that a formula ϕ is *true with respect* to a particular interpretation I, not just that ϕ is true. This is denoted as $||\phi||^I$. Moreover, $||\phi||^I = 1$ is written $I \models \phi$ and $||\phi||^I = 0$ is written $I \not\models \phi$. The relation $\models$ is read 'satisfies'. Semantics are therefore re-expressed as follows (p is a proposition of L, and φ and ψ are formulae of L):

(i) $I \models p$ if and only if (iff) $F(p) = 1$
(ii) $I \models \phi \wedge \psi$ iff $I \models \phi$ and $I \models \psi$
(iii) $I \models \phi \vee \psi$ iff $I \models \phi$ or $I \models \psi$
(iv) $I \models \neg\phi$ iff $I \not\models \phi$
(v) $I \models \phi \rightarrow \psi$ iff $I \not\models \phi$ or $I \models \psi$
(vi) $I \models \phi \leftrightarrow \psi$ iff either $I \not\models \phi$ and $I \not\models \psi$, or $I \models \phi$ and $I \models \psi$

The fact that the formula ϕ is true in any interpretation is denoted $\models \phi$; the formula ϕ is said to be *valid.* It is also called a *tautology* or is said to be *logically true.*

Often, only a few interpretations of the above four are of interest. Suppose that one wants to represent only the cases in which p and q are true; one is then only interested in those interpretations that make these two propositions true. These interpretations are called *models* with respect to p and q. So a model for a formula ϕ, or a set of formulae Φ, is any interpretation that satisfies ϕ or Φ. The relation $\models$, when used as follows $\phi_1, \ldots, \phi_n \models \psi$, expresses that any model of $\phi_1, \ldots, \phi_n$ is also a model of ψ. In such a case, it is said that ψ is a *logical consequence* of $\phi_1, \ldots, \phi_n$.

1.4.3 Axiomatic System

There is another way to characterize validity for formal languages. Syntactic rules can be defined rather than trying to establish whether $\phi_1, \ldots, \phi_n \models \psi$ by enumerating all interpretations. These are *axioms* which are formulae that are assumed true, and *inference rules.* Indeed, Classical Logic has been syntactically defined with several axioms and one inference rule called the *Modus Ponens.* The Modus Ponens states that if both ϕ and $\phi \rightarrow \psi$ are true, then ψ can be inferred or is true.

A *proof* is defined as any sequence of formulae of L such that each formula is either an axiom or follows from one or more of the preceding sentences of the sequences by the application of the

Modus Ponens. A *theorem* of the language is any sentence ϕ for which there is a proof ending in ϕ.

A derivability relationship $\vdash$ is defined between a set of formulae and a formula $\phi_1, \ldots, \phi_n \vdash \psi$ iff there exists a finite sequence of the inference rule that leads $\phi_1, \ldots, \phi_n$ to ψ. The fact that a formula ϕ is a theorem is written $\vdash \phi$. A set of axioms together with all the theorems that can be derived from it is called a *theory*. To finish, the *Deduction Theorem* says that $\phi \vdash \psi$ is equivalent to $\vdash \phi \rightarrow \psi$.

1.4.4 Soundness and Completeness

Soundness means that only true statements can be proven. That is, if $\phi \vdash \psi$ then $\phi \models \psi$. *Completeness* means that all true statements can be proven. That is, if $\phi \models \psi$ then $\phi \vdash \psi$. Classical Logic, or more correctly, Propositional Calculus, is both sound and complete. As a result, there are two ways to prove the truth of a formula, one using $\models$, and the other $\vdash$. The first method is referred to as a *model system* approach and the second as an *axiomatic system* approach. The different notations are summarized in the table below:

Model System		Axiomatic System	
Valid sentence	$\models \phi$	Theorem	$\vdash \phi$
Logical consequence	$\Gamma \models \phi$	Deduction	$\Gamma \vdash \phi$

Table 1.2: Model System and Axiomatic System

In some of the logical frameworks described in Chapter 2, the correspondence between the axiomatic system and the model system does not exist. These frameworks are expressed in a model-type system unless otherwise stated. For the remainder of this chapter, the arguments are made in the model-theoretical system approach, although Classical Logic would allow the use of both.

Next, Classical Logic is used to express the relevance of a document to a query. This allows the reader to become more familiar with the notations used in this thesis, and to understand the requirements of a logic for IR. The precise list of requirements is given in section 1.6.

1.5 Modelling Information Retrieval with Classical Logic: does it work?

Given a logic, let d and q be the sentences, in that logic, representing the information content of the document and the information need phrased in the query, respectively. The relevance of the document to the query can be expressed by the implication $d \rightarrow q$. That is, determining the relevance consists of deciding whether $d \rightarrow q$ is valid, meaning that the implication holds for all interpretations of the logic. As explained in the previous section, evaluating the validity of $d \rightarrow q$ in Propositional Logic is equivalent to asserting $d \models q$, $\models d \rightarrow q$, $d \vdash q$ or $\vdash d \rightarrow q$. Here, $d \models q$ is used, which consists of establishing whether any model of d is a model of q.

A working example is used. Suppose that the vocabulary consists of the propositions $\{t_1, t_2, t_3\}$.

Let the document be $d = t_1 \wedge t_2$. There are two models of d:

t_1	t_2	t_3	d
1	1	0	1
1	1	1	1

Table 1.3: The models of the document d in the Classical Logic

Five queries are defined:

(i) $q_1 = t_1$,
(ii) $q_2 = t_3$,
(iii) $q_3 = t_1 \wedge t_3$,
(iv) $q_4 = t_1 \vee t_3$, and
(v) $q_5 = t_1 \wedge t_2$.

Their evaluations, with respect to the models of d, are given in the table below:

t_1	t_2	t_3	q_1	q_2	q_3	q_4	q_5
1	1	0	1	0	0	1	1
1	1	1	1	1	1	1	1

Table 1.4: Evaluation of different queries in Classical Logic

From the fourth column, it can be seen that $d \models q_1$; the document is relevant to the query. From the fifth column, it can be seen that $d \not\models q_2$; the document is not relevant to the query. From the sixth column, $d \not\models q_3$. However, one would have considered the document represented by the formula d to be *more relevant* to q_3 than it is to q_2 because the document, though not *exhaustively* relevant, is nonetheless *partially* relevant to the query q_3. The problem is that $\models$ is too *rigid* a relation and cannot express partial relevance. From the seventh and the eighth columns, $d \models q_4$ and $d \models q_5$. One would have expected the document represented by d to be more relevant to q_5 than to q_4. This counter-intuitive result is due to the semantics attached to disjunction. If ϕ is valid, then any sentence of the form $\phi \vee \psi$ is also valid even if ϕ and ψ are the representations of information items that are not related. To finish, with the queries q_1 and q_5 (fourth and eight columns), the outcomes are $d \models q_1$ and $d \models q_5$. One would have considered the document to be more relevant to q_5 than to q_1, for all the information items in d concern q_5, whereas fewer are related to q_1. That is to say, the document is more *specific* to q_5 than it is to q_1. Classical Logic cannot express *specificity*.

This last problem was also observed by Nie [Nie90] who then proposed a formulation of the specificity by evaluating the inverse implications, respectively $q_1 \rightarrow d$ and $q_5 \rightarrow d$. This is shown in Table 1.5. The outcome is $q_1 \not\models d$ and $q_5 \models d$. This formulation can reveal the specificity of the document. My objection to this formulation is that, in most cases, a document is composed of many conjuncts whereas a query contains very few conjuncts. Very few implications are valid, thus specificity cannot be expressed. Besides, a document is a provider of information (more about this in the next section), implying that the evaluation of the inverse implication is counter-intuitive.

t_1	t_2	$q_1 \rightarrow d$	$q_5 \rightarrow d$
0	0	1	1
0	1	1	1
1	0	0	1
1	1	1	1

Table 1.5: Example of the representation of the specificity of a document in Classical Logic

This simple example already demonstrates the weakness of Classical Logic as the basis for a model of an IR system. Next, the issues raised here are summarized, and many others are highlighted.

1.6 The problems

Different problems arise with the use of Classical Logic for IR mainly because truth is considered as the fundamental notion. These problems are presented in turn, although they often overlap. At the same time, the fundamental features of a logic for IR are identified.

1.6.1 Truth

There are two views of semantics: the formal interpretation of a logic and the portrayal of the meaning of natural language. An IR system is concerned with the second view, whereas Classical Logic concentrates on the first view. This is because, in Classical Logic, the definition of semantics is truth-based instead of information-based. For example, we usually affirm the disjunction of two sentences only if we believe that one member of the disjunction is true, but we do not know which one. Nonetheless, the sentence "the lawn is green or blue" is valid in Classical Logic, which is nonsense since we all know that "the lawn is never blue" (in normal circumstances). The validity is due to the semantics of disjunction (if ϕ is true, so is $\phi \vee \varphi$). The assertion of a disjunction should be taken as an admission that we do not know which member of the disjunction is true. That is, it should convey *imprecision* [Mor90, KC93] (see also [Lan86] for an extended discussion of disjunctive information).

1.6.2 Significance

In Classical Logic, due to the truth-based interpretation of the conjunction, $\phi \wedge \phi \leftrightarrow \phi$ is a tautology. With respect to IR, $\phi \wedge \phi$ can mean the information represented by the formula ϕ is significant since it appears more than once. Indeed, the fact that an item of information appears many times may indicate that the item is a significant part of the document information content. A weighting mechanism which captures the significance of information is often necessary. Classical Logic does not provide such a mechanism.

1.6.3 Implication

In Classical Logic, $\phi \rightarrow \psi$ is equivalent to $\neg\phi \vee \psi$, implying that $\phi \rightarrow \psi$ is true whenever $\neg\phi$ is true. For example, "$2 + 1 = 5 \rightarrow$ *Mounia likes to swim*" is a valid sentence because

"$2 + 1 = 5$" is always false. This rule is rather inadequate in representing everyday reasoning. For most people, asserting an implication means that antecedent of the implication is true while its consequent is false does not occur. Furthermore, if both ϕ and ψ are valid then $\phi \rightarrow \psi$ is valid as well. Yet, one might hesitate to say that $\phi \rightarrow \psi$ is valid since one would expect some information-based connections between ϕ and ψ before one could determine the actual validity of $\phi \rightarrow \psi$. For example, the sentence "$2 + 2 = 4 \rightarrow$ *Apple is a fruit*" is valid in Classical Logic because both the antecedent and the consequent are valid; however, there is no connection between "$2 + 2 = 4$" and "*Apple is a fruit*".

These two examples show that implications as defined in Classical Logic do not necessarily capture information containment. In ordinary language, one tends to join two sentences with an implication only if there is some connection between them in their form and content. Therefore, the implication $d \rightarrow q$ as defined in Classical Logic is not best at modelling the relevance of a document to a query, since it does not necessarily mean that the document contains information pertinent to the query.

1.6.4 Informative relationship

The evaluation of $d \rightarrow q$ should take into account the meaning of information. For example, a document about "Italy" could be relevant to a query about "Mediterranean country" because Italy is a Mediterranean country. The latter constitutes an informative relationship, and can be modelled by a formula of Classical Logic such as "*Italy* $\rightarrow$ *Mediterranean country*".

Let Γ be the set of formulae representing informative relationships such as that above. One way to incorporate these informative relationships in the evaluation of $d \rightarrow q$ is to evaluate the implication only in those models of d that are also models of Γ, which means evaluating $\Gamma \models d \rightarrow q$. However, the formulae in Γ can be contradictory (information is often contradictory), so it may be impossible to obtain a model of Γ. Also, this evaluation does not eliminate the problem encountered with non-informative implications such as "$2 + 1 = 5 \rightarrow$ *Mounia likes to swim*" or "$2 + 2 = 4 \rightarrow$ *Apple is a fruit*", and the other problems encountered with the use of Classical Logic remain.

1.6.5 Provider of information

Basing relevance on the validity of $d \rightarrow q$ is counter-intuitive because speaking of a true document formula, or a model of the document formula, is meaningless, for a document is the provider of information. A more appropriate use of Classical Logic would be to represent a document by a model in which the formulae representing the information contained in that document are true. This approach endorses the more correct view that relevance is determined on the basis that the document contains information pertinent to the query. In reality, a set of models may be involved since the truth values of some propositions may be unspecified. This matter is further discussed in section 1.6.7.

1.6.6 Intensionality

In Classical Logic, synonymy is symbolized by tautologies. For example, the fact that two terms are synonymous is symbolized by the validity of $\phi \leftrightarrow \psi$, where ϕ and ψ are the formulae representing

the two terms. It follows that every instance of ϕ in a formula can be replaced by ψ. Such a substitution is not always correct because the meaning attached to ϕ can be context-dependant (e.g., ϕ represents a polysemic term). The phenomenon where the meaning of information is context-dependent is referred to as *intensionality* [PtMW90, DWP81, Mon74], and the concerned information is qualified as *intensional*. Classical Logic cannot handle intensionality in any adequate manner.

1.6.7 Partiality

Many items of information are not originally identified as part of a document's information content, though they are implicit in the document information content. The representation of a document is only *partial*; it can grow when the implicit information becomes available due to the flow of information (this is explained in the next section). This characteristic is referred to as the *partiality* of information [Lan86, Bar89][1].

The representation of partiality needs to express that the truth value of a formula is not always known at some point, but can become known at some later stage. In Classical Logic, the representation of the unknown truth of a proposition p necessitates at least two models, one in which p is true and one in which p is false. If models symbolize a document, a set of models may be involved in modeling the document. Nonetheless, the notion of growth of information is foreign to Classical Logic, for models are distinct and non-related entities. Classical Logic cannot capture partiality.

1.6.8 Flow of information

A text document consists of sentences expressed in natural language, and which possess an information content. Part of the information content corresponds to the meaning of these sentences, whereas another part goes beyond this meaning. This is because the content of a document conveys information in two forms: *explicitly*, one can read it; or *implicitly*, one can deduce or infer it. For example, a document about "cross country skiing" may be relevant to a query about "Scandinavian sports", even if the latter is not explicit in the document. The reason is that the information item "Scandinavian sports" is often implicitly contained in any references of "cross country skiing". The phenomenon of *information containment* constitutes the *flow of information* [Dre81, BP83, Bar89, BE90].

The flow of information is a leading component in the modelling of an IR system. There are different types of flows with respect to textual IR systems: there is the flow that allows us to read, that is, the recognition of letters, words, and sentences; there is the flow that allows us to understand what we are reading, that is, the semantics; and there is the flow that allows us, with respect to our knowledge of the subject, to derive additional information from what we have read, that is, the pragmatics. The flow of information, whether related to semantics or pragmatics, is based on information-based relationships between information items. Examples of which are "whisky" to "Scotland", and "Chocolate" to "Belgium". Classical Logic cannot model a flow properly because related information items are symbolized by formulae, which unfortunately are truth-based instead

[1] The terminology is not to be confused with partial relevance, mentioned in section 1.5.

of information-based. Therefore, many relationships are erroneous.

Explicit and implicit information, and the flow of information, are greatly accounted for in this thesis. A complete definition will be given in due course. What should be remembered is that the explicit information, together with the flow of information, derive the implicit information.

1.6.9 Uncertainty

An exact information content cannot be identified appropriately. Indeed, the representation of the meaning of natural language is not an easy task because natural language is ambiguous. For example, intensionality is not always well captured, and as a result, the flow of information that arises from this information is *uncertain* [KC93, DP85, Saf87]. Since the relevance of a document to a query often depends on the existence of a flow that leads the explicit information content of the document to the information being requested by the query, the more uncertain is a flow, the less relevant the document. One approach to express this correspondence is to have a numerical evaluation of relevance that is based on a numerical expression of the uncertainty of the overall flow. Therefore, the logic used to model the IR system must be non-binary. This is not a characteristic of Classical Logic.

1.6.10 Structure

A document has an underlying structure. For example, a document may have a title, several authors, an abstract, the text itself, and some figures. A multimedia document may consist of a mixture of text, image, and video. The structure of a document can also be implicit. For example, a structure may consist of the information (e.g., terms) contained in the document, which defines a document topic. Such types of structures are based on semantics because they take into account the fact that information can be semantically related. For reasons of simplicity, only these types of structures are considered in this thesis.

An example of semantically related information is equivalent items of information. A document should not be more relevant to a query that uses many terms to express an information need than to a query using fewer terms to express the same information need. This equivalence of information can be taken into account by grouping equivalent terms into structures and treating the groups of equivalent terms as entities. The representation of such structures cannot be handled by Classical Logic.

1.6.11 Summary

The inadequacy of the use of Classical Logic for modelling an IR system has been shown. Simultaneously, some points were made about the components and their characteristics that are necessary for modelling an IR system. These components are summarized in the following list:

(i) the representation of information on another basis than truth.

(ii) the representation of a document information content that accounts for intensionality, partiality, explicit and implicit information, significance and structure of information. The document should also be represented as a provider of information.

(iii) the representation of the flow of information.

(iv) the representation of the uncertainty engendered by the flow of information. A quantitative representation of the uncertainty can be used as a basis for a numerical expression of relevance.

This thesis proposes a logic-based IR model that captures the above components. The model is based on the so-called *Transformation Principle*.

1.7 The Transformation Principle

I believe that an IR model should be expressed in an information-based framework. More precisely, a *logic of information* or *theory of information* should be used to build the IR model. In further references, the terminology *theory of information* is used, since this terminology covers more ground than the usual logic framework.

The choice of the appropriate theory is one purpose of this thesis. A *non-binary logic* is required because basing the computation of relevance on the validity of $d \rightarrow q$ leads either to too few or too many documents being retrieved. A non-binary logic would permit the documents that are only partially relevant to the query to be retrieved as well as those directly relevant, and the uncertain nature of the flow of information can be captured.

Two directions are possible: the first is to make the evaluation of $d \rightarrow q$ numerical; the second is to keep the evaluation somewhat binary[2] and to use concurrently a *theory of uncertainty* [KC93, Saf87, Par94] to embody partial relevance. Such an approach is not uncommon as Saffioti [Saf87] mentioned:

> "Many of these solutions [of representing uncertainty] share the attitude of viewing the knowledge and the uncertainty about it as two different entities, and so treating them by means of two distinct loosely-coupled processes: the reasoning process handles knowledge as if it were exact, while a "parallel uncertainty inference" process accompanies it, computing the uncertainty affecting each newly arrived fact. This uncertainty is in turn usually based on the uncertainty affecting the facts used to derive the new fact."

This second approach is adopted because the components of a logic-based IR model identified in section 1.6.11 can be classified as *qualitative* or *quantitative*. The qualitative components are the representation of information and its flow, partiality, explicit and implicit information, structure, and intensionality, whereas the quantitative components are the significance of information, and the uncertainty inherent to information and its flow.

The modelling of the qualitative and quantitative components requires different frameworks. Therefore, the objective is to determine the appropriate framework for each, and to develop a method to combine them in a consistent fashion. A first step towards this is Van Rijsbergen's *Logical Uncertainty Principle* [vR86b]:

> "Given any two sentences x and y; a measure of the uncertainty of $y \rightarrow x$ relative to a

[2] Binary here should be understood as qualitative as opposed to quantitative.

given data set, is determined by the minimal extent to which we have to add information to the data set, to establish the truth of $y \rightarrow x$."

A modified version of this principle is used, which I call the *Transformation Principle*:

"Given a document representation d, a query representation q, and a knowledge set K; the measure of relevance, denoted $d \rightarrow q$, relative to K, is determined by the minimal transformation applied to d to obtain some d' such that d' contains q, denoted $d' \Rightarrow q$."

The Transformation Principle enables a formal expression of the different components that constitute a model of an IR system based on the flow of information. The symbol d represents the document. This representation should view the document as a provider of information and should cater to the modelling of intensionality, partiality and structures. The symbol q represents the query. The symbol K represents the knowledge set which contains the informative relationships upon which the transformation is based. The transformation of a document d to some document d' is due to the flow of information that arises from the document's explicit information content symbolized by d. The result of the flow of information, that is, the document's implicit information content, is symbolized by d'. The notation $d \Rightarrow q$ means that the information represented by q is explicit in the document representation. For example, if d is a set of terms, $d \Rightarrow q$ could mean that the term represented by q belongs to this set. The notation $d \rightarrow q$ signifies that the information represented by q is implicit in the document. In the above example, $d \rightarrow q$ could mean that the term represented by q is synonymous with a term that belongs to the set of terms d. The evaluation of $d \rightarrow q$ depends on the minimal transformation that leads d to some d' such that $d' \Rightarrow q$. d is referred to as the original document whereas d' is referred to as the transformed document. All these components d, q, K, d', $d \Rightarrow q$ and $d \rightarrow q$ are extensively discussed in Chapter 2.

Minimality ensures that the transformation process ceases as soon as the information being sought is reached. This portrays the obvious fact that a document that requires less transformations than another is usually more relevant to the query than the other document.

A correct application of the Transformation Principle is essential. That is, the transformation of a document must be pertinent in the sense that it should capture the flow of information. For example, semantic relationships upon which a flow of information may be based must be accurate, otherwise there is little benefit in applying transformations on documents.

For the sake of clarity, the Transformation Principle does not specifically mention the quantitative components concerned with the representation of uncertainty and significance. In practice, two additional functions are used in tandem with $d \Rightarrow q$ and $d \rightarrow q$; these are w and r respectively. The two quantities $w(d \Rightarrow q)$ and $r(d \rightarrow q)$ are evaluated. The former assesses the significance of q in d, and the latter assesses the extent to which q is implicit in d. The evaluation of $r(d \rightarrow q)$ includes the value $w(d' \Rightarrow q)$ and the uncertainty attached to the transformation of d to d'. The value $r(d \rightarrow q)$ estimates the degree of relevance of the document represented by d to the query represented by q. So the evaluation of $\Rightarrow$ and $\rightarrow$ remains qualitative, while w and r express the uncertainty inherent in this evaluation. Both functions w and r are described in details in Chapter 3.

The Transformation Principle is used in this thesis instead of the Logical Uncertainty Principle for several reasons. First, the transformation process is applied to the document instead of the data set because it is unclear how to transform knowledge, yet it is more intuitive to transform

documents based on the existing knowledge. Furthermore, it is easier to avoid inconsistency by keeping a fixed knowledge, and by varying (enriching) the information content of the document using the knowledge available (but see [NLB96, Nie90, CvR95a, CvR95b] for other uses of the Logical Uncertainty Principle).

A second reason for using the Transformation Principle is that transformation ensures a more general principle which allows any type of information processing, and not just addition of information. Transformation can be addition, deletion or modification of information. A modification can be that a term in the document is replaced by a synonymous or more specific term. A deletion is to indicate what has been achieved so far is incorrect; for example, the system has used the wrong sense of a polysemic term. This indication needs external intervention, for example, a user. The system then has to go back to an earlier state the user recognizes as correct. The model developed in this thesis is not concerned with the intervention of a user, but with information and its flow. Therefore, deletion is not considered, but methods for incorporating deletion in the transformation process are discussed in Chapter 8.

In this thesis, a transformation is either an addition or a modification process, although in many cases, a modification can be viewed as an addition, for no information is discarded.

1.7.1 Examples of transformation

The transformation of documents depends on how documents and queries are indexed. Its appropriate application is essential in modelling the flow of information. Three examples of transformation are discussed in this section, each based on different indexing methods.

1.7.1.1 Documents and queries represented as set of terms

The representations of documents and queries as sets of terms is common to many IR systems. In this case, the transformation of a document can be defined in terms of semantic relationships (e.g., synonymy, related terms) extracted, for example, from a thesaurus. A transformed document contains terms that are semantically related to those used in the original document. The uncertainty of the transformation can be defined from the uncertainty (numerical values) attached to the semantic relationships, where the higher the value, the stronger the relationship.

An appropriate transformation can be ensured in different manners. First, only relationships adequate to a document or a set of documents must be used. This could be achieved by pre-selecting those relationships relevant to a particular document (or a set of them, or the collection itself). For example, if several documents contain the term "bank", and these documents deal with finance, then only the relationships relevant to the finance sense of "bank" must be used. Second, a threshold can be imposed on the uncertainty of the transformation. That is, when this uncertainty is lower than a given value, the transformation is not pursued further.

1.7.1.2 Systems with linked documents

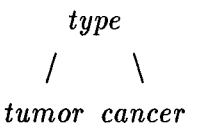
Such systems consist of documents that contain citations, or hypertext documents. In the former, documents explicitly cite other documents; in the latter, documents contain hypertext links to other documents.

With linked documents systems, a transformed document is one that is referred to by another document (via a citation or a link). The uncertainty of a transformation can be defined on the extent to which the original and the transformed documents are similar (the link between documents varies in strength). Similarity measures or statistical-based measures can be used [vR79] for this purpose.

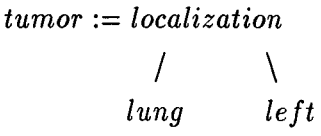
An adequate transformation process may be ascertained by allowing only a certain number of transformations. A better technique would be to compute the similarity (a value) between documents and stop the transformation process when this similarity value is below a given threshold.

1.7.1.3 Natural language information retrieval

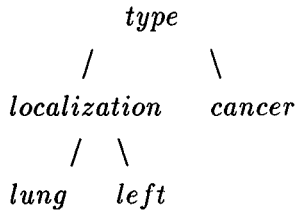
Natural language IR systems make use of natural language process to analysis document's information content and information need, and to evaluate the relevance of a document to a query. Examples of such systems can be found in [Sme92, Nie90], a specific example being RIME developed by Berrut [Ber88] and Nie [Nie90]. Here, documents or queries were indexed by semantic-based trees. For example, the tree



represents the information that the type of tumor is cancer. A transformation consists of deriving trees from original ones. The derivations were based on semantic rules, such as



meaning that the concept of tumor can be refined into one describing its precise location. On the basis of this rule, a document indexed by the above tree can be transformed into a document that contains the following tree (see [Nie90] for details of the transformation process):



Uncertainties were attached to semantic rules expressing their probabilities. The accuracy of transformation could be ensured by allowing a maximal number of transformations, ensuring that only relevant rules were used, or using some threshold values of the uncertainty.

1.7.1.4 Conclusion

Three interpretations of the Transformation Principle were discussed. In the remainder of this thesis, the Transformation Principle is discussed with respect to the first type of systems (section 1.7.1.1), that is, the Transformation Principle is defined in terms of the flow of information based on semantics relationships. However, the work carried is relevant to any type of systems. It is also assumed that the transformation process captures adequately information flows; that is, correct semantic relationships are provided.

1.8 The thesis statement

Van Rijsbergen [vR86a, vR86b, vR89] and Nie [Nie90, Nie88, Nie89, Nie92], explained that current IR models only offer simplistic and specific representations of information, and there is therefore a need for the development of a new formalism able to model IR systems in a more generic manner. I agree with both of them that such formalisms can both be appropriately and powerfully defined within a logic. The resulting formalism should be able to capture information as it appears in an IR system, and also in any of its inherent forms. Therefore, I believe that the time has come to look at some of the most important aspects of an IR system, that is, *information and its flow*.

Information is, and always has been, an elusive concept; nevertheless many philosophers, mathematicians, logicians and computer scientists have felt that it is fundamental. Many attempts have been made to derive some sensible and intuitively acceptable definition of information; until now, none of these have succeeded. Author such as Dretske, Barwise, and Devlin claimed that the notion of information starts from the position that given an ontology of objects individuated by a cognitive agent, it makes sense to speak of the information an object (e.g., a text, an image, a video) contains about another object (e.g., the query). This phenomenon is captured by the flow of information between objects. Its exploitation is the task of an Information Retrieval system.

These authors proposed a theory of information that provides an analysis of the concept of information and the manner in which intelligent organisms (referred to as cognitive agents) handle and respond to the information picked up from their environment. They defined the nature of information flow and the mechanisms that give rise to such a flow.

This theory is the so-called Situation Theory [BP83, Bar89, Dev91], whose aim is the development of a science of information. It is widely recognized that the development of any new scientific tool is better carried out in the abstract. Thus, a science of information should follow a mathematical approach even though the definition of information is itself problematic (which is the case in IR). In the past, this has not stopped scientists from speculating on the nature of objects such as electrons or numbers. Situation Theory can be compared to Quantum Mechanics or Number Theory. In Quantum Mechanics, an ideal representation of an electron is adopted, even if it is not well understood what an electron is. It then becomes possible to model the behavior and the interaction of electrons. In Number Theory, the definition of the number 3 is not clear. Some say it is that set containing three objects. Regardless of the semantics of the numbers 3 and 5 mean, we all know that $3 + 5 = 8$, though there are still arguments about the definition of the numbers 3, 5 and 8.

In this thesis, I show the appropriateness of Situation Theory to model the qualitative components of a logic-based IR model, in particular, the flow of information.

As discussed in the previous section, the quantitative components (the uncertainty and the significance of information) of the model can be represented by a theory of uncertainty. This thesis shows that the Dempster-Shafer Theory of Evidence provides most of the necessary formalisms for the modelling of these quantitative components and in one framework. The use of the overall framework gave the advantage that it could be easily mapped to the qualitative representation of a document, and its transformation. Additionally, it could be suitably mapped onto Situation Theory.

This thesis proposes to use Situation Theory, in tandem with the Dempster-Shafer Theory of Evidence, for constructing a model of an IR system, where Situation Theory largely models the qualitative components of the model, and the Dempster-Shafer Theory of Evidence models its quantitative components. These two theories are combined on the basis of the Transformation Principle.

Two models are proposed, one that caters to an *unstructured* representation of a document, and one that caters to a *structured* representation of a document. This was done in two steps. First, the unstructured model was defined in which the structure and the significance of information were not accounted for. Second, that model was extended into the structured model, which incorporated structures and the significance of information. This strategy was adopted because it enabled the careful representation of the flow of information to be performed initially.

In the first part of this thesis (Chapters 2 to 5), it is assumed that appropriate indexing tools are available, as well as the semantic relationships determining the nature of the flow of information. In Chapter 6 and Chapter 7, the indexing tool and the semantic relationships used to implement the model proposed in this thesis are described. Many problems arise when implementing the models, mainly in extracting the flow of information from documents. This was due to a poor indexing of documents, and inappropriate semantic relationships. These problems are discussed in detail in these two chapters. However, it is shown that if these problems are solved, better performance will be obtained.

The conclusion of this work is that Situation Theory, combined with the Dempster-Shafer's Theory of Evidence, allows the appropriate and powerful representation of several essential features of information in an information retrieval system. Although its implementation presents some difficulties, the model is the first of its kind to capture, in a general manner, these features within a uniform framework. As a result, it can easily be generalized to many types of information retrieval systems (e.g., interactive, multimedia systems), or many aspects of the retrieval process (e.g., user modelling). These applications of the model and others are discussed in Chapter 8.

1.9 Remainder of the thesis

This thesis includes eight chapters, the first being the introduction. Chapter 2 contains a survey of possible qualitative frameworks to capture information and its flows. Chapter 3 is a survey of quantitative frameworks to model uncertainty. Chapter 4 is the description of the model for the unstructured representation of a document. Chapter 5 describes the model for a structured

representation of a document. Chapter 6 describes the implementation of the two models. Chapter 7 relates the experiments and the evaluation of the implemented systems. And finally, Chapter 8 concludes with recommendations for further directions.

Chapter 2
Qualitative Theories for a Logic-based Model of an Information Retrieval System

2.1 Introduction

This chapter examines different theories that can be used to model the qualitative components of an IR system based on the *Transformation Principle*. Five qualitative entities are defined by this principle:

Document	d
Query	q
Knowledge Set	K
Explict Information	$d \Rightarrow q$
Implict Information	$d \rightarrow q$

Table 2.1: The qualitative components

The symbol d is *the representation of the document.* The symbol q is the *representation of the query.* The symbol K is the *representation of the knowledge set.* The notation $d \Rightarrow q$ indicates that the information represented by q is *explicitly* contained in the document represented by d. The notation $d \rightarrow q$ denotes that the information represented by q is *implicitly* contained in the document represented by d. This means that d can be transformed to a document represented by d' such that $d' \Rightarrow q$. Although d and d' are referred to as documents (this is done for clarity of expression), in practice, they correspond to two different representations of the same document, the latter being more "exhaustive" than the former. The transformation is either an addition or a modification process and is dependent upon the knowledge set K. The characteristics of these five qualitative components must be identified to determine the best framework to model them. These characteristics are discussed in section 2.2.

There are three main types of qualitative frameworks that can be used to model the qualitative components of an IR system: ones based on *truth*, ones based on *meaning* and ones based on *information*. The first are extensions of Classical Logic and deal with specific needs such as

modals, partiality or non-monotonic reasoning. They consider the notion of truth as primordial. They are referred to as *truth-based frameworks*, and are described in section 2.3. The second are somewhat concerned with a trade-off between truth and information. They aim to represent the meaning of information, and can be looked upon as *semantics-based frameworks*, and are described in section 2.4. The third are frameworks oriented towards a formalism of information, and treat truth as a secondary concept and are principally concerned with the representation of information content on the basis of information itself. These can be regarded as *information-based frameworks*, and are described in section 2.5.

This chapter describes frameworks for each type and highlights their advantages and disadvantages in modelling some or all of the qualitative components listed above. This survey is not exhaustive, but does cover a wide range of frameworks. The purpose of this chapter is to show that an information-based framework is the most appropriate one.

2.2 The characteristics of the qualitative components

An IR model based on the Transformation Principle defines five qualitative components. The first two components are the representation of the document and the representation of the query. The characteristics of these components are discussed in sections 2.2.1 and 2.2.2, respectively. The other three components are the representation of the knowledge set, the representation of the explicit information content of a document, and the representation of the implicit information content of a document. These three components are the basis of the transformation of a document, and their characteristics are discussed in section 2.2.3.

2.2.1 The representation of a document

One task of an IR system is the *representation* of a document. The representation should reflect the fact that a document is a *provider of information*. Indeed, an IR system determines relevance by checking whether an *information item* is contained in a document. This implies that a document should not be represented by a proposition, as suggested by the discussion of Classical Logic in Chapter 1.

The representation of the document should capture the *partiality of information* because an exhaustive representation of the information content of any document is rarely achieved. This means that the information content of the document is partially captured, but can grow as the representation of that document is transformed into successive less partial representations. This issue is discussed further in section 2.2.3.

In a document, the expressions used to convey information (e.g. words, sentences, etc.) can be *intensional*; that is, the expressions may have multiple meanings. For example, polysemic words such as "bank" are intensional expressions because "bank" has, at least, two meanings, the "money bank" and the "river bank". Another example is the person referred to by the title "Prime Minister"; in Canada this person is J. Chretien, whereas in Great Britain it is J. Major. The embodiment of intensionality necessitates the understanding of the meaning of an expression in a given context. This is a semantic process, the result of which affects the flow of information arising from this

expression. For example, if the term bank is used in the money context, then only flow that is in according to this context arises, and not the flow related to the river context. Therefore, the representation of intensionality in the IR model is important[3].

A document has an underlying structure. For example, a document may consists of a title, a list of authors, an abstract, some keywords, the text itself, several chapters or sections, and some figures. A multimedia document may contain a mixture of text, image, and video. The structure of a document can also be implicit. For example, a structure may consist of the information (e.g., terms) contained in the document, which defines a document topic. Such types of structures are based on semantics because they take into account the fact that information can be semantically related. For reasons of simplicity, only semantic-based structures are considered in this thesis. However, this work is relevant to any type of structure. This issue is discussed in Chapter 8.

An example of information that is *semantically* related is that of equivalent items of information. Take for example the representation of a document's information content as a set of terms. For instance, the "Canadian Prime Minister" and "J. Chretien" are two *equivalent* terms. A document should not be more relevant to a query that uses the two terms "Canadian Prime Minister" and "J. Chretien" in its expression, than to a query that uses only one of these terms, because the information need is the same. Indeed, the first query uses two different terms to refer to the same item of information (here a person), whereas the second query uses only one term. This equivalence of the information need can be taken into account by grouping equivalent terms into *structures*, and treating the groups of equivalent terms as entities. This approach leads to a semantic-based structured representation of the document's information content and subsequently necessitates the representation of a structure.

2.2.2 The representation of a query

An *information need* is communicated to the IR system by a query. In this thesis, the query is not weighted; that is, its expression does not indicate that one item of information is more essential than another. Therefore, a query is modelled by whatever symbolizes information items.

2.2.3 The representation of the transformation process

The Transformation Principle states that the information expressed in the query q is implicit in the *original* document d if that document d can be transformed to a document d' that explicitly contains the information expressed in the query. This principle is based on the observation that the representation of the *information content* of a document, as determined by the indexing process, is often *partial* and depicts usually the *explicit information content* of the document. Additional information can be identified as part of the document information content. This additional information constitutes the *implicit information content* of the document. The recognition of this information comes from the *flow of information* that arises from *some of* the explicit information content of the document, and yields *some of* that implicit information content. The flow of information is a fundamental component of an IR system, and must be adequately represented.

[3] The adequate capturing of the intensionality is a well known problem in philosophical logic [PtMW90]. This is briefly discussed when Intensional Logic is presented in section 2.4.1.

The approach adopted in this thesis is to define the transformation of a document in terms of the flow of information; the explicit information constitutes the original document and the implicit information constitutes the transformed document.

The flow of information characterizes *information containment* and can generally be defined as the information an *object contains* or *carries* about *itself* or another *object*. The information containment is described by informative *relationships* between *items of information* and the *object* affected by this information containment. Let it be a relationship between the two information items p and p' (e.g., synonymity). The flow of information based on this relationship indicates that an object which contains the information item p contains or carries the information that itself or a second object contains the information item p'. For example, in a hypertext system [Con87], a flow of information often arises between two linked documents (the two objects). The relationships determine the nature of the flow.

The representation of the flow of information requires the representation of the relationships upon which the flow is based, and the objects affected by the flow. The knowledge set K consists of the identified relationships between the information items. The objects are the documents, one being the transformation of the other.

The relationships stored in K can be used in sequence, leading to a sequential transformation, or in parallel, leading to a parallel transformation. Both types of transformation are caused by a flow of information, or combination of flows of information either in *sequence* or in *parallel*; this combination constitutes a flow of information. In the first case, a flow which emanates from the explicit information content of the document yields implicit information from which a second flow can emanate, and so forth. For example, suppose that a document about "wine" contains information about a second document on "Chardonnay" which itself contains information about a third document on "Australian wine"; in that case, the first document being about "wine" may contain information about that third document on "Australian wine".

In the second case, the explicit information content of a document can originate simultaneous flows of information, all leading to the same item of information. This can be interpreted as an accumulation of evidence about that item of information. For example, a document about "wine" can contain information about "Chili" because of an explicit reference in that document that many good wines come now from Chili, or that one knows that the United Kingdom is importing many Chilean wines.

The flow of information can be either *certain* or *uncertain*. For example, consider the synonymous relationship as the basis of a flow. If the two terms t and t' have the same meaning in every context, the corresponding flow is certain. Often, two synonymous terms have different meanings in certain contexts. If it is not known which sense the term t refers to in a given context, then the flow that relates this term to t' is uncertain. The relationship between t and t' might not be appropriate with respect to that context. Thus, since a transformation is based on the flow of information, the uncertainty of a transformation is characterized by the uncertainty of the flow of information causing that transformation. The uncertainty engendered by the transformation can be the basis of a numerical formulation of relevance. In this chapter, only the qualitative nature of a transformation is considered; in the next chapter, methods to quantify the uncertainty of a transformation are described. However, the uncertain nature of the flow of information should be borne in mind while examining the different frameworks.

Note that an uncertain transformation, not only represents the flow of information, but it also performs a reduction of ambiguity. An uncertain transformation arises because of uncertain information (e.g., the sense referred by a term is unknown in the document), and represents one way, among possibly several others, to interpret this uncertain information.

2.2.4 Conclusion

The qualitative components of an IR model, together with their characteristics, are summarized in the following table:

Qualitative components	Characteristics
Representation of the document (d)	- Provider of information - Partiality of information - Intensionality (contexts) - Structure (Semantic-based)
Representation of the information expressed in the query (q)	- Information items - Non weighted
Representation of the relationships of the knowledge set (K)	Informative relationships
Representation of the explicit containment of an information item in a document ($d \Rightarrow q$)	The information is part of the document information content
Representation of the implicit containment of an information item in a document ($d \rightarrow q$)	- The document can be transformed (in sequence and/or in parallel) to a document that contains the information item. - The transformation is defined in terms of the flow of information.

Table 2.2: The qualitative components and their characteristics

The modelling of the flow of information necessitates the following representations:

(i) the relationships upon which the flow is based,
(ii) the documents affected by the flow of information,
(iii) sequencing and parallelism of the flow, and
(iv) uncertainty of the flow (qualitative).

In the remainder of this chapter, different frameworks are examined to determine which one best models the above listed qualitative components.

It was stated in Chapter 1 that two models are proposed in this thesis; one that accounts for an *unstructured representation of a document* and one that accounts for a *structured representation of a document*. The study of the different frameworks, except for Scott Domains [Sco82, Lan71], is performed with respect to the unstructured representation, unless otherwise specified. The reasons are twofold. First, the structured representation of a document is a generalization of the unstructured representation of that document. Second, it enables to concentrate on an appropriate representation of the flow of information. The structured representation of a document is discussed in section 2.5.3, where the theory of Scott Domains is described.

2.3 Truth-based frameworks

The weakness of Classical Logic for modelling an IR system comes from the representation of information and informative relationships (used to reason) by truth formulae[4] and the semantics attached to different connectors. To capture the informative relationships, thus constituting the knowledge set, only the interpretations that make their corresponding formulae true are considered in the evaluation of $d \rightarrow q$. The tautologies are also part of the knowledge set. The problem with such a representation of the knowledge set is that the informative relationships or the tautologies cannot be used as the basis of a transformation. Indeed, if a document is represented by a model D (as discussed in Chapter 1) and that $p \rightarrow q$ is an informative relationship, then $D \models p \rightarrow q$. If $D \models p$, then $D \models q$. This means that the transformation of a document represented by the model D into a document represented by another model cannot be expressed. It is not possible to capture that information may subsequently become available in a document (model), where it was not initially available.

Truth-based frameworks are extensions of Classical Logic. A number of them are examined in this section to decide whether they offer an appropriate modelling of an IR system as described in section 2.2. Four frameworks are considered: *Three-valued Logic* [Kle67], *Modal Logic* [HC68, Che80], *Belief Systems* [Gar88, Moo80, Rei80, Mor92], and *Cumulative Logic* [KLM90]. For simplicity, in the remaining of this thesis, informative relationships will also be referred to as tautologies[5].

2.3.1 Three-valued Logic

Three-valued Logic [Kle67] is a model-theoretical framework which introduces a *third* truth value denoted u to indicate a state of ignorance[6]. Three-valued Logic is defined by a set of *models* and a set of *propositions*. In a model M, u is assigned to a proposition p if it is unknown whether p is true or false in M. A model in which the truth value of at least one proposition is u is called *partial*; otherwise, it is called *total*.

A partial model M can be *extended* into a model M' where a proposition p with truth value u in M is resolved. That is, either $M' \models p$ or $M' \models \neg p$. The extension of M into M' is denoted $M \preceq M'$. Some of the partiality is resolved but never revised; something which was known to be either true or false in M remains either true or false in M'.

Three-valued Logic is often used to model systems that are in state of partial ignorance, and which never discard or revise information, and that acquire new information. Therefore a monotonic function G is regularly used in tandem with Three-valued Logic[7]. This function, which can be viewed as a line of reasoning, relates partial models to other partial models.

A model M such that $M \preceq G(M)$ admits a *minimal fixed point* with respect to G. The fixed point is a model M^* such that $M \preceq M^* = G(M^*)$; it represents all the information that can be

[4] In the remainder of this thesis, the terms proposition and formula are used equivalently unless otherwise stated.
[5] As explained in Chapter 1, to capture informative relationships, only some interpretations must be considered. However this did not resolve the problems encountered by the use of Classical Logic to model an IR system.
[6] There are different interpretations of this value leading to different semantics [Tur84]. Kleene's value is mentioned here.
[7] Monotonicity means that if $M \preceq M'$ then $G(M) \preceq G(M')$.

generated from M by means of G.

If a document is represented by a model D and information items are represented by propositions, the partiality of information content is captured with the truth value u. The transformation of the document can be modelled as a monotonic function G. The fixed point D^* can be viewed as the 'maximal' representation of the document, which contains the explicit and the implicit information in the document that can be obtained from G. If q is the proposition representing the query, the evaluation of the relevance consists of determining whether $D^* \models q$. The use of Three-value Logic for modelling an IR system is summarized below:

d	Partial model	D
q	Formula	q
K	Tautologies	$\models p$
$d \Rightarrow q$	Document satisfies the query	$D \models q$
$d \rightarrow q$	The fixed point $D^* = G(D^*)$ satisfies the query	$D^* \models q$

Table 2.3: The modelling of the quantitative components with Three-Valued Logic

In Three-valued Logic, the extension of a model does not depend on specific relationships between information items; the knowledge set is simply the set of tautologies which cannot be used to transform a document. As a result, the extension as defined in Three-valued Logic cannot model the flow of information. Indeed, a proposition p, whose truth value is u in D, does not become true or false in some extension D' of D because a proposition q true in D contains p. The monotonic function G is not defined explicitly in terms of informative relationships.

There are other problems with the use of Three-valued Logic to construct the model as aimed in this thesis. First, intensionality is not represented. Indeed, nothing suggests a function G being used instead of another function G', meaning that there is no notion of one line of reasoning being selected against another one. Second, different lines of reasoning might exist; several G_is, each of them may lead to a fixed point D_i^*. Three-valued Logic does not combine the G_is, and thus cannot model parallel transformations. The combination has to be defined outside the logic.

Three-valued Logic cannot be used appropriately to model the flow of information. Therefore, it is not adopted to model the qualitative components of an IR system. Three-valued Logic is better at modelling monotonic systems that acquire information from their environment, not from the information they already contain.

2.3.2 Modal Logic

A Modal Logic [HC68, Che80] is a model-theoretical extension of Classical Logic. It attempts to deal with *modal operators* [vB85] such as "possibly" and "necessary". Let P be the set of propositions, and let the logical connectors defined in the first chapter be part of the alphabet. Two modal operators $\Box$ and $\Diamond$ are added to the alphabet. They mean "it is necessary that ..." and "it is possible that ..."[8], respectively. A set W of *possible worlds* [Kri63], upon which the

[8] There are other modals that represent past, future, beliefs, etc [vB83]. They are necessary for robust linguistic processes, such as those used in question-answering systems.

semantics are defined, is added to Classical Logic ontology. Possible worlds are related by the so-called *accessibility relation* R. This binary relation captures the intuition that from a possible world w, some other worlds might be deemed possible, which would not be the case from a world different to w.

A model for a Modal Logic is a structure $M = \langle P, W, R \rangle$. In that model, the fact that p is true (false) with respect to a world w is written $M \models p[w]$ ($M \models \neg p[w]$). A formula $\Box p$ is true in a possible world w if p is true in every possible world accessible from w. A formula $\Diamond p$ is true in a possible world w if p is true in at least one possible world accessible from w[9]. There are different types of Modal Logic; their differences result from interpretations or constraints attached to the accessibility relationship[10]. A detailed exposé of Modal Logic can be found in [HC68] and [Che80].

An IR model based on a Modal Logic was developed by Nie [Nie90, Nie88, Nie89, Nie92]. A document is a world w, a query is a proposition q, and the IR model is the structure $M = \langle P, W, R \rangle$. The document is relevant to the query whenever $M \models \Diamond q[w]$; that is, there exists a world w' accessible from w (wRw') such that $M \models q[w']$. In Nie's model, the accessibility relation represents the transformation of the document. The accessibility relation is transitive; hence, it enables the modelling of sequential transformation. The modal $\Diamond$ somewhat captures the partiality of the information in a document because a proposition can be true in some accessible world, though the proposition is not true in the world representing the document (i.e., $M \models q[w']$ and wRw'). This indicates the relevance of the document w to the query q. The relation R may be used to capture the flow of information; the world w contains information about the world w' if wRw'. A model based on a Modal Logic is summarized in the table below:

d	Possible world	w
q	Formula	q
K	Tautologies (with respect to M)	$M \models p$
$d \Rightarrow q$	Model of a formula in a document world	$M \models q[w]$
$d \rightarrow q$	There exists w' such that wRw' and $M \models q[w']$	$M \models \Diamond q[w]$

Table 2.4: The modelling of the quantitative components with Modal Logic

However, the information containment cannot be expressed explicitly because there is no mention of the relationships causing the flow of information. The knowledge set consists of tautologies[11] with respect to M. That is, only deduction about the world itself is explicitly represented. This drawback was also observed with Three-valued Logic. The advantage of Modal Logic over Three-valued Logic is that the representation of parallel transformation is formally embedded in the transitivity of the accessibility relation. In addition, modification can be represented easily because a world accessible from another one does not necessarily contain all the information of the second world. This was not possible with Three-valued Logic where only the addition of information could be considered.

The use of Modal Logic, however, presents a problem resulting from the interpretation attached to

[9] A formula whose truth value in a world depends on the truth values of its parts in other worlds is also referred to as *intensional*.
[10] For example, R can be reflexive, symmetric or transitive.
[11] More correctly, a tautology is a sentence that is true at each world in each model. For the precise definition, see [HC68, Che80].

the accessibility relation. Although the transformation of w into w' (i.e., wRw') may render the fact that the information content of w' is at least partly determined by the information content of w (i.e., there is a flow of information between w and w'), as previously mentioned, the explicit nature of the flow is unknown. This is because the use of the accessibility relationship to model transformation cannot distinguish between transformation (the existence of a flow) and what makes the transformation (the nature of that flow). The informative relationships are therefore not explicit in the model, and their embodiment requires outside concepts. As it is shown later in this chapter, frameworks that enable both the representation of a transformation and its nature exist. Therefore, Modal Logic is not used in this thesis to represent the qualitative components of the IR model.

Another weakness of Modal Logic is that partiality is not represented consistently since all the propositions are evaluated with respect to every world. Indeed, the fact that an item of information represented by that proposition is implicit in a world implies that the proposition is false in that world but true in an accessible world. This is not a correct representation of the partiality, although some extensions of Modal Logic deal with partiality [PS86].

2.3.3 Belief Systems

Belief systems consist of a set of beliefs and a set of implicit or explicit procedures for acquiring new beliefs. The motivation behind belief systems is to model systems that are forced to make decisions in the light of incomplete information such that the possibility of failure may lead to the revision of some assumptions and the subsequent rejection of some conclusions. Belief systems are forms of non-monotonic logic, that is, frameworks in which the introduction of new information can invalidate old information. Three types of belief systems are described: *Default Reasoning* [Rei80], *Belief Revision* [Gar88] and *Epistemic Logic* [Moo80]. The descriptions of these frameworks are axiomatic-based (see Chapter 1).

2.3.3.1 Default Reasoning

Default Reasoning is concerned with the modelling of assumptions of the form "birds usually fly" which are assumptions that sometimes turn out to be ill-founded (e.g., birds such as penguin or ostrich). One instance of Default Reasoning is *Default Theory*, which was proposed by Reiter [Rei80]. This framework is composed of two parts, a set of axioms and a set of *default rules*:

$$\frac{A : B}{C}$$

This rule indicates that if A is true, and if B is *not known* to be false, then infer C. The truth or falsity of B is based on the *closed-world assumption* [Rei78]; that is, if B cannot be proven false, then B is considered true. The set of axioms constitutes the *basic theory*. The application of the default rules to the basic theory constitutes an *extension*, which consists of a deductive-closed and consistent set of propositions. From a basic theory, several extensions can be obtained since default rules can lead to different conclusions, thus capturing the non-monotonicity of the reasoning[12] .

A document can be modelled by a basic theory D, which is the set of axioms that represent the explicit information content of the document. The tautologies and the default rules can constitute

[12] That is, if one knows A is true, e.g., "I put sugar in my coffee", one can infer C is true, e.g., "my coffee tastes sweet". However, if one now knows that B is false, e.g., "I put olive oil in my coffee", one may not infer that C is true anymore. Classical Logic is monotonic because if $p \vdash q$, then $p \wedge r \vdash q$.

the knowledge set K. The transformation of a document comes from the application of default rules to the basic theory that models the document. An extension of the basic theory constitutes a transformed document. The nature of the transformation of the document is explicitly represented by the default rule used to perform that transformation. The transformation is physically represented by the extension built from the application of the default rule. These two features present an advantage over the frameworks so far described, since both the transformation and its nature are represented. A model based on Default Theory is summarized in the following table:

d	Basic Theory	D
q	Formula	q
K	Default Rules Tautologies	$\frac{A:B}{C}$ $\models p$
$d \Rightarrow q$	The basic theory contains the query formula	$q \in D$
$d \rightarrow q$	There is an extension D' that contains the query formula	$q \in D'$

Table 2.5: The modelling of the quantitative components with Default Theory

The extension of the basic theory is based on the fact that B cannot be proved false, not on the fact that B is effectively proven true. Such an assumption conflicts with the partiality feature of information. That is, although it is not possible to know whether any information item is contained in the document, this item should not be considered false with respect to the document until it has been proven (or defined) as such. Default Theory cannot capture this phenomena because it is based on the closed-world assumption. A second problem is that default rules do not embed correctly intensionality. This is more evident with *normal default rules*[13] of the form:

$$\frac{A : C}{C}$$

This rule signifies that if A is true, and if it is correct to assume C, then infer C. The acquisition of implicit information (C) is based on the fact that its existence cannot be denied. Therefore, intensionality is captured in the fact that no inconsistencies are introduced, and not on its explicit characterization.

Default Theory is not the best framework for modelling the flow of information, since it is based on the closed-world assumption. Therefore, it is not used for qualitatively modelling the IR system. Default Theory is more useful for modelling non-monotonic reasoning, where it is understood that any information can be inferred as long it does not conflict with information already available, which is not what information flow is about.

2.3.3.2 Belief Revision

Belief Revision provides a formalism for revising a *belief state* (a set of beliefs) in light of new, possibly conflicting, information. The revision results in a belief state which contains the new information and as much of the original belief state as possible, whilst staying consistent. The standard theory of Belief Revision is known as the *AGM Theory* [Gar88]. Belief states are deductively closed sets of sentences. If s is a belief state and φ a proposition, then $s * \varphi$ is

[13] Normal default rules are most used in Default Theory.

the *revised belief state*. $*$ is called the *belief revision function*. In $s * \varphi$, the new belief φ should be true and the old beliefs should persist through revision if they can. $s * \varphi$ should not contain extraneous information which was present in neither the old state nor the new belief. The belief revision function is generally defined as follows:

$$s * \varphi = \begin{cases} Closed(s \cup \{\varphi\}) & if \neg\varphi \notin s \\ Consistent(s \cup \{\varphi\}) & otherwise \end{cases}$$

$\neg\varphi \notin s$ means that no conflict arises. $Closed(s \cup \{\varphi\})$ is the deductive closure of $s \cup \{\varphi\}$. $Consistent(s \cup \{\varphi\})$ is the set of (consistent) beliefs after revision, which can be constructed by different methods. In one, the maximal sets of consistent propositions in $s \cup \{\varphi\}$ are computed, all of which contain φ; their intersection constitutes $Consistent(s \cup \{\varphi\})$.

A model based on Belief Revision represents a document by a belief state, a query by a proposition, and the transformation of a document by the application of the belief revision function. However, Belief Revision is inadequate for the representation of information containment because the proposition φ is a new belief; it is external to the information contained in the belief state, and is not inferred from the information that is explicit in the document. Moreover, the acquisition of this new information might refute the information that constitutes the original belief state, that is, the explicit information content of the document. In this thesis, the representation of the explicit information content of the document is assumed correct, although it may not be an exhaustive representation of the information content. A transformation is either an addition or a modification of information. The fact that a document is modified into a second document does not mean that some of the original information is discredited in the second document; the information is only different. A belief revision and a transformation are different processes, so the former cannot be used to model the latter. As a consequence, Belief Revision is not selected for modelling the qualitative components of an IR system.

Belief revision is more appropriate in the modelling of a user in the IR system. There, the beliefs of the user can be included in an IR session, and can contradict past beliefs, so a belief revision is therefore necessary. Research on this area of study can be found in [LRJ94, CRJ92].

2.3.3.3 Epistemic Logic

Epistemic Logic [Moo80] is a variant of Modal Logic. The necessary modal $\Box$ is replaced by a family of operators K_a, where $K_a p$ means that a *knows* p[14]. Based on a possible-world semantics approach (see section 2.3.2), $K_a p$ is true in a possible world w iff p is true in every possible world w' that is an epistemic alternative of w. Epistemic Logic aims to model the reasoning of an ideally rational agent, reflecting his or her own knowledge. It takes a set of sentences as a theory, which represents the total knowledge of that agent. An accessibility relation is used, which can be viewed as an alternative epistemic state of the agent. The axioms that define Epistemic Logic are similar to those defined in Modal logic. The main difference is the presence of an argument which *relativizes* each axiom to a particular owner.

A possible application of Epistemic Logic in IR is to consider an agent either as a type of IR system (for example, Boolean, Probabilistic or Logical), a component of the IR system (for example, the system, the indexer, or the user) or as a type of an IR session (for example, user modelling).

[14] Or operators B_a where $B_a p$ means "a believes p".

Different types of reasoning could be consistently modelled, compared or combined. Such IR agents have been proposed in [HvL96]. Another approach can be found in [Seb94], where Autoepistemic Logic (an Epistemic Logic that concerns one agent) is used to model subjective-based beliefs to distinguish this type of belief from frequency-based beliefs. The model is, however, irrelevant to the notion of the flow of information in an IR system.

Epistemic Logic is concerned more with modelling knowledge and action, and could be used at the meta-level modelling of an IR system, where different agents are considered. On its own, Epistemic Logic does not help to model the flow of information.

2.3.4 Cumulative Logic

The axiom-based system of Classical Logic uses the derivability relation $\vdash$. For two propositions p and q, $p \vdash q$ means that q is derivable or inferred from p (see section 1.4 of Chapter 1). The derivation consists of a finite sequence of axioms or applications of inference rules. As stated in Chapter 1, the derivability relation is much too rigid for representing the flow of information. Also, it allows, among others, the following two inferences:

(i) if $p \vdash q$ then $p \wedge r \vdash q$
(ii) if $p \vdash q$ then $p \vdash q \vee r$

The irrationality of (i) can be illustrated by the following example : "if it does not rain, I will not get wet" implies "if it does not rain and I jump in the water, I will not get wet". (i) holds because the derivability relation is a monotonic inference reasoning (see footnote in section 2.3.3.1). An example of (ii) is as follows: "if I work hard, I will finish this report today" implies "if I work hard, either I will finish this report today or frogs are green creatures". (ii) comes both from the transitive property of the derivability relation and the semantics attached to disjunction which do not capture the intensional nature of information.

Weaker derivability relations have been proposed, and one of the weakest is defined in *Cumulative Logic* [KLM90] as the *consequence relation*, denoted $\mid\!\sim$. Given two formulae p and q, $p \mid\!\sim q$ means that p *normally implies* q. $p \mid\!\sim q$ is called a *conditional assertion* or simply an *assertion*. An advantage of Cumulative Logic is that the assertion $p \mid\!\sim q$ is evaluated only if p is true. This is not the case in Classical Logic because the evaluation of $p \vdash q$ includes the cases where p is false[15].

Cumulative Logic addresses three types of knowledge. The first type is unconditional constraints such as "roses are flowers" or definitions such as "tall is equivalent to not short". They are represented by tautologies, for example, $\models p \rightarrow q$ or $\models p \leftrightarrow q$. The second type is facts that describe a situation, and are represented by formulae. The last type is conditional assertions which constitute a database.

The reasoning process of Cumulative Logic is as follows: given two formulae p and q, to answer q from a situation described by p is to infer $p \mid\!\sim q$ from the database. The reasoning process in Cumulative Logic is different to that in Classical Logic. In the latter, given two propositions, to

[15] This is because $p \vdash q$ is equivalent to $\vdash p \rightarrow q$. In contrast, Cumulative Logic does not allow the expression of $\mid\!\sim p \rightarrow q$. Moreover, Cumulative Logic differentiates between $r \wedge p \mid\!\sim q$ and $r \mid\!\sim p \rightarrow q$, where r is a proposition. If r is true, the first assertion says that if it is the case that p is true, then normally q, whereas, the second assertion is automatically verified if p is false.

answer q from p is to prove that $p \vdash q$, and not to infer $p \vdash q$ from a database[16].

The deduction process is based on five inference rules. They are listed below (in the following discussion, p, q and r are formulae). The first rule is *Reflexivity*:

$$p \mid\!\sim p$$

Reflexivity is satisfied by most derivability relations. The second rule is *Left Logical Equivalence*:

$$\frac{\models p \leftrightarrow q, p \mid\!\sim r}{q \mid\!\sim r}$$

This rules expresses the requirement that logically equivalent formulae have exactly the same consequences. The third rule is *Right Weakening*:

$$\frac{\models p \rightarrow q, r \mid\!\sim p}{r \mid\!\sim q}$$

This rule states that any logical consequence of a formula p is normally implied by a formula that normally implies p. The fourth rule is *Cut*:

$$\frac{p \wedge q \mid\!\sim r, p \mid\!\sim q}{p \mid\!\sim r}$$

Cut expresses that a hypothesis proven plausible from a set of facts can be added to this set of facts without altering anything this set of facts normally implies. The last rule is *Cautious Monotonicity*:

$$\frac{p \mid\!\sim q, p \mid\!\sim r}{p \wedge q \mid\!\sim r}$$

This rule indicates that adding a new fact into a set of hypotheses, the truth of which could have been concluded by this set, should not invalidate previous conclusions.

A model of an IR system based on Cumulative Logic represents a knowledge set by the set of assertions and the set of tautologies. If the document is a formula d and the query a formula q, asserting relevance consists of formally deriving $d \mid\!\sim q$ with the use of the above five inference rules. Although it is preferable that a document should not be modelled by a proposition, this approach is still studied, since a weaker derivability relation may lead to an appropriate model of an IR system.

First, Reflexivity expresses that a document is relevant to itself. Second, Left Logical Equivalence and Right Weakening incorporate tautology-based relationships, but these rules cannot embody

[16] In Classical Logic, the reasoning process involved in proving that $p \vdash q$ has the following general form:

$$\frac{p \ \ldots}{\begin{array}{c}\vdots\end{array}}$$
$$q$$

In Cumulative Logic, the reasoning process involved in proving $p \mid\!\sim q$ has the following general form:

$$\frac{database}{\begin{array}{c}\vdots\end{array}}$$
$$p \mid\!\sim q$$

intensional expressions. Finally, both Cut and Cautious Monotonicity capture intensionality on the basis that no inconsistent derivation is allowed. Cut rejects the transitivity of $\mid\sim$ (this is done by mentioning q in the assertion $p \wedge q \mid\sim r$), which ensures that a context is somewhat preserved with respect to the first assertion $p \mid\sim q$. Although the contexts are not explicitly represented, their effects are embedded in the fact that no inconsistency arises. Cautious Monotonicity is a weaker version of monotonicity, and captures contexts by restricting the q in $p \wedge q \mid\sim r$[17].

A model based on Cumulative Logic presents many weaknesses. First, intensionality is not explicitly represented; its effect is just ensured. Second, the transformation of a document is not physically represented because the document and the result of the transformation of that document are not distinguishable. A better use of the consequence relation would be to find the closest formula to d, for example d', such that $d' \mid\sim q$. However, this is beyond the scope of Cumulative Logic. The disadvantage in not distinguishing between a document and its transformation is that alternative transformations cannot be represented. Moreover, the partiality of information seems difficult to embody because the consecutive representations of a document's information content cannot be modelled. Third, there is no distinction between an assertion that is initially given and one that is inferred. Assertions are part of the knowledge set; they constitute informative relationships, and determining relevance consists of deriving an assertion. Hence, the representation of a relationship of the knowledge and the implicit containment are modelled in the same manner. Finally, the evaluation of the relevance of a document as the proof of $d \mid\sim q$ means that the existence of an informative relationship between d and q is sought. This approach does not view the document as the provider of information. It would be more correct to establish relevance if some informative relationships exist between the information contained in the document and the query.

In conclusion, although $\mid\sim$ is a weaker derivability relation and it captures contexts better than $\vdash$, it is inadequate to build an IR model that caters to the representation of the flow of information.

Other types of consequence relations have been used in IR. For example, Bruza [Bru93] defines a consequence relation at the level of information (referred to *index representation*) to model plausible inference, and not at the level of document versus query. An example of such an inference is as follows:

$$pollution\ in\ Australia \mid\sim water\ pollution$$

The model is developed at a linguistic level and does not cater to higher-level relationships such as those based on the flow of information.

2.3.5 Conclusion

None of the truth-based frameworks were successful in modelling the flow of information as defined in this thesis. Three-valued Logic and Modal Logic frameworks model the flow of information by a monotonic function and an accessibility relation, respectively. In both frameworks, the nature of the flow is not explicitly captured, and is simply modelled by the fact that two representations of a document are linked together. Default Theory bases its reasoning on the premise that some

[17] Cut can be problematic if the evaluation of $p \mid\sim q$ is done in parallel with an uncertain mechanism. That is, if $p \mid\sim q$ means that q can be derived from p with certainty greater than 0, $p \wedge q \mid\sim r$ signifies that r can be obtained from $p \wedge q$ with certainty greater than 0. There is no evidence to conclude that r can be obtained from p with certainty greater than 0. As for Cut, an interpretation of Cautious monotonicity that uses an uncertainty measure invalidates this rule. For example, $p \mid\sim q$ and $p \mid\sim r$ cannot capture the fact that sometimes q is inferred from p, and other times r is inferred from p. It is incorrect to infer $p \wedge q \mid\sim r$ since q and r are alternative inferences from p.

information cannot be proven false, which is not in accordance with the flow of information. Belief Revision suffers the drawback that it refutes the information that initially constitutes the document. Also, new beliefs are acquired without necessarily knowing how they were obtained; they do not come from the information containment. Epistemic Logic cannot be used to model the flow of information because it has another philosophy that is not concerned with the flow of information. Cumulative Logic offers a weaker inference mechanism, and is better than the derivability relationship in Classical Logic, but still presents many deficiencies with respect to the modelling of information flow.

2.4 Semantic-based Frameworks

The purpose of a semantics-based framework is to model the meaning of information expressed in a natural language. Semantics-based frameworks are based on truth, but they are more appropriately used for the representation of the meaning of information. Three frameworks are presented: *Intensional Logic* [PtMW90], *Montague Semantics* [DWP81, Mon74] and *Data Semantics* [Lan86].

2.4.1 Intensional Logic

The embodiment of contexts is an important goal of IR. Indeed, natural language is ambiguous so an expression may have different meanings in different contexts. Such an expression was qualified as intensional. The meaning of an expression sometimes depends on the meaning of its sub-expressions in other contexts[18]. This type of expression is also qualified as *intensional.*

The reason for developing Intensional Logic is usually illustrated with the use of *referential* noun phrases, that is, noun phrases that refer to objects or individuals. In the sentences "Hesperus is Phosphorus" and "Hesperus is Hesperus"[19], "Hesperus" and "Phosphorus", both proper names, are referential noun phrases. The two sentences express true statements because both noun phrases refer to the same object, the planet Venus. The noun phrases have the same *semantic value*[20]. If the semantic value of a statement is a truth value, two referential expressions with the same semantic value may be substituted for each other without changing the truth value assigned to the statement[21]. However, the first statement is informative while the second is not. Indisputably, truth value alone is an insufficient semantic value for a statement. As Frege explains [Fre60], the semantic value of an expression involves two entities: the *reference* and the *sense*. Proper names or other referential noun phrases may refer to the same objects or individuals, but they differ in sense. Consequently, identity statements[22] are informative when they are constituted of expressions with different senses; they are true when they refer to the same objects. Similarly, tautologies, which are always true statements, may contain different information.

Intensional Logic symbolizes reference and sense by *relativizing* semantic values to *indexes*, which can be viewed as a generalization of worlds. The reference of an expression is defined for each

[18] An example is the semantics of $\Diamond p$ in Modal Logic where a world acts as a context.
[19] This example is taken from [PtMW90].
[20] In addition to a truth value, an object, an individual, a set of objects, a set of individuals or a function can constitute a semantic value [PtMW90].
[21] Substitution of semantic equivalent expressions is an important rule of inference in most truth-based systems (an example is Left Logical Equivalence in Cumulative Logic).
[22] Statements of the form "A is B".

index. It is known as the *extension* of the expression at that index. The sense of an expression, referred to as its *intension*, is a function from indexes to extensions. An expression is *intensional* if its evaluation involves several indexes; otherwise, it is *extensional*[23].

If an index is viewed as a model of a context, then the text of a document is composed of many intensional expressions. For example, a document that mentions the "100m world record holder" may refer to different individuals; "Carl Lewis" in 1991, then, a month later, to "Leroy Burell". Polysemic words are another example of intensional expressions since their sense varies with the context in which they are used. There are many other examples of intensional expressions in natural language (a detailed exposé can be found in [PtMW90]).

An *intensional model* is a structure $M = \langle P, I, R, F \rangle$ where P is the set of propositions, I is the set of indexes, R is the relation between indexes and F is the function that assigns functions, from indexes to extensions, to basic expressions[24]. The semantics of non-basic extensional expressions (for example $p \vee q$, $\neg p$, or $p \rightarrow q$) are defined in the standard way, but with respect to indexes. The evaluation of non-basic intensional expressions takes into account the connection between indexes, where substitution is index-dependent. The equivalence of expressions is defined at different levels, for example, with respect to one index, all indexes, or at the intensional level. A detailed account of Intensional Logic can be found in [DWP81] and [vB85].

A document can be regarded as a set of intensional expressions. With Modal Logic, a document was represented by one possible world. In Intensional Logic, the representation of a document may involve several indexes. One approach[25] is to represent the document by an intensional model $D = \langle P_D, I_D, R_D, F_D \rangle$. Each identified context of the document constitutes an index of I_D. This set may include indexes related to those that are explicitly determined from the document. For example, the model of a document about past wars could include contexts about actual wars. The intensional model captures the document information content by relativizing information with respect to indexes. These are related to each other to express possible links with each other. For example, the index related to actual wars is connected by R_D to the indexes concerned with past wars. The connection can be viewed to some extent as the expression of the flow of information.

A query is represented by an intensional formula. Determining the relevance of the document to the query consists of evaluating the formula q in the model D.

A model of an IR system based on Intensional Logic presents several problems. The first is related to the evaluation of the query, because the indexes related to the query formulae have to be identified. That is, the semantic value of the query formula has to be determined first. This is not a straightforward task, particularly in automatic-based IR systems. A more substantial problem with this approach is that the transformation of a document into another is not symbolized because the document information content, either explicit or implicit, is embedded in the intensional model.

[23] A basic expression can be explicitly indicated as extensional or intensional with the use of two operators $^{\vee}$ and $^{\wedge}$ [PtMW90]. The understanding of these operators is both complex and unnecessary in the purpose of this discussion. Therefore, these operators are not mentioned.

[24] This is a very simplified description of Intensional Logic.

[25] A second approach would be to define a general intensional model, and to model the document by a set of indexes. If the indexes related to the document are considered only, this approach reduces to the previous one. A third approach would be, again, to consider a general intensional model, and to evaluate in this model $d \rightarrow q$; d is the document formula and q the query formula. Implications are extensional formulae, for their evaluation is with respect to a given index; although the formulae d and q can be intensional. This approach is not considered because of the semantics attached to $\rightarrow$. Indeed, in an index, the evaluation of $d \rightarrow q$ is true if d is false.

The transformed document's representation is already included in the intensional model, which formalizes the document's information content. It then becomes impossible to ensure minimal transformation since the evaluation of the query consists of traversing relevant indexes, and finding those that satisfy the query. There is no notion of transforming a document until the information being sought is obtained. This somewhat diverges from the idea of the flow of information as identifying implicit information. Also, the connection between indexes is not necessary defined in terms of information containment.

An additional weakness of Intensional Logic is that partiality cannot be represented because the semantics of $\rightarrow$, $\wedge$ and $\vee$ still hold, though relativized to indexes. For example, the sentences "it is snowing" and "it is snowing and $2 + 2 = 4$" are equivalent because the sentence "$2 + 2 = 4$" is an item of information that is always a true statement. The two sentences "Keith did or did not drink the wine" and "Mounia did or did not drink the wine" are always true statements. This should not be the case because "Keith" and "Mounia" are not relevant to every context. If the references of "Keith" and "Mounia" were not determined at all indexes, then these two statements would not always be true.

Intensional Logic is a theory of meaning, not a theory of information. It is successful in capturing contexts (or more precisely, intensionality), although the implementation of the indexes is not obvious. Indeed, the identification of indexes, together with the way they are linked to each other is very complex, more so for general-purpose IR systems. Intensional Logic is better at representing the relevance of a document to a query because some expressions used in the document are equivalent to those used in the query, not because the flow of information leads to the expressions used in the query. Therefore, Intensional Logic is not used to model the qualitative components of an IR system.

2.4.2 Montague Semantics

The framework of Intensional Logic has been developed independently from natural language, although it is concerned with the intensional nature of natural language. Intensional Logic has also been used to develop a formal semantics of natural language, namely Montague Semantics [DWP81], whose enhancement towards Intensional Logic is that the basic expressions are words of the English language. Also, the syntactic and semantic rules attached to the construction of complex expressions, or groups of words, are specified in parallel.

The basic parts of speech such as noun, verb, determinant, or preposition constitute the basic syntactic categories. An intensional-based semantic value is assigned to each basic syntactic category, which can be as complex as second order *lambda expressions* [DWP81, Chu41]. For example, the determinant "every" relates two entities, like "every *student works*" or "every *dog barks*". The Montague semantics of "every" is a second-order lambda expression $\lambda P\ \lambda Q\ \forall x[P(x) \rightarrow Q(x)]$. It can be used with any two entities whose semantics are defined by P and Q. In Montague Semantics, every basic expression is formally defined with respect to syntax and semantics, thus leading to a uniform and rigorous natural language process.

The basic expressions are combined according to the syntactic rules of English grammar into successively larger expressions. These constitute the complex syntactic categories, such as noun phrase (a preposition, an adjective and a noun), modified verb (a modifier and a verb) or the

outmost category, a sentence. The semantics of a complex expression depends on the semantics assigned to the basic expressions that compose it. This is called the *Principle of Compositionality*, which is represented in the *lambda calculus*. Recursive rules, both at the syntactic and semantic levels, are used for this purpose. The general idea is as follows:

> If R is a syntactic rule taking input $\alpha, \beta, \ldots, \eta$ and gives ϕ, the semantic rule R' takes as input the semantics of $\alpha, \beta, \ldots, \eta$ and yields something which is the semantics of ϕ.

Using the phrase "every dog barks", assume that the semantics of both "dog" and "barks" are two functions *dog* and $bark$. The semantics attached to the whole sentence is then

$$\begin{gathered}(\lambda P\, \lambda Q\, \forall x\, [P(x) \rightarrow Q(x)])\ \ bark\ \ dog \\ (\lambda Q\, \forall x\, [dog(x) \rightarrow Q(x)])\ \ bark \\ \forall x\, [dog(x) \rightarrow bark(x)]\end{gathered}$$

The sentence is translated into a Montague-style formula, and can then be evaluated according to the rules of Intensional Logic.

Montague Semantics considers a fragment of the English language and cannot portray all cases of meaning. For example, it cannot analyze "bachelor" and "never-married adult human man" as synonymous expressions, the determination of which is important in IR. Also, Montague Semantics is not good at representing expressions that are not always related upon their meaning, as in information containment. The fact that the semantic values of two expressions are common at some indexes (or the fact that two expressions are relevant to related indexes) is not a good indication of whether the information carried by one expression is contained in the information carried by the other expression.

A Montague Semantics-based model of an IR system is similar to that developed with Intensional logic; the main difference is that all the English expressions that constitute the text are translated to an intensional formulae, thus achieving a better uniformity. Montague Semantics should be used if the task is to construct a model of the *meaning* of information, not if the *flow* of information is a crucial qualitative component of the IR model.

Sembok [Sem89] has developed a linguistic model of an IR system using a Prolog-based simplified implementation of Montague Semantics [Jow90, Jan80a, Jan80b]. Intensionality is dropped and all the words in the text are assigned a syntactic category and a predicate-like semantic interpretation. Logical and linguistic connectors (like "and", "or", and "to", "of") are explicitly represented by predicates. An example of the translation of an expression in Sembok's model is

$$\begin{gathered}automatic\ analysis\ of\ information\ text \\ \downarrow \\ automatic(w),\quad analysis(x),\quad information(y),\quad text(z) \\ a(w,x),\quad r(y,z),\quad of(x,z)\end{gathered}$$

where "a" is the adjective-noun relationship, "r" is the noun-noun relationship and "of" is self-explanatory. The evaluation of the relevance of the document to a query is based on a Prolog-style unification rule. Sembok's model was not developed to represent the flow of information, but to deal specifically with linguistics.

2.4.3 Data Semantics

Data Semantics, as developed by Landman [Lan86], is a truth-based framework aimed at developing a theory of information. For example, Data Semantics acknowledges the difference between the information expressed in the sentence "the grass is green or the grass is not green" and a tautology. The sentence contains no information on the color of the grass, whereas the tautology is an item of information that is always true. Data Semantics also recognizes the subtleties in the information expressed in the sentences "the grass is green and the grass is not green" and "the grass is green and the grass is blue". The first sentence has no meaning and reveals no information because it is a construction of no fact. The second sentence conveys incompatible information. Classical Logic does not separate these two cases of true and false statements because its semantics is truth-based instead of information-based. Therefore, the meaning of a sentence cannot be computed exclusively on truth. Indeed, the absolute notion of truth and falsity of a sentence makes little sense for its meaning. As Landman [Lan86] explains:

> "You do not ask when a proposition is true or false, but rather what makes it true or false".

Data Semantics recognizes a sentence and the information conveyed by the utterance of that sentence as two separate notions. Bearing this in mind, it proposes a representation of partiality, modality, and conditionality, as practiced in natural language.

The ontology begins with *facts* as *simple propositions*. *Complex propositions* are constructions derived from facts, and are defined with the connectors $\wedge$, $\vee$ and $\neg$. The propositions carry information about *situations* and classify situations by their informational aspect. The propositions are partially ordered by a relation of *information containment* denoted $\preceq$, where $p \preceq q$ represents that the information which p carries about a situation already contains the information that q carries about that same situation. The impossible proposition, denoted $\perp$, is the smallest proposition of which the information it carries contains the information of all other propositions (which is too much information). $\perp$ portrays the incompatible information.

Situations are represented by *information states*. More precisely, an information state s is a set of propositions such that

(i) s does not contain incompatible information ($\perp \notin s$),
(ii) if p and q carry information about s, so does $p \wedge q$,
(iii) if $p \in s$ and $p \preceq q$, then $q \in s$.

Often, not all the propositions that carry information about a situation are identified, and are therefore *partially* identified. However, the information can *grow* to a certain limit and can be eventually made *total*. The growth of information is symbolized by the notion of *extension* of an information state to another. The fact that the information state s_1 is extended into the information state s_2 is denoted $s_1 \subseteq s_2$. This indicates that s_2 contains all the information which s_1 contains, and possibly more.

An information state is a partial representation of a situation. Thus, it can properly model a document. Extended information states model the results of subsequent transformations.

Based on the definition of facts, propositions, information states, and extensions, Data Semantics

puts forward a theory of information focussed on a proper formalism of conditionals ($p \rightarrow q$) and modals (*must* p and *may* p). The truth or falsity of a proposition is relativized to information states. The fact that a proposition p is true on the basis of the information state s (i.e., $p \in s$) is written $s \models p$. The fact that a proposition p is false on the basis of the information state s (i.e., for some $q \in s$, $p \wedge q = \bot$) is written $s \not\models p$. Note that the negation is explicit.

Data Semantics acknowledges two types of propositions: *stable* and *unstable*. A proposition is stable in a state if is not denied in some *extended* states. Example of stable propositions are $p \vee q$, $p \wedge q$ or $\neg p$, providing that both p and q are stable. Examples of unstable propositions are $p \rightarrow q$ or may p. The reason for their instability is explained later in this section.

Here are some basic concepts defined by Data Semantics:

(i) The set $E(s) = \{s' : s \subseteq s'\}$ is the set of extensions from the information state s.

(ii) C is a *chain* of $E(s)$ iff for all $s', s'' \in C$, either $s' \subseteq s''$ or $s'' \subseteq s'$.

(iii) A chain C is *maximal* if it cannot be extended further. A maximal chain is called a *branch*, which can be viewed as a complete way to follow the extension of information starting from s.

(iv) $B(s)$ is the set of branches from s.

(v) An information state s' is called a minimally $p-$verifying information state of a branch of $B(s)$ if s' is the first information state in that branch in which p is true (i.e., $s' \models p$ and no other situation s'' is such that both $s \subset s''$, $s'' \subset s'$[26] and $s'' \models p$).

The last notion is necessary for the representation of unstable propositions. For example, $may\, p$ is an unstable proposition. Its (simplified) semantic is

$$s \models may\, p \text{ iff there is some } b \in B(s) \text{ such that } s' \in b \text{ and } s' \models p.$$

This means that, in case of limited evidence, $may\, p$ is considered true as long as s can be extended to an information state s' such that $s' \models p$. If no such information state exists, $may\, p$ becomes false. As soon as an information state that contains a proposition that contradicts p is reached, the truth value of $may\, p$ changes. It is therefore necessary to identify the information states in which a new item of evidence is acquired because its arrival might change the truth value of unstable propositions.

Although the notion of unstable propositions is not foreign to the notion of the flow of information, these two notions are nevertheless dissimilar. Unstable propositions model the lack of information at one state. Let p be the proposition representing the information whose obtainment is examined at a given state. The proposition $may\, p$ is true as long as p can be obtained in some extended states, whereas the flow of information is more concerned with the fact that the state is extended to another state that contains the information p. These are two distinct phenomena. The expression of unstable propositions is irrelevant to a model of the flow of information. Therefore, Data Semantics ontology is not further[27] described because its purpose is the modelling of unstable propositions, which do not concern this thesis.

[26] $s \subset s'$ is the same as $s \subseteq s'$, with the restriction that s cannot equate s'.
[27] Data Semantics ontology is more expressive than has been demonstrated so far.

Data Semantics can offer a basis for a model of an IR system, although the expressiveness of its ontology is under-used. If a document is represented by an information state d, and q is the proposition representing the query, then $d \models q$ would mean that the document is relevant to the query. If it is not the case that $d \not\models q$, then the document may still be relevant to the query. In this case, the evaluation of the relevance consists of finding a branch that leads to an extension that is characterized by the query proposition. The minimally $q-$verifying state of that branch explicitly represents the first state that makes q true. Transformations are represented by the extensions of information states and the minimality of a transformation can be explicitly ensured. A possible model of an IR system based on Data Semantics is summarized in the following table[28]:

d	Information State	d
q	Proposition	q
K	Information containment	$\preceq$
$d \Rightarrow q$	The proposition belongs to the information state	$q \in d$
$d \rightarrow q$	The minimally $q-$ veryfying extension d' from d exists	$d' \models q$

Table 2.6: The modelling of the quantitative components with Data Semantics

The knowledge set is represented by the set of relationships of the form $p \preceq q$ (and tautologies). Its nature is more elaborate than if it were based (as in many frameworks) only on tautologies. However, information containment is represented with respect to the same information state, not as the basis of an extension. That is, an information state does not become extended on the basis of the information carried by that information state. The extension of an information state occurs from the acquirement of new information, and not from explicit informative relationships.

Although many concepts of Data Semantics seem appropriate to model minimal transformation, Data Semantics is not used in this thesis to model the qualitative components of the IR systems, mainly because the informative relationships leading to the transformation of a document cannot be represented. However, as it will be shown in Chapter 4, some of its terminology is kept; for example branch and minimal extension. Their definition is modified to be compatible with the ontology of the framework used for modelling the qualitative IR components.

Data Semantics has been used in the area of IR modelling, though in a different perspective. Bruza and Huibers [BH94] use Data Semantics to develop a framework in which different models of IR systems can be expressed. Axioms are defined which represent properties of IR systems. Bruza and Huibers aim to develop a uniform framework in which IR systems are compared on some theoretical background, and not on experimental results. Their work is not concerned with the

[28] An interesting point concerns conditional sentences. A simplified version of the semantics of the conditional $s \models p \rightarrow q$ is as follows:

$$for\ all\ s' \supseteq s,\ if\ s' \models p\ then\ there\ is\ some\ s'' \supseteq s'\ such\ that\ s'' \models q$$

It seems that this definition could be used to model the Logical Uncertainty Principle [vR86a], upon which the Transformation Principle is derived. The Logical Uncertainty Principle considers the addition of information to the knowledge set. s could represent the knowledge set, and the propositions p and q could be representations of the document and the query, respectively. The knowledge set is extended to some state where p is true. The document is relevant if this state can be further extended to a state that makes q true. This process is, however, different from one that results from the behavior of the flow of information. Also, the problem in considering the truth of the document proposition document was discussed in Chapter 1. And, nevertheless, the transformation is applied to the document, not to the knowledge set. This issue is not investigated further, because this thesis considers the document as the provider of information.

representation of the model of an IR system based on the flow of information, which is the main concern of this thesis.

2.4.4 Conclusion

Semantics-based frameworks can be used to develop a model of IR systems, but only if the objective is to model the *meaning* of information. Intensional Logic allows the incorporation of contexts (intensional expressions); however, it does not capture the flow of information as being the basis of a transformation. Montague Semantics, based on Intensional Logic, is an appropriate framework if a robust natural language process of the document is desired. It is concerned with the *meaning of the sentences* in the text document, not on the *information content* of the document. Finally, Data Semantics, which so far is the closest framework for developing a model based on a theory of information, has a different purpose; the representation of unstable propositions. This is not the same as the information that comes from the flow of information. In conclusion, a model based on the flow of information requires other frameworks than those proposed so far.

2.5 Information-based frameworks

This section concerns information-based frameworks. Three information-based frameworks, namely *Situation Theory* [Bar89, BE87, BE90, Dev91], *Channel Theory* [Bar91, Bar92, Sel90], and *Scott Domains* [Sco82] are presented. Situation Theory is the theory adopted in this thesis, so its description is detailed. Channel Theory is an extension of Situation Theory and has been used to develop a theoretical model of the flow of information for IR [Dre81]. Scott Domains enables the representation of structured information, an issue which has not yet been discussed.

2.5.1 Situation Theory

Situation Theory considers that, in a science of information, the most important entity should be information. Therefore, it proposes a mathematics of information and its flow.

The development of Situation Theory is based on the work of Drestke [Dre81] on information and flow. Situation Theory is described in detail because it is adopted as the basis for the modelling of an IR system. As Devlin [Dev91] says, the Situation Theory point of departure is

> "... the assumption that there is such a thing in the world as *information*".

An entity that is able to extract information from the world is called a *cognitive agent.* The acquisition of information is a process analogous to going from the *infinite and continuous to the finite and discrete*; Dretske refers to this process as the *analog to digital* representation of information. The extraction of information is done in two stages: first, the *perception* and, second, the *cognition.* The perception provides the information to the cognitive agent in analog form and the cognition corresponds to the conversion of analog to digital by that cognitive agent. Drestke calls this conversion a *digitalization.* The fundamental forms of information relevant to a cognitive agent are

The objects $a_1, \ldots, a_n$ have the property P
The objects $a_1, \ldots, a_n$ do not have the property P

These forms are modelled in Situation Theory by *infons* [Dev91]:

$$\ll P, a_1, \ldots, a_n; 1 \gg \quad and \quad \ll P, a_1, \ldots, a_n; 0 \gg$$

The objects '1' and '0' are called the *polarity* of the infons. The first infon is said to be *positive* and the second is said to be *negative*. The two infons are the *dual* of each other.

Suppose the information a cognitive agent obtains is that Mounia (myself) is working in her office. Situation Theory models this item of information by the infon $\ll Working, Mounia, Office; 1 \gg$. If a cognitive agent does not observe the fact that I am working in my office (for example, she sees explicitly that I am drinking coffee), the infon is $\ll Working, Mounia, Office; 0 \gg$. Nothing is said so far about the truth or falsity of these two infons; an infon is just the representation of an item of information. What makes an infon true is the *situation* from which the information represented by that infon is extracted. In addition, there might be several situations that make an infon true. Situation Theory models the notion of "make true' by the *support* relation, denoted $\models$. If σ is an infon and s a situation, then

$$s \models \sigma$$

This should be read as s supports σ, which means that s makes σ true. Situations refer to some part of the world, and are the place where the information resides. In Situation Theory, the denotation of a sentence is not its truth value, but a statement that the sentence (the information it expresses) holds in a particular situation. Situation Theory is explicit about the ontology of situations. It treats them as genuine entities in their own right, not merely as formal devices as in semantic-model approaches.

A situation provides information in analog form which is digitalized by a cognitive agent. Often, the amount of information provided is unlimited, but not all of it may be digitalized. The information that is extracted from a situation depends on the agent's *perception capability*, *focus of attention*, and *knowledge of the environment*. Consequently, a situation supports information modelled by way of positive and negative infons, but ignores many other non-related information items. This implies that situations are partial objects; they are partial representations of some parts of the world.

A situation seems like a perfect analogy to a document. This interpretation, as it is shown in this thesis, enables the explicit representation of the flow of information. To show this, further concepts need to be introduced.

Situation Theory introduces a level of abstraction among infons. For example, the three following infons:

$$\ll Working, Mounia, Office, 3pm; 1 \gg$$
$$\ll Working, Mounia, Library, 9am; 1 \gg$$
$$\ll Working, Mounia, Home, 11pm; 1 \gg$$

have the common information that Mounia is working. The differences are the place and time of the action. This commonality is represented in the theory by *types of situations*, or simply *types*. Here the corresponding type would be

$$\varphi = [\dot{s} | \dot{s} \models \ll Working, Mounia, \dot{p}, \dot{t}; 1 \gg]$$

This type classifies all the situations where a person referred to as Mounia is working at a given time and place. Here $\dot{s}$, $\dot{p}$ and $\dot{t}$ are parameters representing a situation, a place and a time, respectively. Any situation that supports the information that Mounia is working at a given time and place is of type φ. This is written as $s \models \varphi$ (in [Bar89], this is written $s : \varphi$)[29]. The instantiation of parameters, called *anchoring*, has to be concluded before affirming that $s \models \varphi$. For example, a situation where Mounia is working at the library at 6pm is of type φ since $\dot{p}$ can be anchored to "library" and $\dot{t}$ to "6pm". Parameters and anchoring are not essential to the understanding of the model, and are therefore not mentioned.

Now that types are defined, the flow of information can be modelled. Consider the two types:

$$\varphi = [\dot{s}|\dot{s} \models \ll Presence, smoke, \dot{p}, \dot{t}; 1 \gg]$$
$$\psi = [\dot{s}|\dot{s} \models \ll Presence, fire, \dot{p}', \dot{t}'; 1 \gg]$$

These types are not information-independent. The information they represent is related because the presence of smoke in a place means that there was, or is, a fire nearby. A flow of information indicates that a situation which supports φ (smoke) additionally carries the information that some other situation supports ψ (fire). The nature of the flow is formally represented by a *constraint*, with respect to the above types, denoted $\varphi \rightarrow \psi$[30]. Constraints, when applied to a situation, bring additional information about the same or other situations. Let the constraint $\varphi \rightarrow \psi$ be applied to a situation s. This is possible only if $s \models \varphi$, and it implies that there is a situation s' such that $s' \models \psi$. The situation s carries or contains information about s', which can be either the same as s or different because a situation can carry information about itself as well as another situation.

The constraint and the two situations affected by this constraint model the transformation of a document. That is, if a document is modelled by a situation d that supports φ, and $\varphi \rightarrow \psi$ is a constraint, then d is transformed into a situation d' that supports ψ. The transformation is explicitly determined by the constraint that makes the transformation. Unlike other frameworks, Situation Theory explicitly represents the flow of information that enables the transformation. The nature of the flow is determined by the constraints. Often, this was not represented in other frameworks, and when it was, it was not a correct representation of information containment (see Default Rules in section 2.3.3.1).

Constraints provide an accurate tool to represent thesaural or any semantic and pragmatic relationships. A document viewed as a situation supports the explicit information, and carries the implicit information which depends on the available constraints. This is a superior aspect of Situation Theory. The knowledge set K is therefore the set of constraints which are informative relationships.

Flows of information do not always materialize due to the unpredictable nature of situations, thus indicating that flows are often uncertain. In Situation Theory, an uncertain flow is modelled by

[29] A type can be constituted of several infons as the example below shows:

$$\left[\dot{s}|\dot{s} \models \left\{\begin{array}{l} \ll Working, Mounia; 1 \gg \\ \ll Writing, Mounia, Thesis; 1 \gg \\ \ll Topic, Wine; 0 \gg \end{array}\right\}\right]$$

The comma between the infons can be read as conjunction. The choice between comma and conjunction depends on the ontology adopted in the theory.

[30] Note that the interpretation of $\rightarrow$ in a constraint is different from that in Classical Logic.

a *conditional constraint* of the form $\varphi \rightarrow \psi | B$. This constraint highlights the fact that $\varphi \rightarrow \psi$ holds if some *background conditions* captured within B are met. A constraint that does not involve background conditions is *unconditional.* If the background conditions are satisfied, the corresponding flow arises. The flow of information depends on the satisfaction of these background conditions. The advantages of the background conditions in an IR system are that intensionality can be represented *and* uncertainty is already qualitatively captured. This was not the case with other frameworks.

In summary, Situation Theory represents a document by a situation, the information in the document by types, the modelling of the explicit information by the support relation, and the implicit information by the application of constraints. Both sequential and parallel transformations can be represented (to be demonstrated in Chapter 4). Situation Theory provides the necessary ontology to model the qualitative components of an IR system. Therefore, it is used in this thesis to model the qualitative aspects of an IR system. The model is summarized in the table below:

d	Situation	s
q	Type	φ
K	The set of constraints	$\{\alpha \rightarrow \beta \| B\}$
$d \Rightarrow q$	The situation supports the type	$s \models \varphi$
$d \rightarrow q$	The situation carries the type	$s \models \phi,\ \phi \rightarrow \varphi \mid B$

Table 2.7: The modelling of the quantitative components with Situation Theory

The use of Situation Theory provides an additional benefit because it has been used to develop a framework for natural language processing. This framework is called Situation Semantics [BP83, Coo]. It models the utterance of a sentence with three entities: the type that represents the *information content* of the sentence, the situation that the sentence *describes*, and the situation in which the sentence is *uttered.* All the components of a sentence are defined in terms of these three types of entities, which are combined to form the three entities of the sentence. A model based on Situation Theory can use Situation Semantics as the natural language process to identify the types that are supported by the situation modelling the document.

2.5.2 Channel Theory

A situation can contain information about another situation, thus there is a flow of information between the two situations. The nature of the flow is determined by constraints which are passive objects; they become active and give rise to a flow of information whenever they are related to pairs of situations. It is often the case that two situations are systematically related to each other. For example, a situation where smoke is perceived is related to a situation where a fire has occurred. A situation where a person hears the door bell ringing is related to a situation where a second person is at the door pressing the bell. Two kinds of relationships are involved: one that links *types*, and one that links *situations.* The concept of a *channel* is introduced by Barwise [Bar92] to express relationships of the second kind.

Let c be the channel that connects two situations s_1 and s_2. This is written $s_1 \overset{c}{\rightarrowtail} s_2$ and expresses the realization of the situation s_1 gives rise to a flow of information, which delivers some of the information supported by s_2 with respect to that channel c. A link between two situations can be

expressed by a channel, although it is not always possible to specify the nature of the flow that circulates in the channel. If the nature of the flow is known, it is characterized by constraints. Let $\varphi_1 \rightarrow \varphi_2$ be one of these constraints. In that case, the channel is said to support the constraint. This is written $c \models \varphi_1 \rightarrow \varphi_2$, which means that if $s_1 \models \varphi_1$, $s_1 \overset{c}{\mapsto} s_2$, and $\varphi_1 \rightarrow \varphi_2$, then $s_2 \models \varphi_2$.

Channel Theory defines the device that supports the flow of information. This device is formally described and its mathematical properties are specified (see [Bar92]). A special channel, the *unity channel*, is defined to represent the fact that flow of information gives information about the same situation. Two operations are defined on channels: ";", the sequential combination of channels and "||", the parallel combination of channels.

In a Channel Theory-based model of an IR system, a document is represented by a situation and a query by a type. Therefore, two types of knowledge exist: constraints and channels. The knowledge set is composed of constraints (as it is in the model based on Situation Theory) and of channels. A channel can be, for example, the link between synonymous information. If a transformation is modelled by a channel, the two operators ";" and "||" can model, respectively, the sequential transformation and the parallel transformation of documents. Determining the relevance of a document is to find the channel, together with its nature, that leads the situation modelling the document to one or several situations that contain the information being sought. A model based on Channel Theory is summarized in the following table:

d	Situation	s
q	Type	φ
K	The set of constraints The set of channels	$\{\alpha \rightarrow \beta \mid B\}$ $\{c\}$
$d \Rightarrow q$	The situation supports the type	$s \models \varphi$
$d \rightarrow q$	A channel leads to a situation that supports the type	$s \overset{c}{\mapsto} s',\ s' \models \varphi$

Table 2.8: The modelling of the quantitative components with Channel Theory

The advantage of Channel Theory is that a transformation is ontologically defined; that is, it is physically represented. This allows theoretical studies at two levels: the *transformation* and its *nature*. These two levels can be studied separately, eventually leading to a better understanding of the flow of information in IR.

A Channel Theory based theoretical model is proposed by Van Rijsbergen and Lalmas [vRL96]. However, in this thesis, Situation Theory is adopted for modelling the qualitative components of the IR system because an implementation of a model based on Situation Theory is sufficiently complex enough; the implementation of channels is not obvious.

2.5.3 Scott Domains

In section 2.2.1, it was mentioned that information can be organized into structures, thus leading to a structured representation of a document. In this thesis, the information that constitutes a structure is semantically related. For example, a structure can be viewed as the representation of a topic of a document. That is, the information which concerns the same topic are grouped together in the

same structure. This structured representation of the document can be used to define the specificity of a document (the concept of structures is formally defined in Chapter 5).

The representation of structures by the frameworks has not been discussed so far. The reason being that the concept used for modelling a document in these frameworks can be used to model a structure. The document is then modelled by a set of these concepts. For example, with Situation Theory, a set of situations can be used to model a document instead of one individual situation. Therefore, a model developed for the unstructured representation of a document can be used to develop a model for a structured representation of a document. This approach is followed in this thesis. Situation Theory is used to develop the model for an unstructured representation of a document, and the model is then generalized to account for a structured representation of the document.

The expression of the generalized model involves concepts that are not part of Situation Theory ontology, although they can be defined with this ontology (this is explained in Chapter 5). The missing concepts are taken from Scott Domains ontology [Sco82].

A Scott Domain is a set of elements which are described by properties. If a document is modelled by a domain, an element constitutes a structure. The benefit of this is that a model based on Situation Theory can be extended to include structured information because situations are comparable to elements. In Chapter 5 the analogy between an element and a situation is described in detail. A formal comparison can also be found in [Bar92].

Domains can be linked or related to each other by an *approximate mapping*, a function f, the properties of which are given in Chapter 5. If elements are thought of as situations, the function f can formally be defined based on the constraints in Situation Theory. If a document is represented by a domain, then the transformation of a document is formally represented by the function f. Moreover, approximate mappings can be combined to form the composition of approximate mappings; this models sequential flow of information.

The description of Scott Domains is given in Chapter 5 because the understanding of the use of Scott Domains requires the understanding of the approach adopted for modelling an unstructured representation of a document.

2.5.4 Conclusion

Three frameworks were discussed in this section. Situation Theory can represent all the qualitative components that define the IR model, and is adopted in this thesis for the qualitative modelling of an IR system. Channel Theory is an extension of Situation Theory, and has not been selected since its implementation is complex. A theoretical model based on Channel Theory was developed in [vRL96]. Scott Domains provides the terminology for the model which accounts for a structured representation of a document.

2.6 Conclusion

Three types of frameworks that can be used for the qualitative modelling of an IR system have been

described. They are truth-based frameworks, semantics-based frameworks and information-based frameworks. The weakness of truth-based frameworks lies mainly in two areas:

(i) the information contained in a document is represented by *true* propositions;

(ii) the flow of information is inappropriately represented because in most cases it is not possible to represent a transformation (the existence of a flow of information) and the nature of that transformation (the flow of information itself).

The disadvantage of semantic-based frameworks is that their aim is to represent the *meaning* of the information, not to model *information content.* Though determining the meaning of words, phrases, sentences, etc., that appear in a document is important, meaning on its own is still not sufficient to capture the information content of a document, for the latter often exceeds its meaning.

The use of Channel Theory for the modelling of the flow of information has been discussed in [vRL96]. The problem with an IR model based on Channel Theory is that the implementation of the model will be complex.

In conclusion, Situation Theory, an information-based theory, is used to model the qualitative features of an IR system because it can provide all the qualitative components of an IR model. That is, it satisfies all the requirements of a model based on the flow of information, which are: the document is a provider of information; its partial representation is modelled; intensionality is captured; the knowledge set is explicitly expressed; the uncertainty is captured (although qualitatively).

The majority of truth or semantics based frameworks use a syntax that has nothing to do with information content. The semantics are then attached to the syntax to model the information content. In Situation Theory, the semantics and the pragmatics are explicitly incorporated as *first-class citizens*, and a syntax is used so the semantics of information can be expressed.

Two models are proposed in this thesis, one that account for an *unstructured* representation of a document and one that accounts for a *structured* representation of a document. Although the expression of both models is based on Situation Theory, some terminology from Data Semantics and Scott Domains are borrowed, leading to a clear description of each model.

In the next chapter, frameworks that can provide the quantitative components of the model of an IR system are studied.

Chapter 3
Quantitative Theories for a Logic-Based Model of an Information Retrieval System

3.1 Introduction

A logic-based model of an IR system that follows the *Transformation Principle* involves both qualitative and quantitative components. The qualitative components and their modelling were discussed in Chapter 2. The present chapter is concerned with the *quantitative components and their modelling*. It is shown in this chapter that the frameworks that can be used to model these components are those used to model *uncertain inference* [KC93, Fro86]. Three of the most used frameworks representing uncertain inference for IR purpose are examined: the *probabilistic-based framework* [Goo50], *fuzzy logic* [Zad65], and *Dempster-Shafer's Theory of Evidence* [Dem68, Sha76]. This chapter demonstrates that the last framework models the quantitative components best.

3.2 The quantitative components

In the Transformation Principle, the *relevance* of a document d to a query q, represented as $d \rightarrow q$, given a knowledge set, depends on the *minimal transformation* of that document to a document d' which contains the information solicited in the query. In the previous chapter, only a qualitative evaluation of the relevance was considered; the document d is relevant to a query q if such a document d' exists. However, some documents are often more relevant to a query than other documents, so retrieved documents should be ranked according to their relevance to the query. Therefore, a *quantitative evaluation* of the relevance is necessary to express the extent to which a document is relevant to a query in addition to the fact that the document is relevant to the query.

This quantitative evaluation of the relevance can be represented by a *numerical value* $r(d \rightarrow q)$ such that the higher the value, the higher the relevance. $r(d \rightarrow q)$ is referred to as the *degree of relevance* of the document d to the query q. Its value ranges in the interval $[0, 1]$ because, as explained later in this section, its computation is based on the uncertainty of the transformation, and uncertainty is usually represented by values of the interval $[0, 1]$.

The evaluation of $r(d \rightarrow q)$ requires the definition of other numerical entities. These, together with $r(d \rightarrow q)$, constitute the quantitative components of an IR system based on the Transformation Principle. This section identifies and studies these components and their characteristics. As discussed previously, two models of an IR system based on the Transformation Principle are proposed in this thesis: one that accounts for an *unstructured representation* of a document, referred to as the *unstructured model*, and one that accounts for a *structured representation* of a document, referred to as the *structured model*. The identification of the quantitative components is discussed for each model.

3.2.1 Quantitative components of the unstructured model

The following example of a transformation of a document is used to understand the functionality of the quantitative components of the unstructured model:

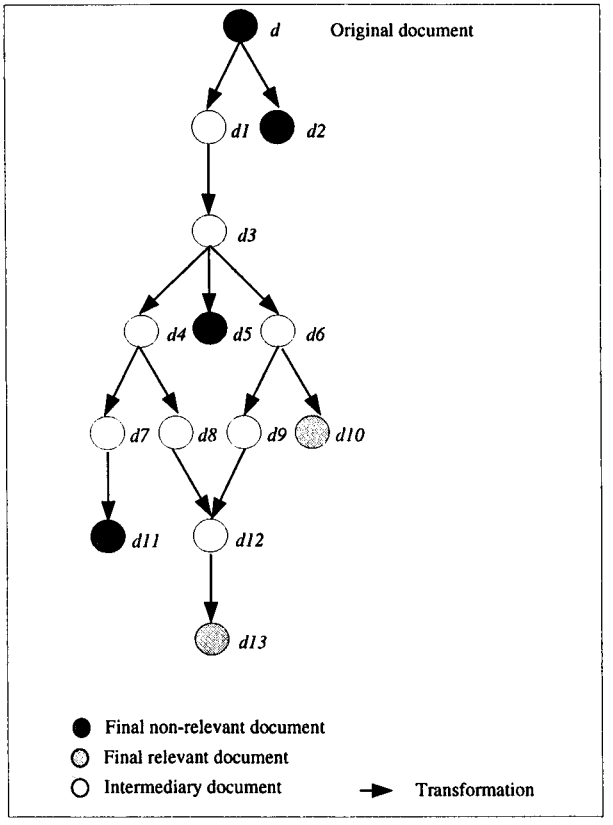


Figure 3.1: Example of the transformation of a document in the unstructured model

The original document d represents the information explicit in the document. The original document can be transformed into a document, which can itself be transformed to another document. These transformed documents contain the information implicit in the document[31]. The transformation is caused by the flow of information, which, in this thesis, is defined by relationships expressing the

[31] Note that the original and the transformed documents refer to one document, but different representations of it. For simplicity, an original document represents a document's explicit information content whereas a transformed document represents (some of) the document's implicit information content.

semantics between informations items (e.g., synonyms, generic terms, broader terms, etc.). These relationships are stored in the knowledge set. In this chapter, a relationship is denoted $p \rightsquigarrow p'$, where p and p' are information items. If the document d (explicitly) contains the information item p, a flow of information based on the relationship $p \rightsquigarrow p'$ transforms d into, for instance, the document d_1 which (explicitly) contains the information item p'. Then, d implicitly contains p'; that is, $d \rightarrow p'$.

A sequence of transformations may be necessary to obtain a document that contains the information requested by the query (e.g., d to d_{11}). There may be alternative ways to transform a document. The transformation of documents ceases when all the final documents (e.g., d_2, d_5, d_{10}, d_{11} and d_{13}) either cannot be further transformed (e.g., d_2, d_5 and d_{11}) or contain the information being sought by the query (e.g., d_{10} and d_{13}). Also, alternative transformations may lead the document d to the same transformed document (e.g., d_{12}). These transformations are referred to as parallel transformations.

As it will be explained in the next sections, the transformation of a document may involve *uncertainty*, which is *propagated* and *aggregated* along the sequential and parallel transformations of documents. These sections show that a numerical expression of this uncertainty, its propagation, and its aggregation can act as an indication of the degree of relevance of the document to the query. Moreover, the uncertainty of a transformation, the propagation and the aggregation of this uncertainty, and the degree of relevance constitute the quantitative components of the unstructured model.

In the next sections, Doc represents a set of documents, either original or transformed, Inf represents a set of information items, and K represents the knowledge set, defined as $K \subseteq Inf \times Inf$. The methods in which the documents and queries are represented, or the relationships of the knowledge set are determined, are left aside.

3.2.1.1 Uncertainty of the transformation

Consider the relationship[32] $function \rightsquigarrow purpose \in K$. The term "function" has several meanings in English, and the relationship $function \rightsquigarrow purpose$ holds only for one of these meanings. Suppose that the transformation of d into d_1 is based on this relationship. If it is not known which meaning of "function" is referred to in d, then the relationship is *uncertain* with respect to d. As a result, the information contained in d_1 (e.g., "purpose") is *uncertain* with respect to d. In other words, "purpose" is implicit and uncertain in d.

This example shows that a flow based on *uncertain relationships* leads to an *uncertain transformation*. The more uncertain the relationships, the more uncertain the transformation, and the more uncertain the information contained in the transformed document (with respect to the original document). In the unstructured model, the relevance of a document to a query depends on the existence of a transformation that leads to the information sought by the query. An uncertain transformation will then affect this relevance. Therefore, the uncertainty of a transformation constitutes a quantitative component of the unstructured model. Let

$$C : Doc \times Doc \rightarrow [0, 1]$$

[32] This relationship is taken from WordNet™ [Mil90], in which "function" and "purpose" are defined as synonyms (they have the same meaning) in some contexts. For sake of clarity, in the examples used in this chapter, information items correspond to terms.

be the numerical function representing the uncertainty of a transformation. If d is transformed into the document d_1, then $C(d, d_1)$ defines the uncertainty of the transformation. The lower the value of $C(d, d_1)$, the more uncertain the transformation.

The function C should reflect two facts. First, for a document d_i being transformed into a document d_j, the value of $C(d_i, d_j)$ depends on the uncertainty of the relationships used in the transformation. The properties of the relationships and their uncertainty are discussed in Chapter 4. In the present chapter, the function C is considered given; that is, the value of $C(d_i, d_j)$ is determined. Second, for any document d_i, $C(d_i, d_i) = 1$, because no uncertainty is involved in transforming a document onto itself.

3.2.1.2 Propagation of the uncertainty

A document that is transformed to another document can itself be the result of a transformation. In Figure 3.1, d is transformed into d_1 which is then transformed into d_3. Suppose that the relationship $purpose \rightsquigarrow goal \in K$ is uncertain with respect to d_1 and that the document d_1 is transformed into the document d_3 based on this relationship. The information contained in d_3 (e.g., "goal") is uncertain with respect to the information contained in d_1, which is itself uncertain with respect to the information contained in d. Therefore, the information becomes increasingly more uncertain with each transformed document. The uncertainty is said to be *propagated* along the sequence of transformations of the documents d, d_1 and d_3.

Consequently, the more transformations are necessary to obtain the information being sought by a query, the more uncertainty is propagated along these transformations, and the more uncertain is this information. A numerical formulation of this propagated uncertainty can be used to quantitatively express the relevance of the document to the query. Therefore, the propagation of the uncertainty constitutes a quantitative component of the unstructured model. Let

$$w : Doc \rightarrow [0, 1]$$

be the numerical function expressing the propagation of uncertainty. For a document d_i, $w(d_i)$ is the uncertainty associated with d_i, and represents the uncertainty of the information contained in d_i with respect to d. In other words, $w(d_i)$ is the uncertainty thus far propagated from the sequence of transformations that lead d to d_i. The higher the uncertainty, the lower the value of $w(d_i)$.

The computation of the value of $w(d_i)$ depends on whether d_i is an original or a transformed document. No uncertainty is attached to the original document, because the information contained in that document is certain. For example, in Figure 3.1,

$$w(d) = 1$$

If d_j is a transformed document, then let d_i be the document that is transformed into d_j. In that case, $w(d_j)$ depends on $w(d_i)$ (i.e., the uncertainty thus far propagated from the sequence of transformations that lead d to d_i), and $C(d_i, d_j)$ (i.e., the uncertainty associated with the transformation of d_i into d_j). Since information becomes more uncertain with each additional transformation, the values of $w(d_i)$ and $w(d_j)$ should be such that

$$w(d_j) \leq w(d_i)$$

3.2.1.3 Aggregation of the uncertainty

There may be parallel ways to transform a document into another document. In Figure 3.1, two transformations lead the original document d to the document d_{12}, where the transformations are, for example, based on different relationships of the knowledge set. This can be viewed as an accumulation of evidence towards the information contained in the document d_{12}; that is, the information contained in d_{12} should be less uncertain than if it was obtained by one transformation alone. Therefore, if $w^1(d_{12})$ and $w^2(d_{12})$ are the uncertainty[33] values attached to d_{12} with, respectively, the first transformation (via d_8) and the second transformation (via d_9), then the overall uncertainty attached to obtaining d_{12} from d should be a combination of the values of $w^1(d_{12})$ and $w^2(d_{12})$ such that its value is *at least as high* as $w^1(d_{12})$ and $w^2(d_{12})$. This combination corresponds to an *aggregation of the uncertainty*, and will affect the relevance of the document to the query. Therefore, the aggregation of the uncertainty is a quantitative component of the unstructured model.

In general, a document d can be transformed in $n > 0$ parallel ways into some document d_i, and that the uncertainty associated with each transformation is

$$w^j(d_i)$$

for $j = 1, n$. These values are defined by the propagation of the uncertainty. If $n = 1$, then $w^1(d_i) = w(d_i)$. Otherwise, the overall uncertainty attached to the document d_i with respect to d is defined as the aggregation of the uncertainty value of each of the parallel transformations. That is, the values $w^j(d_i)$ are aggregated into $w(d_i)$ such that

$$w(d_i) \geq w^j(d_i)$$

meaning that the uncertainty decreases with the number of aggregations, thus reflecting the accumulation of evidence towards the information contained in d_i.

3.2.1.4 Relevance degree

The relevance degree of a document to a query, manifestly, constitutes a quantitative component of the unstructured model. It is modelled by the numerical function

$$r : Doc \times Inf \rightarrow [0, 1]$$

The degree of relevance is expressed with respect to the original document. In Figure 3.1, $r(d \rightarrow q)$ denotes the extent to which the document d is relevant to a query q. The value of $r(d \rightarrow q)$ should capture two important facts. First, the relevance of the document should increase with the number of transformations of d that lead to the information requested by the query. Indeed, a high number of such transformations means that there exists *many alternative transformations* of the document resulting into documents containing information that concerns the query. This should indicate a higher relevance of the document to the query. For example, in Figure 3.1, the relevance of the document should be higher than if only d_{10} or d_{13} was obtained alone, where obtaining the two transformed documents d_{10} and d_{13} could be due to the use of different semantic relationships.

[33] The subscripts are used to differentiate the two transformations. If a is the uncertainty associated with the first transformation and b is the uncertainty associated with the second transformation, then $w^1(d_{12}) = a$ and $w^2(d_{12}) = b$.

Second, the more uncertain a transformation, the less relevant the document is with respect to that transformation. In Figure 3.1, the relevance of the document should increase if the uncertainty attached to the transformation of d into, for instance, d_{10} was decreasing.

These two facts can be captured by defining the value of $r(d \rightarrow q)$ as the aggregation of the uncertainty of those transformed documents which contain the query q. That is, the value of $r(d \rightarrow q)$ is defined as the aggregation of the values of $w(d')$ such that d' is a transformation of d and d' contains q. If there is only one such document d', then

$$r(d \rightarrow q) = w(d')$$

Otherwise, $r(d \rightarrow q)$ is such that

$$r(d \rightarrow q) \geq w(d')$$

for all those concerned d'. In Figure 3.1, d_{10} and d_{13} are the two transformed documents that contain the information sought by the query. Then, $r(d \rightarrow q)$ is the aggregation of $w(d_{10})$ and $w(d_{13})$. The special case is when d already contains the information being sought by the query. In that case,

$$r(d \rightarrow q) = w(d) = 1$$

An aspect which has not been discussed so far is the *minimality* of the transformation. A minimal transformation is a sequence of transformations in which all the transformations are necessary; that is, the transformations are based on relationships that are *essential* to obtain the information being sought by the query. An example of a non-minimal transformation is given below:

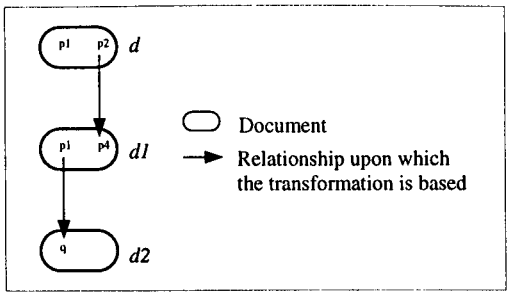


Figure 3.2: Example of a non-minimal transformation

The transformation is non-minimal with respect to q because the transformation of d into d_1 is not necessary. The reason being that the relationship $p_2 \rightsquigarrow p_4$ is non-essential[34] to obtain the information item q. An example of a minimal transformation is given in Figure 3.3. There, the transformation is minimal with respect to q because both relationships $p_1 \rightsquigarrow p_3$ and $p_3 \rightsquigarrow q$ are required to obtain the document d_2 which contains q.

[34] In this example, the use of the relationship $p_1 \rightsquigarrow q$ to transform d_2 is assumed independent to the fact that d_2 contains p_4.

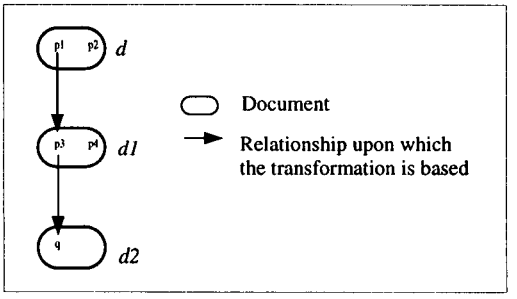


Figure 3.3: Example of a minimal transformation

The minimality of the transformation of a document is a qualitative aspect as opposed to a quantitative aspect. In this chapter, the determination of a minimal transformation is not an issue; that is, only transformations involving necessary relationships are considered. In the model developed by Nie [Nie90], which also involves the idea of transformation, minimality was captured by keeping the most certain transformation, and hence was a quantitative aspect. Transformations which involve unnecessary relationships were more uncertain, and so were always ignored.

3.2.2 Quantitative components of the structured model

As discussed in Chapter 1, a document's information content should be represented as *a set of structures*. In this thesis, a structure contains semantically related information items and can be viewed as denoting a topic (other types of structures are discussed in Chapter 8). Consider the following document (for simplicity, the document is represented as a set of terms)[35]:

$$d = \{\text{rose, Sun, giraffe, tulip, table, dog, Macintosh, elephant}\}$$

Four structures can be identified in this document denoting the topics "flower", "animal", "computer", and "furniture". An illustration of a structured representation of this document is given in the following schema:

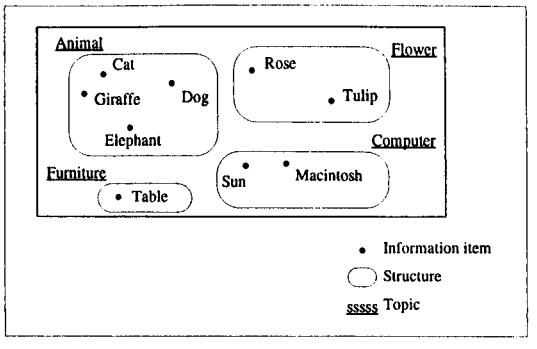


Figure 3.4: Example of a structured representation of a document

Some structures may be more significant in a document than others because they constitute a more prominent part of the document information content than other structures. For example, the structure denoted by the topic "animal" in Figure 3.4 can be considered more significant that the structure denoted by the topic "furniture" because the former structure contains more information

[35] Here, Sun refers to a computer brand.

than the latter. In that case, the document is more relevant (being more specific) to a query about "animal" than to a query about "furniture".

It is necessary to represent that some structures are more significant than others. The most intuitive approach is that a *weight* (a numerical value) is assigned to each structure to represent its significance. The higher the weight, the more significant the structure. The representation of these weights constitutes a quantitative component which is only relevant to the structured model. The formal expression of this component is discussed later in this section since it requires discussion on the transformation of a document defined as a set of structures.

The transformation of a document in the structured model is also due to the flow of information, which is determined by relationships of the knowledge set. However, the transformation of a document must take into account the structures that constitute that document. In this thesis, the transformation of a document is defined in terms of the transformations of these structures. This is illustrated in the following figure:

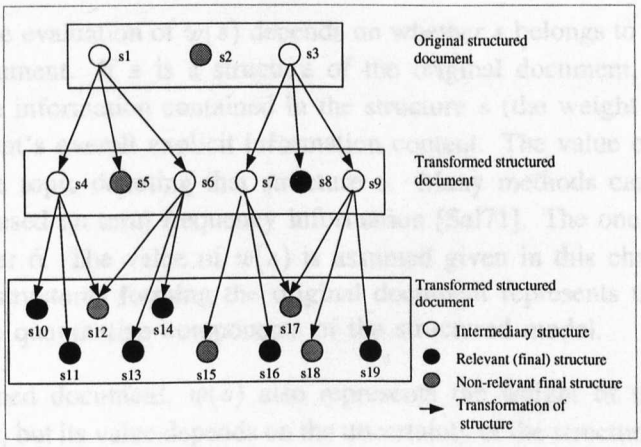


Figure 3.5: Example of the transformation of a document in the structured representation

The transformation process is identical to that in the unstructured model, except that it applies to structures and it starts from a set of structures, as opposed to one entity (the original document). It will be formally shown in Chapter 5 that this approach allows a better representation of a document's information content. Let us assume for the moment that this approach is intuitive, but a fuller description of the underlying features will be given in Chapter 5.

Both structures and documents contain information, with the difference that the information contained in a structure is semantically related. The transformation of a structure is the same as that of a document; it is based on relationships which may be uncertain, and the uncertainty is propagated along the sequential transformations of structures (instead of documents). The uncertainty is also aggregated along the parallel transformations. In the unstructured model, parallel transformations mean alternative ways to obtain a document from the original document. In the structured model, the transformation process starts from a set of structures, so parallel transformations include two cases:

(i) alternative ways to obtain a structure from a given structure, or

(ii) a structure resulting from several transformations which originate from distinct structures.

However both cases mean that there are different ways to obtain a structure. Therefore, the aggregation of the uncertainty is the same in both cases. The uncertainty values associated to the different transformation are computed and then aggregated as described in section 3.2.1.3.

Therefore, the uncertainty of a transformation, the propagation of uncertainty and the aggregation of the uncertainty also constitute quantitative components of the structured model. They can be captured by the same functions C and w used in the unstructured model. These functions are however redefined with respect to structures:

$$C : Struct \times Struct \rightarrow [0,1]$$
$$w : Struct \rightarrow [0,1]$$

where $Struct$ is a set of structures. The function C models the uncertainty attached to the transformations of structures. If a structure s is transformed into a structure s', then $C(s,s')$ expresses the uncertainty associated to the transformation. The function C has the same properties as discussed in section 3.2.1.1, and is also assumed given.

Given a structure s, the evaluation of $w(s)$ depends on whether s belongs to the original document or a transformed document. If s is a structure of the original document, then $w(s)$ represents the significance of the information contained in the structure s (the weight of the structure) with respect to the document's overall explicit information content. The value of $w(s)$ increases with the prominence of the topic denoting that structure s. Many methods can be used to compute $w(s)$, such as those based on term frequency information [Sal71]. The one adopted in this thesis is described in Chapter 6. The value of $w(s)$ is assumed given in this chapter. The function w when applied to the structures forming the original document represents the weights previously mentioned, one of the quantitative components of the structured model.

If s is in a transformed document, $w(s)$ also represents the weight of the structure s in that transformed document, but its value depends on the uncertainty of the structures that are transformed into s and the uncertainty of these transformations. That is, when applied to structures of the transformed document, the function w represents the propagation and the aggregation of uncertainty in the structured model. The value of $w(s)$ satisfies the same properties described in sections 3.2.1.2 and 3.2.1.3; that is, uncertainty increases when propagated, and decreases when aggregated.

Finally, the relevance of a document to a query is defined as the aggregation of the uncertainty of those transformed structures (instead of documents) that contain the information sought. The relevance is also expressed by the function r

$$r : Doc \times Inf \rightarrow [0,1]$$

where $Doc = 2^{Struct}$ (the power of $Struct$) because a document is represented as a set of structures. In Figure 3.5, the structures representing the original document are s_1, s_2, s_3. The structures that contain the information being sought by the query are s_8, s_{10}, s_{11}, s_{13}, s_{14}, s_{17} and s_{18}. The relevance of the document to the query will be the aggregation of $w(s_8)$, $w(s_{10})$, $w(s_{11})$, $w(s_{13})$, $w(s_{14})$, $w(s_{17})$ and $w(s_{18})$. The computation of these values is as discussed in sections 3.2.1.2 and 3.2.1.3, but applied to structures.

In summary, the same functions can be used to express the quantitative components used in the structured and the unstructured models. Moreover, the unstructured model is a special case of

the structured model; one in which a single structure (i.e., d) is involved to represent the original document and the weight reflecting its significance is equal to 1 (i.e., $w(d) = 1$). Therefore, in the remainder of this chapter, only the representation of the quantitative components of the structured model is discussed. The structured model will be referred to as *the model* unless otherwise specified. Its quantitative components are summarized in the following table:

Uncertainty of a transformation	$C : Struct \times Struct \rightarrow [0,1]$
Significance of information	$w : Struct \rightarrow [0,1]$
Propagation of uncertainty	
Aggregation of uncertainty	
Relevance degree	$r : 2^{Struct} \times Inf \rightarrow [0,1]$

Table 3.1: The quantitative components of a logic-based model of an IR system based on the Transformation Principle

3.2.3 Remainder of the chapter

A logic-based model of an IR system based on the Transformation Principle involves quantitative components, which are

(i) the *significance of the information*,
(ii) the *uncertainty of a transformation*,
(iii) the *propagation of the uncertainty*,
(iv) the *aggregation of the uncertainty*, and
(v) the *numerical expression of the relevance degree.*

The process defined by these quantitative components can be compared to an *uncertain inference process* [KC93, Saf87] of the following form:

$$\frac{\begin{array}{l}\text{Uncertain fact:} \quad p \text{ is true with uncertainty } a \\ \text{Uncertain rule:} \quad if\ p\ then\ q \text{ is true with uncertainty } b\end{array}}{\text{Uncertain fact:} \quad q \text{ is true with uncertainty } c}$$

The uncertainty of the inferred fact q (the value c) is defined *in terms* of the uncertainty of the fact p (the value a) and the uncertainty of the rule $if\ p\ then\ q$ (the value b). By analogy, the uncertainty of a structure is defined in terms of the uncertainty of the structure of which it is a transformation and the uncertainty of the transformation. A second feature of an uncertain inference process is that the fact q can be used to infer other uncertain facts if there exist rules of the form $if\ q\ then\ r$. The uncertainty is *propagated* along this inference; that is, the inference takes into account that the uncertainty of the fact q is now c. Likewise, uncertainty is propagated along a sequential transformations of structures. A third feature is that several inferences may yield the same fact q. The uncertainty values that result from each these inferences must be *aggregated* into one value that expresses the overall uncertainty attached to the inferred fact q. Similarly, uncertainty is aggregated when a structure is the result of several transformations.

The representation of the uncertainty in an uncertain inference process depends on the frameworks used to model this uncertainty. The use of one framework instead of another usually depends on the properties attached to the inference process [Saf87]. These frameworks are often referred to as

a *Theory of Uncertainty*. Three Theories of Uncertainty are examined in this chapter to determine the one that represents best the quantitative components of a logic-based model of an IR system.

These theories are *probabilistic-based ones* [Goo50], *Fuzzy Logic* [Zad65] and *Dempster-Shafer's Theory of Evidence* [Dem68, Sha76]. These theories have already been used to develop models of IR systems (see for example [BS75, Rob77, CGD92, Fuh92, Mar92, vR92]), but most of these models are not logic-based, and hence not mentioned. The discussion in this chapter centers around the fact that the quantitative components are defined for a logic-based model of an IR system based on the Transformation Principle.

3.2.3.1 Test cases

To understand whether the different Theories of Uncertainty examined in this chapter can or cannot model the quantitative components, test cases are used. There are five test cases, one for each of the quantitative components. The first two test cases concern the expression of the significance of information and the uncertainty of a transformation. The entities involved in these test cases are illustrated in the following figure:

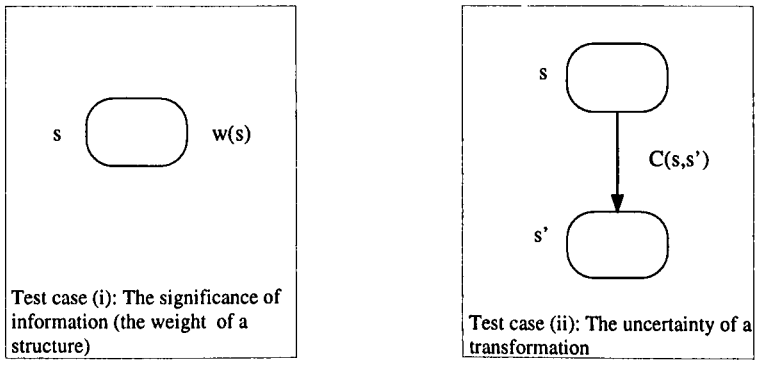


Figure 3.6: The entities involved in test cases (i) and (ii)

The weights of the structures of the original documents are supposed determined. That is, for a given structure s of the original document, $w(s)$ must be represented, but its value is computed elsewhere. The same applies for the representation of the uncertainty of a transformation.

The three other test cases concern the propagation and the aggregation of the uncertainty, and the expression of the relevance. The entities involved in these test cases are shown in Figure 3.7. To satisfy the test case (iii), a Theory of Uncertainty must provide an appropriate representation of $w(s)$ and $C(s, s')$, and then compute $w(s')$ in terms of $w(s)$ and $C(s, s')$ such that $w(s) \geq w(s')$, meaning that uncertainty increases when propagated. To satisfy the test case (iv), the Theory of Uncertainty must provide an appropriate representation of $w(s')$ in terms of $w^1(s')$ and $w^2(s')$, where s' has been obtained via two different transformations, and should reflect an accumulation of evidence towards the information contained in s'. To satisfy the test case (v), the Theory of Uncertainty must provide an appropriate representation of $r(d \rightarrow q)$ in terms of $w(s_1)$ and $w(s_2)$, where both s_1 and s_2 contain q, and obtaining two transformed documents containing information that concerns the query instead of one should indicate an increase of the relevance of the document d to the query q.

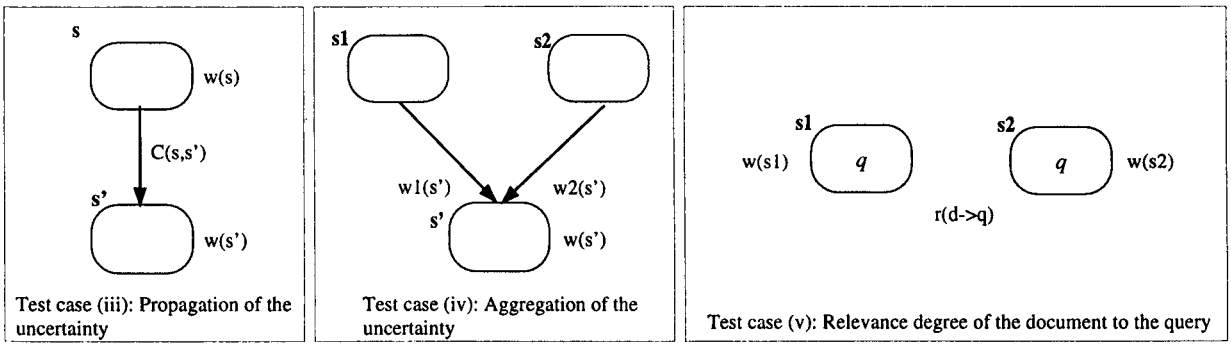


Figure 3.7: The components involved in test cases (iii), (iv) and (v)

The five test cases are summarized in table 3.2 below.

Cases	To represent
(i)	$w(s)$ for s in the original document
(ii)	$C(s, s')$ for s being transformed into s'

Cases	Given		To compute
(iii)	$C(s, s') = a$	$w(s) = b$	$w(s')$ in terms of a and b such that $w(s') \leq b$
(iv)	$w^1(s') = a$	$w^2(s') = b$	$w(s')$ in terms of a and b such that $w(s') \geq a, b$
(v)	$w(s_1) = a$	$w(s_2) = b$	$r(d \rightarrow q)$ in terms of a and b such that $r(d \rightarrow q) \geq a, b$

Table 3.2: The five test cases

The discussion will usually concentrate on the satisfaction of test cases (iii), (iv) and (v) because they often constrain the representation of the entities involved in the test cases (i) and (ii). Also, only parts of the table may be used at times since one of the quantitative components may not be represented. In that case, the representation of the other components is not examined (e.g., the study of the representation of the aggregation of the uncertainty is pointless if the propagation of the uncertainty cannot be represented correctly).

3.3 Probabilistic-based frameworks

Probabilistic-based frameworks are based on Probability Theory [Goo50]. Three probabilistic-based frameworks are examined in this section: *Probability Theory* itself [Goo50], *Bayesian Inference* [Pea88, Nea90] and *Imaging* [Lew73]. In these frameworks, the probability of the implication is used to model the uncertainty of the inference. In the first two frameworks, the uncertainty of the implication is based on *conditional probability* [Goo50], whereas in the third framework, it is based on *conditional logic* [Sta84, Nut80].

3.3.1 Probability Theory

A *probability function* $P : U \rightarrow [0,1]$ formalizes the phenomena that some propositions of a *universe of discourse* U are *more probable* than other propositions. For a proposition p, $P(p)$ is the probability that p is true. The properties of a probability function are

(i) $P(p \vee q) = P(p) + P(q) - P(p \wedge q)$

(ii) $P(p) + P(\neg p) = 1$

(ii) is often referred to as the *coherence rule* [Saf87]. Probability Theory also defines a *conditional probability*, denoted $P(q|p)$, which states the probability that a proposition q is true given that a proposition p is true. The conditional probability $P(q|p)$ is defined as follows (for $P(p) \neq 0$):

$$P(q|p) = \frac{P(p \wedge q)}{P(p)}$$

An obvious use of a conditional probability is to model the propagation of the uncertainty [Pea88, Nea90]. However, as it will be demonstrated in this section, no interpretation of the conditional probability leads to an appropriate representation of this quantitative component.

Wong and Yao [WY91] use conditional probabilities to define a probabilistic logic-based model of an IR system. The relevance[36] of a document d to a query q, $r(d \rightarrow q)$, is evaluated as a conditional probability, that is

$$r(d \rightarrow q) = P(q|d) = \frac{P(d \wedge q)}{P(d)}$$

The probabilities are defined in an universe of discourse U which is defined as a set of terms (or eventually structures, although Wong and Yao do not mention this case). d and q are defined as sets of terms. $P(d)$ is interpreted as the degree to which U is covered by the terms contained in d and $P(d \wedge q)$ is interpreted as the degree to which U is covered by terms common to both d and q. The values of $P(d)$ and $P(d \wedge q)$ are based on weights associated to the terms in d and q which are estimated based on term frequency information. Wong and Yao discuss different formulations of $P(d \wedge q)$, which depend on the properties assumed between terms, and which lead to different IR models[37]. However, all of these formulations take only into account the information that is explicit in the document. That is, although it may be possible to define $P(d)$ and $P(d \wedge q)$ based on structures, none of these formulations account for a representation of a transformation yielding the implicit information of a document. Therefore, the probabilistic model defined by Wong and Yao is inadequate to model the quantitative components since the transformation of a document cannot be represented.

A more productive use of a conditional probability to represent the propagation of uncertainty is to define the universe of discourse as a set of structures. Let the structure s be transformed into the structure s'. The propagation of the uncertainty may be expressed as follows:

$$P(s'|s) = \frac{P(s \wedge s')}{P(s)}$$

[36] Other formulations of the relevance were also defined by Wong and Yao such as $r(q \rightarrow d)$. $r(d \rightarrow q)$ is a recall-oriented measure of the relevance whereas $r(q \rightarrow d)$ is a precision-oriented one.

[37] For example, Wong and Yao show that their probabilistic model covers the Boolean Model, the Vector Space Model [SM80, Sal71], and the Fuzzy Model [Boo85]. Other formulations were also discussed in [CCLvR96], which lead to different types of probabilistic IR models (see also [Fuh92]).

$P(s'|s)$ and $P(s)$ represent the uncertainty attached to s' and s, respectively. As a result, $P(s \wedge s')$ is the uncertainty attached to the transformation. This different probabilities match the test case (iii) as follows:

$C(s, s')$	$w(s)$	$w(s')$
$P(s \wedge s') = a$	$P(s) = b$	$P(s'\|s) = \frac{a}{b}$

Table 3.3: The representation of the propagation of uncertainty in Probability Theory: first attempt

However, this representation is both counter-intuitive and incorrect. It is counter-intuitive because $P(s \wedge s')$ reflects the uncertainty attached to the information common to the two structures s and s', and not the uncertainty of the transformation of the structure s into the structure s'. Indeed,

$$P(s \wedge s') = P(s' \wedge s)$$

meaning that the uncertainty attached to the transformation of s into s' is the same as that of the transformation of s' into s, which is, often, not a valid assumption. Moreover, this representation of the uncertainty of a transformation is incorrect because it does not capture the fact that the uncertainty associated to a transformation of a structure s onto itself is equal to 1. Indeed, this uncertainty would be represented as

$$P(s \wedge s) = P(s)$$

which is not necessarily equal to 1.

A more appropriate use of a conditional probability to model the propagation of uncertainty is due to the following rule which derives from the definition of a conditional probability (the proof can be found in [Par94]):

$$\text{if } P(q|p) = a \text{ and } P(p) = b \text{ then } a * b \leq P(q) \leq 1 - b + a * b$$

Applied to structures, this different probabilities match the test case (iii) as follows:

$C(s, s')$	$w(s)$	$w(s')$
$P(s'\|s) = a$	$P(s) = b$	$a * b \leq P(s') \leq 1 - b + a * b$

Table 3.4: The representation of the propagation of uncertainty in Probability Theory: second attempt

Here, $P(s'|s)$ represents the uncertainty of the transformation of s into s', which is more intuitive than the above $P(s \wedge s')$. $P(s)$ and $P(s')$ are the uncertainty values attached to s and s', respectively. One problem with this representation of the propagation of uncertainty is that the values of the uncertainty become increasingly more imprecise along the transformations because only intervals are provided. Indeed, if the structure s' is itself transformed into a structure s'', $P(s'')$ becomes expressed in terms of an interval. A second problem is that the uncertainty values attached to the transformations are intervals. As a result, the aggregation of uncertainty becomes defined as the aggregation of intervals, which, if not correctly formulized, may lead to incorrect results. Also, the relevance of a document to a query becomes defined as the aggregation of

intervals and will be itself an interval. The comparison of intervals to rank documents according to their relevance degree is not obvious to express.

In conclusion, Probability Theory cannot adequately model the quantitative components of the model because either no appropriate interpretation or representation of the different probabilities can be found to model the propagation of the uncertainty, or the uncertainty (hence, the relevance) is expressed as an interval.

3.3.2 Bayesian methods

Bayesian methods are used to model inference within a probabilistic-based framework. They are based on *Bayes' Theorem* [Pea88, KC93], which has many formulations. One of the most commonly used formulation is the following:

$$P(h|e) = \frac{P(h) * P(e|h)}{P(e)}$$

h is the *hypothesis*, and e is the piece of *evidence* that is observed. $P(h|e)$ is the probability that h is true given the evidence e. Although the same function P is used, different functions are involved. $P(h|e)$ is the *posterior* probability of h and could be noted $P_e(h)$, whereas $P(h)$ is its *prior* probability. The prior probabilities are assigned to events and can be revised in light of new evidence, thus leading to the posterior probabilities. One advantage with the use of a Bayesian method with respect to Probability Theory is that precise values of uncertainty are delivered.

In IR, a Bayesian method is often used in tandem with an *inference network*, which is a directed acyclic graph [Fau78] constituted of nodes linked by arcs. Nodes represent IR entities such as documents, concepts, information items, query, etc. Arcs represent probabilities dependencies between nodes. Prior probabilities are assigned to all the nodes and conditional probabilities are assigned to all the arcs. The posterior probabilities of the different nodes are based on the probabilities of their *active* parent nodes, and are computed according to Bayes' theorem. Figure 3.8 shows an example of a Bayesian inference network for IR.

The root nodes represent the document collection. The leaf node represents the information need expressed by a query. The t_i nodes represent the information items (or whatever is used to index a document) in the document, and the c_j nodes are the information items expressing the information need. The evaluation of the relevance of a document, for instance d_1, to the query is defined as $P(q|d_1)$. d_1 is the piece of observed evidence, which activates the nodes t_1 and t_2 (d_1 is the active parent of t_1 and t_2) which then activate c_1, which then activates q. The probabilities along the activated nodes are calculated according to Bayes' theorem. Different IR models based on Bayesian inference networks have been implemented [Tur90, Sav92], but none of them is relevant to the notion of transformation[38].

[38] An interesting implementation was proposed by Croft and Turtle [Tur90, Cro92], in which multiple representations of document and query were used, which allowed the computation of the relevance based on different strategies.

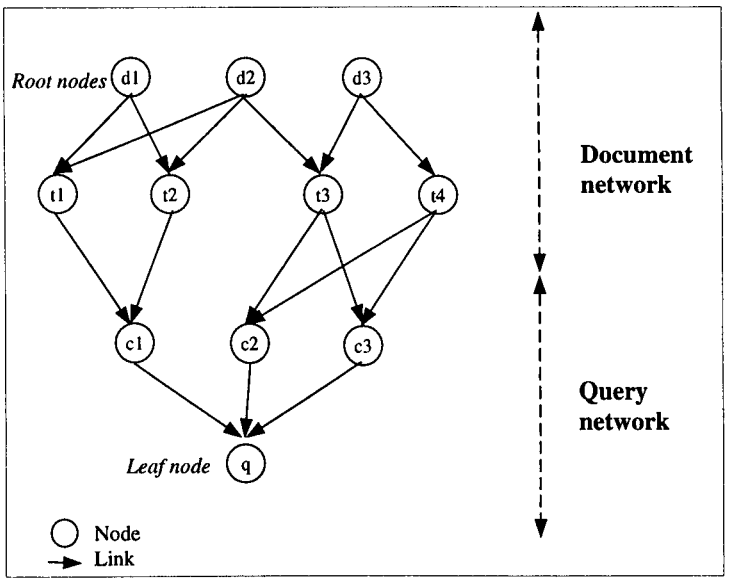


Figure 3.8: An example of an inference network in IR

The use of a Bayesian inference network to model the quantitative components of the model proposed in this thesis is discussed in the remainder of this section. To represent the propagation and the aggregation of the uncertainty, the network must encompass structures. A more intuitive approach is illustrated in the following figure:

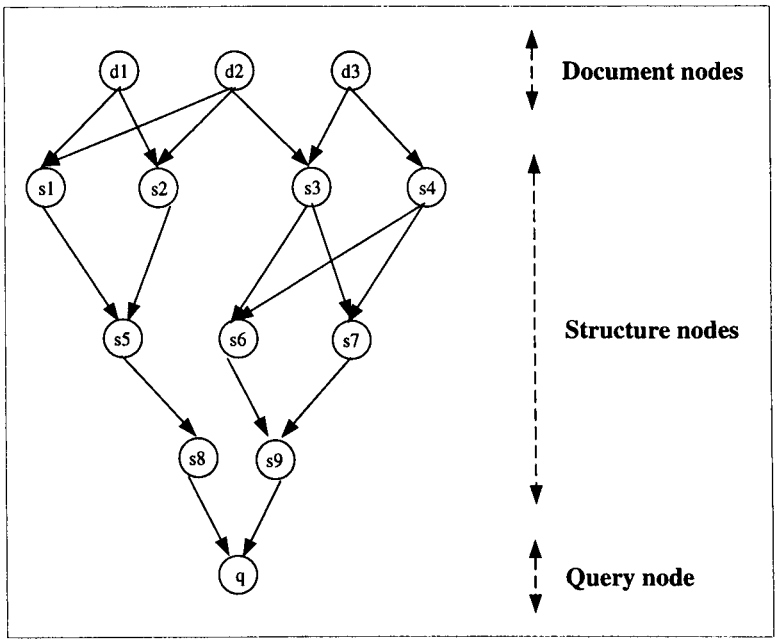


Figure 3.9: A Bayesian inference network for a logic-based model of an IR system

A document node is linked to a structure node if that structure exists in that document. A structure node is linked to another structure node if the former is transformed into the later. Two structures are linked to the same structure if they can be both transformed into that same structure. The query node portrays the information need (the information items that constitute the query). A structure

node is linked to the query node if that structure can be transformed to a structure that contains information concerning the query.

An obvious representation of the propagation of the uncertainty which match the test case (iii) is given in the following table:

$C(s,s')$	$w(s)$	$w(s')$
$P(s\|s') = a$	$P(s) = b$	$P(s'\|s) = P(s') * a/b$

Table 3.5: Representation of the propagation of the uncertainty in a Bayesian inference network

$P(s|s')$ represents the uncertainty attached to the transformation of the structure s to the structure s'. $P(s)$ is the uncertainty attached to the structure s, which is also computed according to Bayes' theorem. The uncertainty attached to s' is $P(s'|s)$, the computation of which requires an extra value, $P(s')$. This value is the prior probability of s'. Following most of IR models based on Bayes' theorem[39], $P(s')$ can be interpreted as the distribution of the structure s' in the document collection, which can be estimated, for example, by the inverse document frequency [Sal71] or the term discrimination value [SWY76].

The propagation of the uncertainty in a Bayesian inference network and the one in the model developed in this thesis are different phenomena. In the model developed in this thesis, the probability of a structure consists of the uncertainty attached to the obtaining of that structure via a transformation. No prior uncertainty is attached to this structure. In the Bayesian framework, a structure has an initial probability (the distribution), which may be altered in light of new evidence. The new evidence is a particular document, and the uncertainty is propagated along the network with respect to that new evidence. As a result, the representation of the propagation of uncertainty by Bayes' theorem does not necessarily satisfy the property that uncertainty increases when propagated (i.e., $P(s'|s) \leq P(s)$). Take the case of propositions. Given two propositions p and q, the analogous inequality is $P(q|p) \leq P(p)$. Let the probability P be an uniform distribution on the numbers $\{1,2,3,4,5,6,7,8,9,10\}$ (i.e., each occurs exactly one tenth of the time). Let p = greater than 5 and q = greater than 3. Then $P(q|p) = 1$ and $P(p) = 0.5$, which contradicts the previous inequality.

There are other problems with the use of a Bayesian network to model the propagation of the uncertainty. First, the interpretations of the different probability functions are not always evident. For example, in the formulation of the propagation of the uncertainty, $P(s|s')$ is the uncertainty attached to the transformation of the structure s into the structure s'. A more intuitive probability would be $P(s'|s)$, which is already used to represent the uncertainty associated to s'. Second, the different prior probabilities must be estimated (e.g., $P(s')$). The estimation of these probabilities is not easy since a Bayesian framework is probability-based, meaning that the coherence rule applies. The satisfaction of this rule by the different estimations must be ensured.

In summary, a Bayesian method is not used to model the quantitative components in the model developed in this thesis because it cannot capture the propagation of the uncertainty as defined in the model proposed in this thesis.

[39] Distributions are often used in probabilistic IR models [vR79, Fuh92].

3.3.3 Imaging

In Imaging [Lew73, Lew76, HSP81], the formulation of the probability of an implication is based on Conditional Logic [Nut80, HSP81], and not on conditional probability. In Conditional Logic, the evaluation of an implication is based on the possible-worlds semantics [Kri63]. Given a set of possible worlds W, the truth value of the implication $p \rightarrow q$ in a world w of W depends on two cases. If p is true in w, then $p \rightarrow q$ is true (false) in that world if q is true (false) in that world. If p is not true in w, then the implication is evaluated in the world, for instance, w' that differs minimally from w and in which p is true. $p \rightarrow q$ is true (false) in w just in case q is true (false) in w'[40].

Formally, the truth value of a proposition p in a world w is represented as

$$w(p) = \begin{cases} 1 & \text{if } p \text{ is true in } w \\ 0 & \text{otherwise} \end{cases}$$

The truth value of the implication $p \rightarrow q$ in that world w is defined as

$$w(p \rightarrow q) = w_p(q)$$

where w_p is the world that is reached by the least drastic revision of the facts (true propositions) of w that makes p true. The world w_p is called the closest *p−world* of w.

Imaging defines the probability on an implication based on this notion of closest world. A probability function P is first defined as a distribution on the set of worlds W such that

$$\sum_{w \in W} P(w) = 1$$

This distribution is extended to propositions as follows:

$$P(p) = \sum_{w \in W} P(w) * w(p)$$

The probability $P(p)$ is an overall distribution of the proposition p in W. The *p−image* of P, denoted P_p, is defined as the following probability function

$$P_p(w') = \sum_{w \in W} P(w) * \begin{cases} 1 & \text{if } w_p = w' \\ 0 & \text{otherwise} \end{cases}$$

The original probability of each world w is shifted to w_p, the closest $p - world$ to w. $P_p(w')$ is the summation of the probability of any world w, the $p-$closest world of which is w'. The probability P_p is extended to propositions, as was the original probability function P, and it can be proven that (see [Lew73] for proof)

$$P(p \rightarrow q) = P_p(q)$$

[40] Conditional Logic is particularly appropriate for the modelling of *counterfactuals* [Lew73]. These are sentences, the antecedents of which are false in the world in which the sentences are evaluated. An example is the utterance of the sentence "If it were less cold, I could have gone for a walk" in a situation (actual world) where it is cold.

The imaging process seems compatible with the notion of minimal transformation defined in this thesis. If d represents the document and q models the query, then

$$r(d \rightarrow q) = P_d(q)$$

thus leading to a numerical expression of the relevance of the document to the query. However, as shown in the remainder in this section, no adequate interpretation of the different concepts defined in Imaging caters to an appropriate representation of a transformation.

Suppose that both the document and the query are propositions; consequently, worlds model different stages of the knowledge set. This approach was followed by Nie & al [NLB96], who claims that the relevance of a document to a query does not only depend on whether the document satisfies the information need expressed by that query, but also on other aspects such as users' knowledge, background, intention, etc. The knowledge set then models users' knowledge and the worlds represent possible states of knowledge that can be held by users. The document d is true in a world w if the document is compatible with the state of the knowledge associated with this world. w_d is the closest world to w such that d is true in this world. The imaging process is illustrated in the following figure:

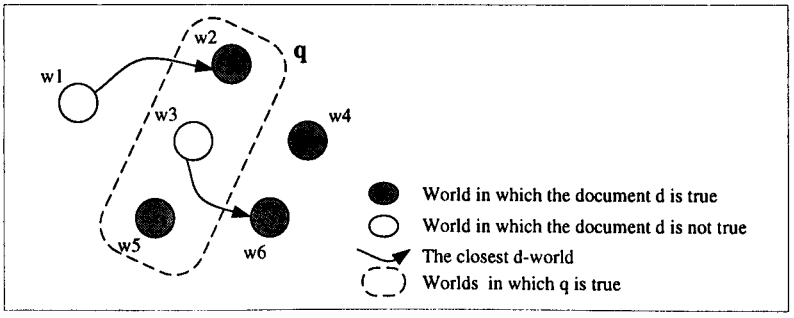


Figure 3.10: Representation of the transformation by Imaging: first attempt

The relevance of the document to the query is

$$\begin{aligned} P_d(q) &= P_d(w_2) + P_d(w_5) \\ &= P(w_1) + P(w_2) + P(w_5) \end{aligned}$$

This interpretation is not appropriate for a logic-based model based on the Transformation Principle because it should be the document that is transformed until the information requested by the query is found, and not that the knowledge set is transformed until d becomes true.

Another use of Imaging for a logic-based model of an IR system was proposed by Crestani and Van Rijsbergen [CvR94, CvR95a, CvR95b]. There, worlds model terms, and propositions model documents and queries. A term t "makes a document true" if that term belongs to that document. The function t_d (instead of w_t) gives the closest term of t that is contained in d. It is t if the latter is contained in the document. Imaging consists then of shifting the probabilities to the terms contained in d (i.e., the terms that make d true). This is illustrated in the following figure:

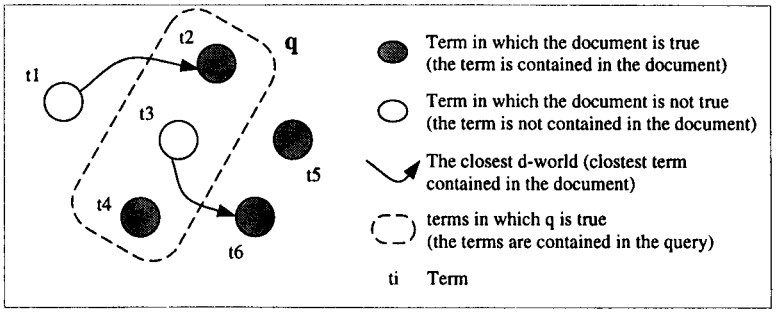


Figure 3.11: Representation of the transformation by Imaging: second attempt

The relevance degree is

$$\begin{aligned} P_d(q) &= P_d(t_2) + P_d(t_4) \\ &= P(t_1) + P(t_2) + P(t_4) \end{aligned}$$

Obviously, this approach is not appropriate to model the quantitative components of the model proposed in this thesis because it is not based on the notion of the transformation of a document. It also presents other drawbacks. Although the evaluation of the relevance takes into account the semantics between terms by shifting to those closer (semantically) terms contained in the document, this approach disregards the less closer terms, which may be contained in the query. For example, in Figure 3.11, suppose that t_4, which is contained in the query, is a second closest term of t_3. This relationship between t_3 and t_4 is not captured in the above expression of the relevance. Also, this approach cannot represent that two terms are closer to each other in some contexts, and are not in other contexts. Consequently, uncertain relationships cannot be encompassed.

The last possible interpretation is that a world w represents the document. Imaging consists then in finding all the worlds closest to w in which the query, represented by q, is true (the w_q world). Although it was not mentioned, there can be several closest worlds to a world. This is illustrated in the following figure:

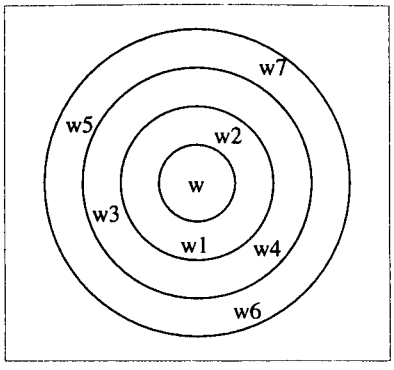


Figure 3.12: Representation of the transformation by Imaging: third attempt

The $q-$closest worlds to w are w_1 and w_2. The second $q-$closest worlds to w are w_3 and w_4. The imaging process will then shift the probability of $P(w)$ to the worlds w_1 and w_2[41]. This use of the closest world cannot model the transformation of the document, because the imaging process is

[41] The formulation is slightly different than in case of a single closest world (see the work of Nute [Nut80]). The formulation is not given since it requires the definition of other concepts which are not necessary for the above discussion.

only applied once. That is, it is not possible to represent that the document is initially represented by that particular world w, and then transformed into worlds, which may be then transformed into other worlds, until a world that makes q true is obtained.

In summary, Imaging cannot be used to model the quantitative components of the model since no appropriate representation of a transformation as defined in the Transformation Principle is possible.

3.4 Fuzzy Logic

Fuzzy Logic[42] [Zad65, Zim91, Zad87] extends Classical Logic by allowing multi-valued propositions. A numerical function $u : U \rightarrow [0,1]$ represents the truth of the propositions of a set U. The higher the truth of a proposition p of U, the higher the value $u(p)$. The truth values of the conjunction, disjunction and implication are defined as follows (p and q are two propositions):

(i) $u(p \wedge q) = \min(u(p), u(q))$

(ii) $u(p \vee q) = \max(u(p), u(q))$

(iii) $u(p \rightarrow q) = \min(1, 1 - u(q) + u(p))$[43]

A logic-based model of an IR system based on Fuzzy Logic was proposed by Nie [Nie90]. There, the relevance of a document to a query was also based on the transformation of the document into a document that contains the information expressed in the query. Nie's model does not consider the structured representation of documents, but it will still be described with respect to structures. The same components that were identified in this chapter were also identified in Nie's model: the uncertainty of a transformation, the propagation and the aggregation of uncertainty along the transformations of the structures, and the numerical expression of the relevance of a document to a query[44].

Nie defines the propagation of the uncertainty by a general function with some given properties. This function is not explicitly expressed but its behavior is similar to that representing the propagation of uncertainty in the model described in this thesis. One of these properties is that the uncertainty increases with the number of transformations.

The aggregation of the uncertainty is represented by the max function. If a structure is transformed in several ways to the same structure, then the uncertainty of the overall transformation is defined as that of the transformation that is less uncertain. This representation of the aggregation satisfies the property that uncertainty decreases when aggregated, and matches the test case (iv) as follows:

[42] There is some confusion on what Fuzzy Logic is about. In some cases, Fuzzy Logic is the framework concerned with the representation of metalinguistic predicates and natural language quantifiers (such as *most* and *often*). The vagueness of these concepts is modelled by fuzzy sets [Zad65, Zad87].

[43] The implication $\rightarrow$ is not to be confused with the representation of the implicit information content of a document as in $d \rightarrow q$.

[44] In Nie's model, given an item of information (a proposition), whether or not that item was in the document was evaluated as a fuzzy value. The value did not depend on how the item became contained in that document. This aspect is not discussed since in the model developed in this thesis, either an information item is contained in a document, or it is not.

$w^1(s)$	$w^2(s)$	$w(s')$
$u^1(s') = a$	$u^2(s') = b$	$u(s') = \max(a, b)$

Table 3.6: The aggregation of the uncertainty in Fuzzy Logic

The problem with this representation of the aggregation of the uncertainty is that it does not agree with the view of the existence of several transformations as an accumulation of evidence. Suppose that there are n parallel transformations. The overall uncertainty will be the one associated to the less uncertain transformation. If only that transformation was obtained, the resulting uncertainty will be exactly the same as if all the transformations were obtained. Moreover, the use of max does not distinguish between the two cases:

(i) reaching n times a structure with high uncertainty values, and

(ii) reaching $n - 1$ times that structure with low uncertainty values and once with a high uncertainty value.

The uncertainty of a structure that is derived in different ways should be higher and not just the highest.

Another drawback with Nie's model is that the function which models the propagation of the uncertainty is left undefined. Nie states that his model covers other IR models by instantiating this unspecified function. However, in most cases, these instantiations are not based on Fuzzy Logic. That is, Nie uses a non-uniform framework to model the uncertainty, which is not rigorous.

The representation of the propagation of the uncertainty in Fuzzy Logic can be derived from the formula defining the truth value of the implication. Applied to structures, this formula is

$$u(s \rightarrow s') = \min\left(1, 1 - u(s') + u(s)\right)$$

In this formula, only $u(s \rightarrow s')$ expresses a connection between the two structures s and s', so it must represent the uncertainty of the transformation of s into s'. The uncertainty of the structures s and s' is then represented by $u(s')$ and $u(s)$, respectively. Therefore, to represent the propagation of the uncertainty, $u(s')$ must be expressed in terms of $u(s \rightarrow s')$ and $u(s)$ so that to match the test case (iii). The expression is shown in the following table (the proof can be found in [Par94]):

$C(s, s')$	$w(s)$	$w(s')$
$u(s \rightarrow s') = a$	$u(s) = b$	Has a solution if $a + b - 1 \geq 0$ (i) if $a = 1$ then $b \leq u(s') \leq 1$ (ii) if $a < 1$ then $u(s') = a + b - 1$

Table 3.7: Representation of the propagation of uncertainty in Fuzzy Logic

There are obvious problems which such a representation of the propagation of the uncertainty. First, the propagation of the uncertainty cannot be always represented since there is a restriction on the values of $u(s \rightarrow s')$ and $u(s)$. Second, in some cases, the propagation of uncertainty is expressed as an interval, thus, leading to the same problem as with Probability Theory described in section 3.3.1. In addition, the intervals are larger than that obtained in Probability Theory

$(a * b \leq P(s') \leq 1 - b + a * b)$, so the representation of the propagation of the uncertainty is even more imprecise. Third, some of the results are counter-intuitive. Suppose that

$$u(s) = 0.5 \qquad \text{and} \qquad u(s \rightarrow s') = 0.5$$

According to the formula in Table 3.7,

$$u(s') = 0.5 + 0.5 - 1 = 0$$

meaning that the structure s' is completely uncertain. This result is rather contradictory, since the uncertainty of $u(s)$ and $u(s \rightarrow s')$ are not particularly low.

To conclude, Fuzzy Logic is not used to model the quantitative components of the model developed in this thesis since it cannot model properly the propagation and the aggregation of the uncertainty.

3.5 Dempster-Shafer's Theory of Evidence

The Theory of Evidence was first developed by Dempster [Dem68], then finalized by Shafer [Sha76]. The theory is presented in two steps. First, the initial theory is described, and is shown to represent some of the quantitative components of the model. Second, the *refinement function* later defined by Shafer is described, and is shown to represent the other quantitative components of the model.

3.5.1 The initial Theory of Evidence

The purpose of the Theory of Evidence is to represent beliefs in a set of propositions referred to as a *frame of discernment.* A *belief function* $Bel : 2^U \rightarrow [0, 1]$ is defined on a frame of discernment U. The beliefs are usually computed based on a density function m called a *basic probability assignment* (BPA) which has the following properties:

$$m(\emptyset) = 0 \quad and \quad \sum_{P \subseteq U} m(P) = 1$$

$m(P)$ represents the degree of belief that is exactly committed to the set P. If $m(P) > 0$ then P is called a *focal element.* The set of focal elements and its associated BPA define a *body of evidence* on U. The belief associated with a set $Q \subseteq U$, denoted as $Bel(Q)$, is defined on m as follows:

$$Bel(Q) = \sum_{P \subseteq Q} m(P)$$

$Bel(Q)$ is the total belief committed to Q, that is, the belief that the truth is in Q.

A commonly used rule is *Dempster's combination rule.* This rule aggregates two bodies of evidence defined within the same frame of discernment into one body of evidence. Let m_1 and m_2 be two bodies of evidence defined in the frame of discernment U. The new body of evidence is defined by a BPA m as follows:

$$m(A) = \frac{\sum_{B \cap C = A} m_1(B) * m_2(C)}{\sum_{B \cap C \neq \emptyset} m_1(B) * m_2(C)}$$

A logic-based model of an IR system has been developed by de Silva and Milidiu [dSM93]. Their model is described with some details since some of its features pertain to the quantitative components as studied in this thesis. The model starts with the definition of a set of terms and their associated semantics. Given a term t, $S(t)$ and $N(t)$ are the sets of synonyms and narrower terms of t, respectively. $N(t) \subseteq S(t)$, where $\subseteq$ is viewed as semantically included. Examples of the narrower terms and the synonyms of the term "flower" are, respectively

$$N(flower) = \{rose, tulip\}$$
$$S(flower) = \{blossom, bud, flower, pompon, rose, tulip\}$$

For each set of synonyms $S(t)$, one term in this set is used as a descriptor. For example, the term "flower" is the descriptor of the above set of synonyms. A descriptor α is atomic if it does not have a narrower term; that is, $N(\alpha) = \emptyset$.

The set of atomic descriptors constitutes a frame of discernment, denoted Θ. Both the document and the query are defined as a body of evidence in this frame of discernment. α is a descriptor of the document if at least one term in $S(\alpha)$ appears in the document. Each descriptor of the document defines a focal element. The focal element associated to a non-atomic descriptor α of the document is defined as the union of the atomic descriptors in $N(\alpha)$. For example, suppose that "flower" is a descriptor of the document. If "rose" and "tulip" are atomic descriptors (i.e., $N(rose) = \emptyset$ and $N(tulip) = \emptyset$), then the focal set associated to the descriptor "flower" is

$$\{rose, tulip\}$$

The BPA of a focal element representing the descriptor α is defined as follows:

$$m_d(\alpha) = \frac{\sum_{t \in S(\alpha)} f(t,d)}{\sum_{t \in T(d)} f(t,d)}$$

where $T(d)$ is the set of terms in the document and $f(t,d)$ is the frequency of the term t in the document[45].

A similar approach is adopted in this thesis to represent the structures forming the original document and their weights. This document is represented by a frame of discernment, and the propositions in this frame represent information items. Structures are represented by focal elements. The use of the relationships between information items to define the focal elements is analogous to that above described, although it depends on how semantically related information items are defined. The weights associated with these structures are represented by the BPA, which is computed similarly to that above described; that is, the computation takes into account frequency information. The detailed obtaining of the focal elements and their BPA is discussed in Chapters 5 and 6. Furthermore, this approach can also model the document in the unstructured model. One focal

[45] The belief associated to this frame of discernment is defined for each $S \subset \Theta$ as

$$Bel_d(S) = \sum_{t \subset S} m_d(t)$$

$Bel(S)$ is the belief that the descriptors in S are the best semantic representation of the document. As mentioned by de Silva and Milidiu, the representation of the document requires only the definition of the focal elements and the BPA, since the belief function is computed from them.

element is defined and corresponds to the frame of discernment itself. The BPA associated to this focal element is equal to 1.

To conclude, the representation of the significance of information (the weight of the structures) can be well captured with Dempster-Shafer's framework. If m_d is the BPA associated to the original document, the matching of the test case (i) is as follows:

$w(s)$ for s a structure of the original document
$m_d(s)$

Table 3.8: The representation of the significance of information in the Theory of Evidence

In de Silvia and Milidiu's model, the query is also represented as a body of evidence associated with the frame of discernment Θ. Let $T(q)$ be the set of terms used in the query q, which all correspond to descriptors. Let $w(\alpha)$ be the weight that expresses a user's belief in α being a descriptor that represents the semantic content of the document to be retrieved. The BPA associated to this frame is defined in terms of this weight as follows[46]:

$$m_q(\alpha) = \frac{w(\alpha)}{\sum_{t \in T(q)} w(t)}$$

The relevance of the document to the query is computed as the agreement, denoted $A(d,q)$, between the document and the query. Several formulations of $A(d,q)$ are possible, depending on the properties attached to the terms. In one of them, the descriptors in the document and the query are independently determined, which leads to the following formulation of $A(d,q)$ (refer to [dSM93] for the proof):

$$A(d,q) = \sum_{A \cap B \neq \emptyset} m_d(A) * m_q(B)$$

The intuition behind the expression of the relevance degree in de Silvia and Milidiu's model is different from that in a logic-based model based on the Transformation Principle. In the former, the relevance consists of a comparison between the information contained in the document and the information requested by the query (although it did take into account semantics of information). In the model proposed in this thesis, the relevance is based on obtaining a transformed document that contains the information requested by the query. Therefore, except from the representation of the significance of information, de Silvia and Milidiu's model cannot represent the other quantitative components of a logic-based model proposed in this thesis, since it does not account for the representation of transformed documents.

None of the other concepts of the Theory of Evidence thus far described can appropriately represent the other quantitative components of the model because they cannot capture the transformation of a document. The belief function associated to the frame of discernment is computed based on the BPA of the focal elements of that frame. If that frame represents the original document, the

[46] $Bel(\alpha)$ is the user's belief in α being the best representative content of the document that he or she would like to retrieve. It is computed as for the document. Its value is also defined in terms of the focal elements and the BPA.

propositions in that frame represent information items that are explicit in the document. Therefore, the belief function does not take into account the information that is implicit and uncertain, which needs to be represented elsewhere (the transformed document).

As well, Dempster's combination rule cannot embody the transformed documents. Indeed, Dempster's combination rule aggregates two independent bodies of evidence into one body of evidence. Only the bodies of evidence change, not the propositions of the frame of discernment. To embody the transformation of a document, both the original document and the transformed document must be represented by the same frame of discernment. Moreover, the first body of evidence would represent the original document, and the second body of evidence would model the transformed document. However, the second body of evidence is not computed from the first body of evidence. In fact, Dempster's rule defines a third body of evidence in terms of the two previous ones. Therefore, it is not possible to represent that the transformed document is constructed in terms of the original document.

In conclusion, the initial Theory of Evidence can model the significance of information, but not the other quantitative components. The reason is that a single frame of discernment is used, so transformed documents cannot be represented. Shafer's refinement function [Sha76] overcomes this problem.

3.5.2 Shafer's refinement function

There are two aspects to the refinement function, a qualitative and a quantitative one. These are discussed in turn.

3.5.2.1 The qualitative aspects of the refinement function

The refinement of a frame of discernment U into a frame of discernment V is defined by *splitting* the propositions of U into the propositions of V. Splitting a proposition into a set of propositions can be viewed as the latter representing more precise items of information that the former. For example, "animal" can be split into "dog", "cat" and "horse", since "dog", "cat" and "horse", are, each of them, more precise than "animal". The refinement is formally defined by a function $\omega : 2^U \rightarrow 2^V$ as follows:

(i) $\omega(\{p\}) \neq \emptyset$ for all $p \in U$

(ii) $\omega(\{p\}) \cap \omega(\{p'\}) = \emptyset$ if $p \neq p'$ for all $p, p' \in U$

(iii) $\bigcup_{p \in U} \omega(\{p\}) = V$

(i) means that every proposition of U is split into propositions of V. (ii) means that two propositions cannot be split into the same proposition. Finally, (iii) means that the result of a refinement is a frame of discernment. U and V are called the *coarse* and the *refined* frame, respectively. In the above example, suppose that "animal" is in the coarse frame of discernment U, then

$$w(\{animal\}) = \{dog, cat, horse\}$$

and "dog", "cat" and horse" are in the refined frame of discernment V. The refinement function

is extended to sets of propositions as follows:

$$for\ all\ A \subseteq U,\ \ \omega(A) = \bigcup_{p \in A} \omega(\{p\})$$

$\omega(A)$ consists of all the propositions in V that are obtained by splitting all the propositions in A. For example, if "flower" is split into "rose" and "tulip", then

$$w(\{animal, flower\}) = \{dog, cat, horse, rose, tulip\}$$

The refinement function links two frames of discernment, such that one is defined in terms of the other. If the original document is modelled by the coarse frame, then the refinement function can represent the transformation of that document; the refined frame models the transformed document. The splitting process must then be defined in terms of relationships of the knowledge set K. That is, the fact that a proposition p is split into a proposition p' can be viewed as that $p \rightsquigarrow p'$ (to be more correct, the information items these propositions represent) is a relationship of the knowledge set. For example, the splitting of the term "animal" into "dog", "cat" and "horse" means that

$$animal \rightsquigarrow dog$$
$$animal \rightsquigarrow cat$$
$$animal \rightsquigarrow horse$$

are relationships. Note that the relationships can be uncertain, since when mentioning "animal", it is not sure whether one means "dog", "cat" or "horse".

Shafer demonstrates that the composition of two refinement functions is also a refinement function. That is, given the two refinement functions

$$w_1 : 2^U \rightarrow 2^V$$
$$w_2 : 2^V \rightarrow 2^W$$

where W is a frame of discernment, into which V is refined, this means that ($\circ$ is the composition operator)

$$w_2 \circ w_1 : 2^U \rightarrow 2^W$$

is also a refinement function. If a refinement function is used to model the transformation of a document, the composition of refinement functions can model a sequence of transformations.

In the model proposed in this thesis, the structures of the transformed document are defined in terms of the structures original document. If the refined frame is to model the transformed document, the focal elements of the refined frame must be defined in terms of the focal elements of the coarse frame of discernment, since the focal elements model structures. However, in Dempster-Shafer's framework, the focal elements of the refined frame are not explicitly defined in terms of the focal elements of the original frame, because the refinement function is defined at the proposition levels and then generalized to set levels. There are, however, properties relating the two sets of focal elements. These properties concerns the BPA associated to the focal elements of the two frames. These are discussed next since they constitute a quantitative aspect of the refinement function.

3.5.2.2 The quantitative aspects of the refinement function

Let Bel_U and Bel_V be the belief functions defined on the coarse frame U and the refined frame V, respectively. Let m_U and m_V be their respective BPAs. In Shafer's definition, m_V is not explicitly defined in terms of m_U. However, the belief functions Bel_U and Bel_V must satisfy the criteria that the coarse and the refinement frames are *compatible*. This means that the two frames must agree on the information defined in them. Shafer explains [Sha76] that for a given set A of the frame of discernment U, and for a given refinement function $w : 2^U \rightarrow 2^V$, the sets A and $w(A)$ represent the same information. That is, although refining a set means that more precise items of information are obtained, the union of these items carries the same information as the original set. For example, if the set

$$A = \{animal\}$$

is refined into the set

$$w(A) = \{dog, cat, horse, elephant, \ldots\}$$

(where ... refers to any living animal), then the same information is carried out by the two sets. Shafer explains in details this notion of compatible frames and formulizes it. The details and the formalism are not given here since they involve notions that are not necessary to the understanding of the concepts used in this chapter. What should be known is that the belief functions Bel_U and Bel_V are compatible if, for a given set A of the frame U, the following property holds:

$$Bel_U(A) = Bel_V(\omega(A))$$

Shafer [Sha76] proved that this is ensured if the following equality holds:

$$m_U(A) = \sum_{B \subseteq V, A = \overline{\theta}(B)} m_V(B)$$

where

$$\overline{\theta}(B) = \{x \in U | w(\{x\}) \cap B \neq \emptyset\}$$

This set $\overline{\theta}(B)$ is called the *outer reduction* of the refinement of the set B. This link between the BPAs m_U and m_V is illustrated in the following example of a refinement of the frame U to the frame V:

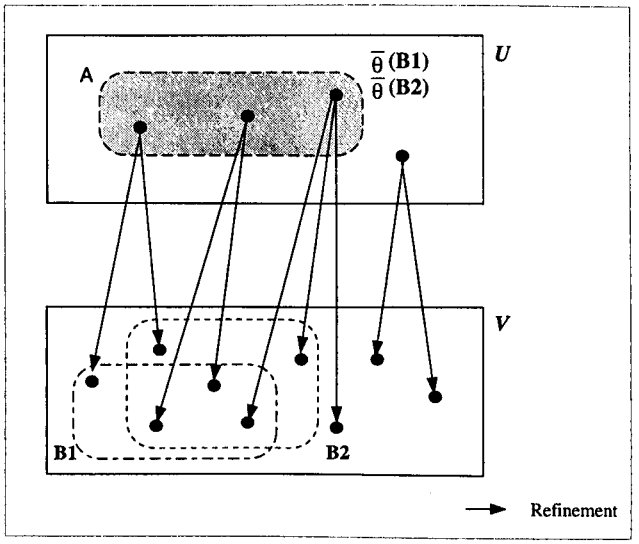


Figure 3.13: Outer reduction of a refinement

Both $\overline{\theta}(B_1) = A$ and $\overline{\theta}(B_2) = A$; that is B_1 and B_2 have the same outer reduction. Therefore, the following equality must hold:

$$m_U(A) = m_V(B_1) + m_V(B_2)$$

If A represents a focal element of U, and B_1 and B_2 constitute focal elements of the refined frame V, then this link between the set A and the sets B_1 and B_2 and their respective BPAs can be used to model the transformation of structures, and the propagation of the uncertainty associated with the transformation. This is illustrated in the following example of a refinement which involves more sets:

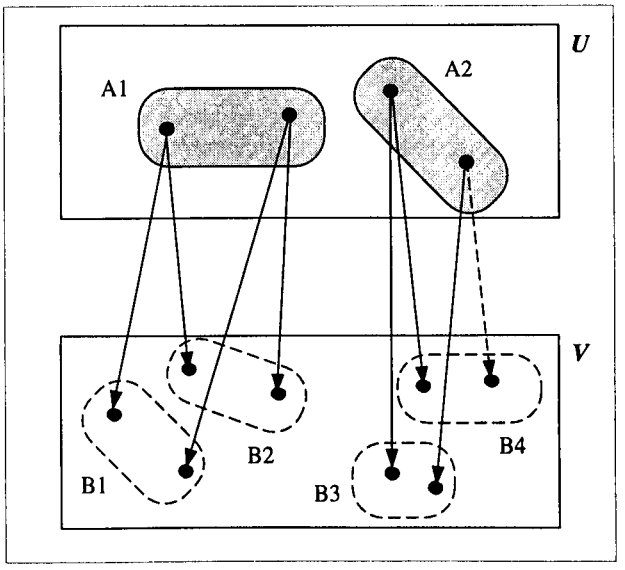


Figure 3.14: Example of a refinement that leads to the representation of the transformation of structures

Both $\overline{\theta}(B_1) = \overline{\theta}(B_2) = A_1$ and $\overline{\theta}(B_3) = \overline{\theta}(B_4) = A_2$. Suppose that these sets correspond to focal elements. For the two frames U and V to be compatible, the following equalities must hold:

$$m_U(A_1) = m_V(B_1) + m_V(B_2)$$
$$m_U(A_2) = m_V(B_3) + m_V(B_4)$$

which implies the following inequalities:

$$m_U(A_1) \geq m_V(B_1) \quad m_U(A_2) \geq m_V(B_3)$$
$$m_U(A_1) \geq m_V(B_2) \quad m_U(A_2) \geq m_V(B_4)$$

The BPA associated to a focal element in the refined frame is lower than that of the focal element of the original frame which corresponds to its outer reduction. Consequently, if the focal elements represent structures, and if the BPAs m_U and m_V represent the uncertainty attached to the structures in the original document and the transformed document, respectively, then the BPA associated to V can be used to embed the propagation of the uncertainty since it captures the fact that uncertainty increases when propagated. For a structure s represented by a focal element of the original frame U that is transformed into a structure s', the focal element representing that transformed structure (s') in the refined frame V must be such that its outer reduction is that exact focal element representing the structure s. This special case of the refinement can then be used to model the transformation of the structures.

What is left is to explicitly define the BPA m_V in terms of the BPA m_U so that it matches the test case (iii). That is, if $m_V(B_1)$ and $m_U(A_1)$ represent the focal elements associated to the structures s and s', it is first necessary to define an entity representing the uncertainty of the transformation of s into s', and second, to define $m_V(B_1)$ in terms of $m_U(A_1)$ and this uncertainty. The latter means that an entity representing the uncertainty of A_1 being refined into B_1 must be defined. This could be viewed as defining a *quantification* of the refinement function. Shafer's refinement function does not compute the BPA m_V in terms of the BPA m_U, nor does it quantify the refinement function. However, the quantification of the refinement function and the explicit definition of m_V in terms of m_U do not contradict the ontology of Shafer's refinement function; they only express a specific use of the refinement function, for a BPA is defined in each frame.

To conclude, the use of the BPAs of the coarse frame and refined frame to model the propagation of the uncertainty matches the test case (iii) as follows:

$C(s, s')$	$w(s)$	$w(s')$
To define	$m_U(s) = b$	$m_V(s') \leq b$

Table 3.9: Representation of the propagation of the uncertainty in the Theory of Evidence

Documents can be modelled by frames of discernment, and the transformation process can be modelled by a refinement function. The composition of refinement functions can model sequential transformations. The last refined frame is constituted of all the information items, either explicitly or implicitly, contained in the document. The belief function associated with that frame can act as a measure of relevance. If m_f is the BPA associated to that final frame, then the belief function is defined as the summation of the BPA of those focal elements that contain information relevant to the query represented by q. This matches case (v) as follows:

$w(s_1)$	$w(s_2)$	$r(d \rightarrow q)$
$m_f(s_1) = a$	$m_f(s_2) = b$	$Bel(q) = a + b$

Table 3.10: The representation of the relevance degree in the Theory of Evidence

The above formulation then captures the fact that the more such transformed structures are obtained, the higher the belief, and the higher the relevance.

There is however one problem which the use of a refinement function to model the transformation of a document; parallel transformations cannot be captured. An example of a refinement that would capture parallel transformations is given in the Figure 3.15.

Suppose that A_1 and A_2 are the only two focal elements of the coarse frame, and B_1, B_2, B_3 and B_4 are those of the refined frame.

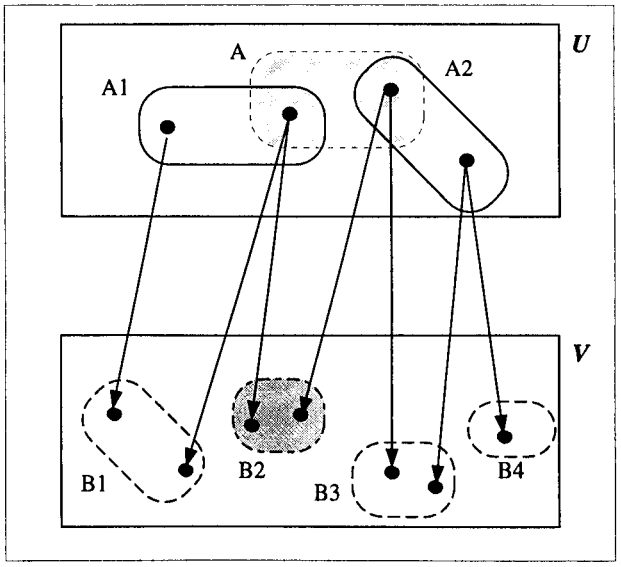


Figure 3.15: Example of a refinement function that would lead to the representation of parallel transformations

To symbolize that the structures represented by the focal elements A_1 and A_2 can be both transformed into the structure represented by the focal element B_2, first, the outer reduction of B_2 must be defined as the union of A_1 and A_2, and, second, $m_V(B_2)$, $m_U(A_1)$ and $m_U(A_2)$ must be somewhat related. However, the outer reductions of the sets B_1, B_2, B_3 and B_4 are

$$\overline{\theta}(B_1) = A_1 \quad \overline{\theta}(B_2) = A$$
$$\overline{\theta}(B_3) = A_2 \quad \overline{\theta}(B_4) = A_2$$

For the two frames of discernment to be compatible, the following equalities must hold:

$$m_U(A_1) = m_V(B_1)$$
$$m_U(A_2) = m_V(B_3) + m_V(B_4)$$
$$m_U(A) = m_V(B_2)$$

That is, A must be a focal element of the original frame, if B_2 is to be a focal element. Moreover, $m_V(B_2)$ does not relate to $m_U(A_1)$, nor $m_U(A_2)$. Therefore, it is not possible to captures that a structure is the result of several transformations based on the notion of outer reduction.

In conclusion, the Theory of Evidence is the most appropriate framework to represent the quantitative components of a logic-based model of an IR system based on the Transformation Principle. Expect for the problem occurring with the representation of parallel transformations, which means that the aggregation of uncertainty, as studied in this thesis, is not captured, all the other components can be adequately represented, or at least captured (i.e., the uncertainty of a transformation) by such a framework. Therefore, the framework is used in this thesis to model these components. A slightly modified definition of the refinement function that encompass the problem related to the representation of parallel transformations will be used.

3.6 Conclusion

This chapter was concerned with the representation of the quantitative components of a logic-based model of an IR system which follows the Transformation Principle. These quantitative components

are *the significance of the information, the uncertainty of the transformation, the propagation and the aggregation of the uncertainty, and the numerical expression of the relevance.*

The process defined by these quantitative components is similar to that of an uncertain inference process, which is usually modelled by a Theory of Uncertainty. Several Theories of Uncertainty are examined in this chapter to represent these quantitative components. These theories are Probabilistic-based frameworks, Fuzzy Logic and Dempster-Shafer's Theory of Evidence.

Three probabilistic-based frameworks were examined: Probability Theory itself, Bayesian Inference and Imaging. Probability Theory could not be used for modelling the quantitative components because no adequate interpretation or representation of the components was possible. The Bayesian Methods were rejected because they assumed many estimations, which were not obvious to compute. Imaging could not be used because no appropriate interpretation of the concepts was adequate to model the transformations of documents.

Fuzzy Logic proved to be counter-intuitive. For example, the use of the max as the aggregation of the uncertainty could not capture an accumulation of evidence. Also, the use of Fuzzy Logic resulted in very imprecise values of uncertainty.

The framework that models best the quantitative components is Dempster-Shafer's Theory of Evidence, together with the notion of refinement later introduced by Shafer. Dempster-Shafer's initial framework allows the representation of the significance of information. The later framework allows the representation of the other quantitative components. The use of the overall framework presents the advantage that it can be easily mapped to the quantitative structured representation of a document, and its transformation.

A Theory of Information and a Theory of Uncertainty have been selected to model the qualitative and the quantitative components of a model of an IR system based on the Transformation Principle. These two theories are Situation Theory and Dempster-Shafer's Theory of Evidence, respectively. In the next two chapters the unstructured and the structured models are proposed based on these theories.

Chapter 4
Description of the Model for an Unstructured Representation of Information

4.1 Introduction

This thesis proposes a model for an IR system with both a theory of information and a theory of uncertainty. The theory of information is used to represent the qualitative components of the system, whereas the theory of uncertainty is used to model its quantitative components. One method of achieving this goal is to base the model on the Transformation Principle. This enables the identification of the qualitative and quantitative components listed in the following table:

Qualitative components	Quantitative components
Document	Significance of information (Weight)
Query	Propagation of uncertainty
Knowledge set (Semantic relationships)	Aggregation of uncertainty
Transformation (Flow of information)	Relevance degree (Uncertainty value)

Table 4.1: The quantitative and the qualitative components

The survey of qualitative frameworks in Chapter 2 indicated that Situation Theory forms a relevant theory of information for IR. The survey of quantitative methods in Chapter 3 showed that Dempster-Shafer's Theory of Evidence embodies all the quantitative components described above. This chapter and the next chapter demonstrate that both theories achieve their goal of modelling their respective components of the IR system. Moreover, they show that the theories can be combined to form a model of the IR system that follows the Transformation Principle.

The model is developed in two stages. The first stage is to represent information and its flow, without taking into account the significance of information or its structure. The uncertainty is represented by a general inference mechanism. The second stage takes into account the structured representation of the document. Dempster-Shafer's Theory of Evidence is used to capture the uncertainty. The description of the second stage of the model is the topic of the next chapter.

The present chapter is concerned with the first stage, which is the development of a model of an IR system for an unstructured representation of information. This model is referred to as the *unstructured model.* First, the merits of the use of Situation Theory as a basis for a model of an IR system are discussed.

4.2 Situation Theory for Information Retrieval

There are many reasons for using Situation Theory as the qualitative framework for modelling an IR system. The foremost is that a suitable model should manipulate information and its flow as they appear and are handled in the real world. In this thesis, the information comes from text documents. A brief description of the main components of Situation Theory is given first.

4.2.1 Infons, situations and types

Situation Theory represents information without specifically indicating what information is. It considers information as a fundamental entity; from this, a model of the flow of information is derived. Situation Theory is concerned with information of the form

A property P holds / does not hold for the set of objects $a_1, \ldots, a_n$.

These two items of information are modelled by the two *infons*, one being the *dual* of the other, respectively

$$\ll P, a_1, \ldots, a_n; 1 \gg \text{ and } \ll P, a_1, \ldots, a_n; 0 \gg$$

Situations are parts of the world from which information is extracted. Let σ be an infon representing an item of information. If a situation s makes this information true, this is denoted $s \models \sigma$ (read "support").

Types represent the uniformities that cut across infons. For example, the three following infons

$$\ll Weather, Glasgow, sunny; 1 \gg$$
$$\ll Weather, Windsor, sunny; 1 \gg$$
$$\ll Weather, Algiers, sunny; 1 \gg$$

have the common information that it is sunny. What differs in these infons is the city. The type abstracting among these infons can be defined as

$$\varphi = [\dot{s} | \dot{s} \models \ll Weather, \dot{c}, sunny; 1 \gg]$$

which is the type of any situation about a city (represented in the type by the parameter $\dot{c}$) where the sun is shining. If s is one of them, this is written $s \models \varphi$. In [Dev91], a detailed description of infons and situations, together with a set of rules that ensure proper instantiating (called *anchoring*) of parameters, is given.

Dretske [Dre81] provides a comprehensive read about the role of information and its flow. Indeed, the philosophy behind the development of Situation Theory conforms to many of the points expressed by Dretske. Those relevant to IR are discussed next, and their representations within Situation Theory ontology are emphasized.

4.2.2 Digital vs. analog

Dretske [Dre81] explains that information is knowledge about a source, which is communicated by a signal to a receiver. Here, the source is the document and the receiver is anybody observing the document (reading a text, listening to an audiotape or observing an image). Signals are whatever means by which the information about a source is delivered to the receiver. For example, if the document is a text, the signal is a mixture of the reader's vision capability, his/her understanding of the information read, and his/her general knowledge about its subject. A signal can also be the indexing process which delivers a representation of the information content of the document.

Dretske [Dre81] defines a source as any structure with an information content; situation is of primary concern here. Let d be a situation and φ be an information item. If the signal carries the information that d contains φ (or, as expressed by Drestke, "d is φ"), this is written in Situation Theory $d \models \varphi$.

A signal which carries $d \models \varphi$ often carries additional information about the situation d or other situations owing to the exact fact that d supports φ. This information is said to be *nested* into the fact that d supports φ. This notion is important to Chapter 5 where information structures are defined.

Dretske notes that information can be encoded in two forms: a signal carries the information $d \models \varphi$ in *digital form* if, and only if, the signal carries no additional information that is not nested in d supporting φ; otherwise, the signal carries this information in *analog form.*

4.2.3 Perception

According to Dretske [Dre81], *perception* is the process by which information is delivered to a *cognitive agent* for its selective use. It is identified with a signal that carries information about a source which is coded in analog form. Until information has been extracted from this signal, nothing corresponding to recognition, classification or identification has occurred. It is the successful conversion of information into the appropriate digital form that constitutes the essence of a cognitive activity. In Situation Theory, situations are the objects of perception. They provide the information that signals carry in analog form.

A perception process often embodies information about a variety of details that, if carried over in total to the cognitive agent, would require immense storage and retrieval capabilities. Moreover, there is more information than can be extracted and/or exploited by the cognitive agent. Only some of the information the perception process carries in analog form is retained. The same holds true with most (if not all) IR systems. The indexing of a document does not give an exhaustive description of the information content of that document. There would be too much information to store, and sometimes it is not even possible to exhaustively determine the information content.

A perception process is determined not by *what* information is carried, but by the *way* it is carried. Seeing, hearing or reading are not different processes because of the information they carry (the information might be the same), but because of the *vehicle* by which this information is delivered. Two different concepts are involved here:

(i) *how* the information is *delivered*, and
(ii) *what* the information *represents*.

Situation Theory is concerned with the latter, for a situation can be a text, an image or a speech. Therefore, a model based on Situation Theory could eventually incorporate multimedia IR systems.

4.2.4 Cognition

Dretske [Dre81] describes *cognition* as the conversion of the information a cognitive agent receives in analog form into digital form. The result is often qualified as a *knowledge* with respect to the cognitive agent. The conversion, referred to by Dretske as *digitalization*, involves a loss of information because it turns a structure of greater information content to one of lesser information content.

The indexing process in IR can be compared to a digitalization process. The document is a situation that contains information in analog form. The information which is (successfully) digitalized constitutes the document representation. The goal is to minimize the loss of information involved in the conversion while at the same time obtaining a small enough document representation for both storage capacity and retrieval speed.

Some researchers [Lan86, Bar89] refer to a situation as a *partial object*, which can contain a vast amount of information, though only part of it is digitalized. For example, ask different people to describe the same event and you will often obtain different descriptions of that event. Whether an item of information is to be digitalized or not depends on two properties attached to the cognitive agent:

(i) its *capability of perception*. For example, a human being and a robot do not perceive information at the same level. A robot can identify entities that a human being cannot, and vice versa.

(ii) its *focus of attention*, because cognitive agents are often constructed to fulfill a task. For example, the color of a wall may be of no interest to a moving device whose purpose is to avoid the wall.

The essence of Situation Theory is to capture these facts, which is often not the case with most truth-based frameworks. In these, every representation of an information item is assessed to either belong or not to belong to the document (the assessment is often a truth value). This is unreasonable because many information items have no connection whatsoever with the information content of the document, so the assessment should only be made if necessary. This could be either *negative* (e.g., "the document is not about the political situation in Quebec")[47] or *positive* (e.g., "the document is about the religious problems in Algeria"). Situation Theory captures this phenomena, which was referred to as *partiality* in Chapter 1.

4.2.5 Information vs. meaning

Dretske [Dre81] claims that information and meaning are two different concepts. Indeed, there is

[47] Negativity here does not mean the non-existence of the item of information.

no reason to assume that the information a signal carries is identical to its meaning. Often, the information contained in a signal exceeds its meaning. For example, the statement "Keith is at home" means that Keith is indeed at home. It does not mean that "Keith is at home and not at work", though Keith being at home implies that Keith is at home and not at work. A signal that carries "Keith is at home" also carries "Keith is at home and not at work". This difference is highlighted by Dretske [Dre81]:

> "... information is that commodity of yielding knowledge, and what information a signal carries is what we can learn from it".

In IR, understanding the meaning attached to the sentences of a document is important, but is insufficient for determining the information content of the document. This is why frameworks such as Montague Semantics [DWP81, Mon74] are not appropriate, since they are theories of meaning, whereas Situation Theory is a theory of information.

4.2.6 Constraints and the flow of information

Constraints model relationships between types to represent, for example, relationships such as "if I keep practicing my free style I will become a good swimmer" or "Scandinavian countries have very cold winters". Let ϕ and φ be two types that constitute the constraint $\phi \rightarrow \varphi$. The application of this constraint to a situation s_1 is possible if first $s_1 \models \phi$ and then informs on the existence of a situation s_2 such that $s_2 \models \varphi$. The fact $s_1 \models \phi$ carries the information that $s_2 \models \varphi$. A *flow of information* circulates between the situations s_1 and s_2, and the nature of the flow is defined by the constraint $\phi \rightarrow \varphi$. The result of the flow is that $s_2 \models \varphi$.

A flow of information arises between two situations, meaning that the information about one situation contains information about the other situation. If the two situations are the same, the information about the situation carries information about itself. That is, if $s_1 = s_2$, the flow gives additional information about the situation s_1 itself.

In IR, constraints can model any thesaural, semantic or pragmatic relationships, or more complex relationships like those handled by artificial intelligence. In further references, the term "semantics" is used to refer to information-based relationships. These include relationships defined upon meaning; an example is "airplane" and "aircraft", which can be considered as synonymous[48]. They also include relationships defined upon background knowledge. An example is the systematic relationship that most people attach to "wine" and "France".

4.2.7 Conditional and unconditional constraints

Constraints do not always hold. For example "Winters in Windsor are mild" is a generally true assumption which can sometimes fail to hold, as it did on my arrival in January 1994 (it was the coldest winter of the decade). The constraints that always hold are called *unconditional* and those that do not are called *conditional*. The latter indicates that the realization of some constraints may be *uncertain*. In Situation Theory, this uncertainty is captured by *background conditions*. A conditional constraint is written $\phi \rightarrow \varphi | B$, which highlights the fact that the constraint $\phi \rightarrow \varphi$

[48] In reality, airplane is a subset of aircraft.

holds if the background conditions captured within B are met. The background conditions are often represented as a set of types. So $\phi \rightarrow \varphi|B$ holds if a situation s such that $s \models \phi$ is also of type B, that is $s \models B$.

The use of background conditions allows the rejection of the two following rules of Classical Logic:

(i) from $\phi \rightarrow \varphi$ and $\varphi \rightarrow \chi$ infer $\phi \rightarrow \chi$
(ii) from $\varphi \rightarrow \chi$ infer $\varphi \wedge \phi \rightarrow \chi$

The first rule holds if the background conditions associated with $\phi \rightarrow \varphi$ and $\varphi \rightarrow \chi$ are compatible (they present some commonality). The second is sustained if φ and ϕ are supported by the same situation and ϕ is compatible with the background conditions associated with $\varphi \rightarrow \chi$. In IR, background conditions can represent intensional expressions, examples of which are polysemic words. Consider the word "bank" in a document dealing with finance. Inference with respect to that word should relate to the "money bank" context, and not "river bank"[49].

The background conditions can be particularly complex to identify. In every day reasoning, people often use background conditions, though they are not aware of them. People often, if asked, cannot express them. This should not imply the non-existence of the background conditions. As Devlin [Dev91] points out, background conditions become a concern only when a constraint fails.

4.2.8 The general idea of a model based on Situation Theory

From Dretske's account of the role of a theory of information [Dre81], Situation Theory seems the right framework for the qualitative modelling of the IR system. Situations and types show similarities with documents and their information content. Supported information corresponds to the explicit information content (digitalized) of the document, whereas carried information corresponds to its implicit information content. The Transformation Principle is re-expressed within Situation Theory ontology:

> "The extent to which d is relevant to φ, relative to a given knowledge set K, is based on the minimal extent to which it is necessary to transform d into d' such that $d' \models \varphi$".

d is the situation modelling the document and φ is the type modelling the information need expressed in the query. In the remainder of this chapter, the existence of a set of situations S and a set of types T is assumed.

The knowledge set K is an essential component of the model because the transformation of documents depends on K. The representation of the knowledge set within Situation Theory ontology is the topic of the next section.

4.3 The knowledge set

The knowledge set symbolizes the semantics of information. In Situation Theory, semantics are relationships between information items, where the relationships are modelled by constraints. When

[49] Disambiguation is necessary, which unfortunately is not always successful or even possible [San94, Voo93, KC92].

viewed as a situation, a document supports information and often carries implicit information that depends on the available constraints. It is the application of a constraint to a situation that leads to a flow of information, which then relates the same situation or two different situations. For example, with the constraint (from [Bar89])

$$[\dot{s}|\dot{s} \models\ll Kissing, \dot{x}; 1 \gg] \rightarrow [\dot{s}|\dot{s} \models\ll Touching, \dot{x}; 1 \gg]$$

the same situation is involved. With the constraint

$$[\dot{s}|\dot{s} \models\ll In, Windsor, Mounia; 1 \gg] \rightarrow [\dot{t}|\dot{t} \models\ll Sad, Steve, Glasgow; 1 \gg]$$

two situations are linked, one related to the city of Windsor, the other to the city of Glasgow. In addition, constraints are either unconditional or conditional. Their applications lead to certain or uncertain flows, respectively. For example, the constraint (from [Dev91])

$$[\dot{s}|\dot{s} \models\ll Presence, smoke; 1 \gg] \rightarrow [\dot{s}|\dot{s} \models\ll Presence, fire; 1 \gg]$$

is certain because whenever there is fire, there is smoke. The constraint

$$[\dot{s}|\dot{s} \models\ll Ringing, doorbell; 1 \gg] \rightarrow [\dot{s}|\dot{s} \models\ll At, door, somebody; 1 \gg]$$

is uncertain because it is not always the case that when the door bell rings, someone is standing at the door. The latter constraint is defined in terms of a set of background conditions. Hence, four types of flow occur:

(i) certain and relating the same situation,
(ii) certain and relating two different situations,
(iii) uncertain and relating the same situation, and
(iv) uncertain and relating two different situations.

As studied in this thesis, the flow of information that may originate from a document is not the information that document has about another document, but what yields the implicit information of a document. If the flow is certain, this information can be considered part of the information content of the document. If the flow is uncertain, this information might not be part of the document's information content. In that instance, a *fictitious* document (which is modelled by a situation) is constructed[50]. This situation contains the information delivered by this flow. Consequently, only flows of types (i) and (iv) are considered here.

The constraints that lead to flows of type (i) or (iv) are modelled by the two sets K_1 and K_2, which are the set of unconditional constraints and the set of conditional constraints, respectively. Let d be a situation such that $d \models \phi$. The application of an unconditional constraint $\phi \rightarrow \varphi \in K_1$ on the situation d means that $d \models \phi$ carries the information that $d \models \varphi$. The application of a conditional constraint $\phi \rightarrow \varphi|B \in K_2$ on the situation d depends on the satisfaction of the background conditions B by d. Three cases occur:

(i) $d \models B$, the flow is certain, therefore $d \models \varphi$. The constraint behaves as if unconditional. The constraint is said to be certain with respect to the situation d.

[50] As explained in Chapter 2, this fictitious document (or the corresponding situation) is, in practice, a representation of the document that includes implicit information.

(ii) $d \not\models B$, the flow is not realized.

(iii) nothing can be said as to whether or not $d \models B$ (this is often the case in IR). The resulting uncertain flow leads to a situation d' such that $d' \models \varphi$. The constraint is said to be uncertain with respect to the situation d.

In the third case, the flow of information is uncertain. The relevance of the document is strongly dependent on this uncertainty because the more uncertain is the flow, the less relevant the document is to the query. One method to represent the effect of this uncertainty upon the relevance is to quantify the uncertainty engendered by the flow. Since this uncertainty is engendered by the use of uncertain conditional constraints, the quantification can be derived from the background conditions associated to these constraints.

An uncertainty value is associated with the background conditions of each constraint, and measures the uncertainty involved in using that constraint when it is not known if its background conditions are satisfied by a situation. The function

$$cert : S \times BC \rightarrow [0,1]$$

is introduced for that purpose, where $BC \subseteq 2^T$ is the set of background conditions. For the situation s and the background conditions B, $cert(s,B)$ measures the extent to which the background conditions in B are satisfied by the situation s[51]. The value $cert(s,B)$ is used only if the satisfaction of the background conditions is undetermined in the situation.

The quantification of the uncertainty involved is one method of providing a numerical expression of relevance. With Situation Theory, the uncertainty is already represented in the background conditions, though qualitatively. However, the construction of the function $cert$ from the background conditions is not easy. Indeed, what should be the values of $cert(s,B_1)$ and $cert(s,B_2)$ in the following cases:

(i) B_1 and B_2 are the same or disjoint set of types,
(ii) B_1 and B_2 have common types, or
(iii) B_1 is included in B_2?

Some of these questions are considered throughout this chapter.

Attaching uncertainty to relationships (here, through their background conditions) is not a new concept in IR. For example, among synonyms, (in some contexts) two terms can be "more synonymous" than two others. In IR, the computation of the strength of a relationship is often based on statistical/linguistic analysis of text documents or thesauri (see for example [Den64, Rug92]). The result is often viewed as an uncertainty value.

In conclusion, the set of unconditional constraints K_1 and the set of conditional constraints K_2, together with the function $cert : S \times BC \rightarrow [0,1]$, constitute the representation of the knowledge set referred to as K in the Transformation Principle. They constitute the semantic relationships, together with the uncertainty pertaining to them. Note that a constraint in K_2 behaves like a constraint of K_1 if the situation to which the constraint is applied satisfies the background

[51] The correspondence between the qualitative and quantitative representations of the uncertainty of the background conditions has been discussed in [Lal95b].

conditions. The constraint is certain, and delivers implicit and certain information. Otherwise, the constraint is uncertain, delivering uncertain and implicit information, and the value of its uncertainty is given by the function $cert$. In further references, the sets K_1 and K_2, together with $cert$ are assumed defined.

Next, the model of an IR system that accounts for the unstructured representation of documents is presented. The model uses the information stored in K_1 and K_2, and the uncertainty function $cert$.

4.4 The model for unstructured information

A document is an object with an information content; it can be modelled by a situation $d \in S$. A query is an information need; it is then modelled by a type $\varphi \in T$. The role of the IR system (the cognitive agent, if one can say so) is to determine to what extent it can be said that d supports φ. If $d \models \varphi$ then φ is part of the information content of the document; the document is relevant with certainty. Otherwise, constraints from the knowledge set are used to find a flow that leads to that information φ. The uncertainty attached to this flow (if it exists) is used in the computation of the degree of relevance.

The translation of the textual information into types concerns natural language processing, of which there is extensive literature [Win83, Sme92]; the one which is most relevant to this context is Situation Semantics [BP83, FLV87, Bla92]. In this and the next chapters, it is assumed that appropriate tools are available to index documents and queries. That is, it is assumed that the indexing process is done and that the explicit information content of the document and the information need has been determined. A discussion on that matter is given in Chapter 6.

Although the model is described with respect to information items as defined by Situation Theory, it also applies to more complex information structures, such as semantic trees, frames, discourse, etc.

For clarity, the unstructured model is presented in two stages: first, single type queries are considered in the model; second, the model is generalized to queries containing several types.

4.4.1 Single type query

The qualitative components of the unstructured model are based principally on Situation Theory. However, some of the terminology identified in Data Semantics [Lan86] is borrowed because it leads to simpler definitions. The uncertainty is represented by a general uncertainty mechanism.

Let $\Re : S \times T \rightarrow [0,1]$ be the function measuring the relevance degree of a document with respect to a query. For the document modelled by d and the query represented by φ, $\Re(d,\varphi)$ expresses to what extent d is relevant to φ. The computation of $\Re(d,\varphi)$ involves the following cases:

(a) $d \models \varphi$ then the document is relevant, thus $\Re(d,\varphi) = 1$.

(b) $d \models \phi$ and $\phi \rightarrow \varphi \in K_1$. The flow is certain so it relates the same situation; $d \models \varphi$ and $\Re(d,\varphi) = 1$. The information φ is implicit but certain with respect to the document's information content. This case also includes conditional constraints that have their background conditions satisfied by d; that is $\phi \rightarrow \varphi | B \in K_2$ and $d \models B$

(c) $d \models \phi$ and $\phi \rightarrow \varphi|B \in K_2$. The flow is uncertain. It relates two situations; d is transformed into d' such that $d' \models \varphi$. If no other constraints are used to construct d', $\Re(d, \varphi) = cert(d, B)$, indicating that the degree of uncertainty matches the uncertainty attached to the use of the constraint. The formulation captures the fact that the more uncertain is the satisfaction of the background conditions by d, the less relevant the document is to the query.

(d) Several constraints in sequence might be required to arrive at φ. To compute $\Re(d, \varphi)$, the uncertainty has to be propagated. The propagation should reflect the fact that the more transformations are required to obtain φ, the more uncertain is φ. Point (c) above is a special instance of this case.

(e) Several constraints in parallel might lead to φ. To compute $\Re(d, \varphi)$, the uncertainty has to be aggregated, and should reflect that the more transformations lead to φ, the less uncertain is φ.

(f) Any combination of (d) and (e).

(g) Otherwise $\Re(d, \varphi) = 0$, the document is irrelevant to the query.

d is the situation that models the document's initial information content. That is, d supports the information that is explicit, and implicit and certain in the document. Implicit and certain information comes from the application of unconditional constraints and conditional certain constraints to the situation d. The representation of the implicit and uncertain information content of the document must be defined. This representation requires the formal expression of cases (c) to (f) above. All the concepts that are necessary to express (c), (d), (e) and (f) are formally defined in the next sections. The concepts used for expressing (a) and (b) are situations, types, support, and unconditional constraints. These concepts have already be defined. For clarity, in the remainder of this section, the above cases will be referred to by their numbering.

4.4.1.1 Transformation

The transformation of a document (situation) captures the flow of information, and is either an addition or a modification of information. In the first case, the transformed situation also supports the information supported by the initial situation, whereas in the second it does not necessarily. The flow of information links two situations, such the former contains information about the latter. In this thesis, the flow of information is restricted to the phenomenon that leads to the implicit information of the document from its explicit information content, meaning that, with a transformation, additional information is identified as part of the information content (although with uncertainty). Therefore, a transformation, in practice, corresponds to an addition of information, which from now on, will only be considered. The transformation process is referred to as an *extension* process, as the transformed situation supports the information of the latter situation, and more. This decision also leads to a less complex implementation (see Chapter 6). Transformation in general is discussed at the end of this chapter and in Chapter 8.

4.4.1.2 Extension

The concept of extension is defined by both Landman [Lan86] and Barwise [Bar89]. A slightly more restrictive definition is used in this thesis in order to base the extension exclusively on the

application of the conditional constraints.

Suppose that a situation $d \models \phi$ and $\phi \rightarrow \varphi | B \in K_2$. If it is not known whether d satisfies B, d is *extended* to a situation d' such that

(i) $d' \models \varphi$,

(ii) $d' \models B$ (i.e., the background conditions are supported by the extended situation)

(iii) if $d \models \psi$ then $d' \models \psi$ (i.e., every type that d supports is supported by d'),

(iv) if $d' \models \chi$ and $\chi \rightarrow \psi \in K_1$ then $d' \models \psi$ (i.e., all implicit certain information that comes from the fact that $d' \models \chi$ is supported by d'),

(v) if $d' \models \chi$, $\chi \rightarrow \psi | B' \in K_2$ and the background conditions B' are satisfied by d', then $d' \models \psi$ (i.e., the conditional constraint leads to a certain flow).

The situation d' is an *extension* of the situation d. This is denoted $d \triangleright d'$. The extension operator $\triangleright$ is a partially ordered relation on the set of extensions of the situation d. This set is defined as $E(d)$.

An extension models the application of conditional constraints of which the satisfaction of the background conditions is undetermined. The *extended situation* supports the information which was supported by the initial situation and the information that results from the application of the conditional constraints. It also supports the information that results from the application to itself of unconditional constraints and conditional certain constrains. Extensions only partly model case (c), for no uncertainty is yet embodied.

The relationships used in extending documents are assumed appropriate. Obviously, they depend of the domain covered by the documents and the way these are indexed.

A more general type of extension, denoted $d \subseteq d'$, is also defined. $\subseteq$ coincides with Barwise [Bar89] and Landman's [Lan86] definition of extension. This operator satisfies only property (ii) above. $\subseteq$ is used, for example, when new information about a situation can be completely independent to the rest of the information already gained about that situation. This information becomes available not necessarily because of the information that is already supported by the situation, but also because the cognitive agent digitalizes it. Extensions that come exclusively from constraints are modelled by $\triangleright$ which is a special case of $\subseteq$. $\triangleright$ is referred to as the extension operator, whereas $\subseteq$ is referred to as the inclusion operator[52].

4.4.1.3 Sequential extension or branch

A *branch* b is a subset of S with a special situation d, called the *root*, such that

(i) $d \in b$,

(ii) for all $d' \in b$, $d \triangleright d'$, and

(iii) for all $d', d'' \in b$, either $d' \triangleright d''$, $d'' \triangleright d'$ or $d = d'$ (i.e., the situations are the same).

The set $\{d_1, \ldots, d_n\}$ is a branch if there exists an order with respect to $\triangleright$ between the situations in

[52] Barwise also differentiates the two types of relations between situations: a situation is extended to another one (as defined with $\subseteq$) and a situation contains information about another situation (as defined here with $\triangleright$). The latter relation is modelled by the so-called channels discussed in Chapter 2. A model based on these was developed in [vRL96].

that set. For simplicity, if the ordering is $d_i \triangleright d_{i+1}$ for $i = 1, n-1$, the branch is denoted $d_1 \triangleright \ldots \triangleright d_n$.

A *leaf* l of a branch b is the end point situation of that branch; that is, for all $d' \in b$, $d' \triangleright l$.

$B(d)$ is the set of all branches with root d. This set constitutes all the *alternative extensions* of the root situation.

A branch models a sequential transformation of a document; that is part of case (d). The uncertain constraints are responsible for the extensions, thus indicating that branches are themselves uncertain.

4.4.1.4 Uncertainty of a branch

The following function is introduced to measure the uncertainty of the branches of $B(d)$:

$$\partial : B(d) \rightarrow [0, 1]$$

Suppose that $d \models \psi$, and $\{\psi \rightarrow \psi_i | B_i\}_{i=1,n} \subseteq K_2$ are the only uncertain conditional constraints that can be applied to d. Suppose that all the background conditions are incompatible; that is, none of the constraints can be applied together (this is explained in section 4.6.1). Therefore, d can be extended into n *alternative* situations d_i, each from the application of the constraint $\psi \rightarrow \psi_i | B_i$. The uncertainty of each branch $d \triangleright d_i$ is set as

$$\partial(d \triangleright d_i) = cert(d, B_i)$$

for $i = 1, n$. The value $cert(d, B_i)$ represents the uncertainty of d_i being the appropriate extension of d.

Suppose that $\{\phi \rightarrow \phi_i | B_i\}_{i=1,2} \subseteq K_2$, $\left\{\varphi \rightarrow \varphi_j \mid B'_j\right\}_{j=1,3} \subseteq K_2$ and that $d \models \phi$ and $d \models \varphi$. Further, suppose that the satisfaction of the background conditions of these constraints is undetermined. Suppose that all the background conditions B_is are incompatible with each other, and the same applies for the background conditions B'_js. Finally, suppose that each of the B_is can be applied with each of the B'_js. In that case, d can be extended into $2 * 3 = 6$ situations d_{ij} due to $\phi \rightarrow \phi_i | B_i$ and $\varphi \rightarrow \varphi_j | B'_j$ for $i = 1, 2$ and $j = 1, 3$. This is illustrated in the following figure:

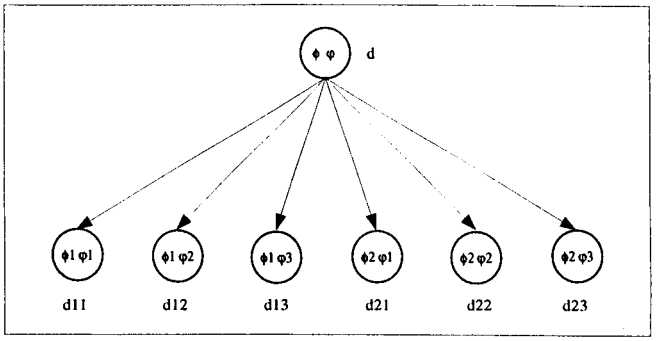


Figure 4.1: Example of the alternative extensions of a situation

The uncertainty attached to the extension d_{ij} is defined as

$$\partial(d \triangleright d_{ij}) = cert(d, B_i) * cert(d, B'_j)$$

This expresses the case of a single extension; that is case (c). Multiplication reflects the fact that the more uncertain constraints that are used, the more uncertain the extended situation. It also indicates that constraints are applied independently. This issue is discussed later in this chapter.

If one of the B_is is the same as one of the B'_js, then the application of the constraints $\phi \rightarrow \phi_i|B_i$ and $\varphi \rightarrow \varphi_j|B'_j$ depends on one set of background conditions B_i $(= B'_j)$. In that case, the uncertainty of the branch extension d_{ij} is defined as

$$\partial(d \triangleright d_{ij}) = cert(d, B_i)$$

Finally, if one of the B_is is incompatible with one of the B'_js, then the application of the constraints $\phi \rightarrow \phi_i|B_i$ and $\varphi \rightarrow \varphi_j|B'_j$ is not possible, and the situation d_{ij} is not constructed.

A branch represents the consecutive addition of uncertain information, which is based on conditional constraints. The more information that is added (i.e., the more extensions are required), the more uncertain the resulting information is with respect to the document's initial information content. Consider the branch $b = d_1 \triangleright \ldots \triangleright d_n$ where $d = d_1$. Let the uncertainty of the extension of d_i into d_{i+1} be $\partial(d_i \triangleright d_{i+1})$. This value depends on the uncertainty of the constraints upon which the extension is based. The uncertainty of the branch b, $\partial(b)$, which can be interpreted as the uncertainty of obtaining the situation d_n from d_1, is defined as

$$\partial(b) = \prod_{i=1,n-1} \partial(d_i \triangleright d_{i+1})$$

This models case (d), that is, the sequential transformation of a document. Many inference processes model the propagation of uncertainty in this manner (see [KC93] for a survey). This formulation and the one used previously are just one way to treat uncertainty. They both satisfy the requirement that uncertainty increases with the number of transformations (extensions) and the number of constraints used in an extension. Both formulations also make computation less complex.

4.4.1.5 Parallel extensions

Suppose that there are m branches b_js that extend the situation d into the same situation d'. These branches constitute parallel extensions of the situation d into the situation d'. The occurrence of parallel extensions indicates that the information supported by d' is less uncertain than if one branch alone was leading to it. It could be said that there are more evidences leading to that situation. Consequently, the uncertainty attached to the obtainment of d' should be higher. Let $\partial(b_j)$ be the uncertainty associated to b_j. The values $\partial(b_j)$s are aggregated into $\partial(d \triangleright d')$ as follows:

$$\partial(d \triangleright d') = \sum_{j=1,m} \partial(b_j)$$

Summation models the accumulation of evidence. A property attached to constraints which is given later ensures that the value of $\partial(d \triangleright d')$ lies in the interval $[0, 1]$.

The representation of parallel transformations of a document and the aggregation of uncertainty pertaining to them constitute case (e) of the evaluation of relevance. Since a branch can be either a single extension or a sequence of extensions, the above formulation captures case (f) of the evaluation of relevance, that is, the combination of sequential and parallel transformations. What remains to be expressed is the relevance degree of the document to a query. This requires the introduction of two additional concepts, which are given in the next two sections.

4.4.1.6 Pertinent situation

For simplicity of expression, it is necessary to distinguish between a document being relevant to a query and the situations involved in modelling that document containing information relevant to the query. The notion of pertinence is introduced for that purpose.

A situation d such that $d \models \varphi$ is said to be *pertinent* to the information φ. Pertinence refers to situations whereas relevance refers to document. A document is (somewhat) relevant to a query φ if the situation modelling that document or at least one of its extension is pertinent to φ.

4.4.1.7 Minimal branch

The Transformation Principle refers to the notion of minimality. Indeed, a document is transformed *until* the information being sought is found. This characteristic is taken into account in the expression of the model with the introduction of *minimal branches.*

Transformations are modelled by branches. The process of extending a branch ceases in two cases:

(i) when the branch leads to a pertinent situation. It is shown later that extending that branch does not modify the degree of relevance with respect to that branch.

(ii) when the branch cannot be extended anymore because either no constraint can be applied to its situation leaf or all the appropriate constraints have already been applied.

Minimal branches are branches of type (i). A branch $b \in B(d)$ is a *minimal branch* with respect to φ, called a $\varphi-$minimal branch, if its leaf is the only situation in that branch that supports φ. $B(d, \varphi)$ is the set of $\varphi-$minimal branches with root d.

In practice, the transformation process does not only cease when information pertinent to the query is found; otherwise, this may be a lengthily process. Techniques that control the transformation process are necessary to allow the system to deliver results in an acceptable amount of time. For example, a maximal number of transformation could be imposed (this techniques is adopted in this thesis — see Chapter 6). Another method would be to ensure that the *quality* of the information contained in the transformed documents is always above a given threshold. The uncertainty propagated along the transformed documents may be used as an indication of this quality; when the uncertainty is below the threshold, the quality of information supported by the transformed situations become non-acceptable, and the transformation process ceases.

4.4.1.8 Relevance degree

The computation of the relevance degree of a document to a query involves several cases as listed in the beginning of section 4.4.1. All the concepts necessary to express those cases have been defined: a transformation and its uncertainty, sequential transformations and the propagation of the uncertainty, parallel transformations and the aggregation of the uncertainty, and minimal branches. The latter model the minimal transformations of the document into fictitious documents (situations) that contain the information being sought (which was phrased in a query). The aggregation of these minimal branches can express the relevance degree. That is, given the situation d which represents the document's initial information content and the type φ which represents the information need,

the value of $\Re(d,\varphi)$ depends on the uncertainty of the $\varphi-$minimal branches in $B(d,\varphi)$. I pose

$$\Re(d,\varphi) = \sum_{b \in B(d,\varphi)} \partial(b)$$

This formulation captures the fact that the bigger the set $B(d,\varphi)$ (i.e., the more minimal extensions lead to pertinent situations), the higher the degree of relevance. The use of summation embodies the case of parallel extensions and it makes the generalization into the structured model possible. Some would argue that this combination behaves as if independence was assumed; this aspect is discussed at the end of this chapter[53].

The formulation of $\Re(d,\varphi)$ does not always yield a value between 0 and 1. One way to obtain this is to normalize the constraints.

4.4.1.9 Normalization

To ensure that $\Re(d,\varphi) \leq 1$, a normalization process is performed on the background conditions that may or may not be satisfied by a situation d. Suppose that $d \models \psi$. Given a set of constraints $\{\psi \rightarrow \psi_i | B_i\}_{i=1,n} \subseteq K_2$, the normalization is expressed as follows:

$$\sum_{i=1}^{n} cert(d, B_i) = 1$$

if the background conditions of these constraints are all incompatible with each other. Otherwise, the normalization process is done with respect to the set of mutually incompatible background conditions.

The set $\{\psi \rightarrow \psi_i | B_i\}_{i=1,n}$ can be viewed as the set of exhaustive constraints with respect to ψ[54]. Since those constraints are uncertain, the set $\{\psi_1, \ldots, \psi_n\}$ can be interpreted as forming a set of exclusive choices with respect to ψ, thus somewhat modelling imprecision. Therefore, the normalization process is not counter-intuitive, though the normalization process does not taken into account the eventual relationships between sets of background conditions. This issue is discussed at the end of this chapter.

To avoid normalizing the constraints, an alternative formulation could be

$$\Re(d,\varphi) = \frac{\sum_{b \in B(d,\varphi)} \partial(b)}{\sum_{b \in B(d)} \partial(b)}$$

This formulation consists of normalizing the overall result. In the structured model described in the next chapter, the extension process is performed layer by layer (the reasons are explained in that

[53] An entropy-like formulation could have been used. The branches would be the possible extensions of a document and $\partial(b)$ their probability of occurrence. In such case, the amount of information generated by the document with respect to the type φ is

$$\sum_{b \in B(d,\varphi)} -\partial(b) * \log(\partial(b))$$

This formulation is not used; first, ∂ is not a probability function; and second, the model is later expanded to incorporate the significance of information, in which this formulation cannot be merged.

[54] Barwise, Devlin or Huibers would object to this since it is not possible to be aware of a set of exhaustive constraints. However, in IR all identified relationships are stored. Although they are not exhaustive with respect to the outside world, they are exhaustive with respect to the IR system.

chapter). In that instance, if the situation d is extended into n situations $d_1, \ldots, d_n$, it is required that the summation of the uncertainty attached to the extended situations $d_1, \ldots, d_n$ equates the uncertainty associated with d. The above formulation cannot satisfy this requirement because the normalization is done in the overall result.

Another formulation could be to normalize the uncertainty of the newly extended situations, layer by layer. In that case, the uncertainty of a situation decreases/increases if the number of situations in that layer is high/low. It is shown in the next chapter that, in the structured model, the extension of a situation in a layer does not depend on the extensions of the other situations of that same layer. The same applies for the uncertainty attached to these extensions. In particular, a situation that cannot be extended is put as such at the next layer, and retains its uncertainty, which would be changed if normalized. Therefore, the second formulation is also inappropriate. Furthermore, it is known that normalizing should be performed once and not repeatedly. Since an overall normalization is not adequate, the best approach is to normalize the background conditions involved in the extension of a situation.

In the remainder of this chapter, the background conditions used in the extension of a situation are assumed normalized. The corresponding constraints are referred to as normalized.

4.4.1.10 Properties of the formulation of the relevance degree

A formulation of the relevance degree has been defined. Questions may be raised regarding the properties that derive from this formulation:

(i) Does the formulation capture minimality?
(ii) Does it satisfy the basic case where a document is relevant to the query?
(iii) Can it express exhaustivity and specificity?

These questions are examined in turn in this section.

The evaluation of $\Re(d, \varphi)$ includes all the minimal branches in $B(d)$ that lead to pertinent situations; that is, $B(d, \varphi)$. Assume that this set was not defined. Let d' be a pertinent situation such that $d \triangleright d' \in B(d)$. Suppose that d' can be extended into precisely two situations, d'_1 and d'_2. These are also pertinent, so both $\partial(d \triangleright d' \triangleright d'_1)$ and $\partial(d \triangleright d' \triangleright d'_2)$ are included in $\Re(d, \varphi)$. The normalization of the constraints makes $\partial(d \triangleright d' \triangleright d'_1) + \partial(d \triangleright d' \triangleright d'_2) = \partial(d \triangleright d')$, implying that extending a pertinent situation does not affect the value of the relevance degree. This explains why minimal branches capture minimality.

Consider the simple case where the query is φ and that $d \models \varphi$. In that case, $\Re(d, \varphi) = 1$. This can be captured with the reflexive property of extension ($\triangleright$ being a partial order); that is, $d \triangleright d$. If I pose $\partial(d \triangleright d) = 1$, then the formulation of $\Re$ satisfies the case of a document that is modelled by a situation pertinent to the query.

Two situations can both be pertinent to a query, but one of them can support more irrelevant information to the query than does the other. The latter situation is more specific to the query than the former. This cannot be reflected in the formulation of $\Re(d, \varphi)$. In the structured model, where semantics are attached to situations and their extensions, this is shown to be different. This is discussed in the next chapter. Since this section considers the single-type query, the question of the exhaustivity of the document to the query is irrelevant.

4.4.1.11 Summary

Transformation corresponds to an extension process. A document's initial information content is modelled by a situation d. The flow of information is modelled by sequential and/or parallel extensions of that situation, which constitute branches. Uncertainty is propagated along the sequential extensions and aggregated along the parallel extensions. Branches then become quantified with uncertainty values. A branch is minimal to a query if its leaf is the only situation in that branch that supports the information being sought in that query. The uncertainty values of the minimal branches whose leafs are pertinent to the query are aggregated to compute the degree of relevance. This constitutes the model for unstructured information and single type query.

4.4.2 Complex query

Often, the information need is complex, leading to queries composed of more than one item of information. These queries are called complex queries. The model described in the previous section is expanded to accommodate complex queries. The extension process is the same as previously defined. Those items that need redefining are the representation of queries, and the pertinence of situations since a set of types may be involved. The formulation of the relevance degree is re-expressed based on these new definitions.

4.4.2.1 Representation of complex queries

Depending on the ontology adopted, a complex query can be modelled by a set of types or a single type. Assume that the conversion of the information need into infons has been done and that the query is translated into n infons σ_i, for $i = 1, n$. The representation of the query can be the type $\varphi = [\dot{s}|\dot{s} \models \{\sigma_1, \ldots, \sigma_n\}]$ or the set of types $\{\varphi_i = [\dot{s}|\dot{s} \models \sigma_i]\}_{i=1,n}$, for $i = 1, n$. However, extensions often do not lead to situations of type φ. Instead, they more often lead to situations that support some of the φ_is, thus showing a partial relevance to the query. Therefore, the second representation is best; that is, a complex query is modelled as the set $\Phi = \{\varphi_1, \ldots, \varphi_n\}$. The single type query φ could have been used if an additional operator was introduced to represent that a situation is partly of a given type. However, there are already enough operators.

Barwise [Bar89] accommodates the support relation to include a set of types. In this case, a situation d supports $\Phi \subseteq T$, written $d \models \Phi$, if and only if, $s \models \varphi$ for all $\varphi \in \Phi$.

4.4.2.2 Pertinent situations and minimal branches

The extension of the situation d does not always generate situations that support all the types in Φ. To represent partial relevance, any situation that supports at least one type in Φ is considered pertinent to Φ.

The $\Phi-$minimal branches are defined as the set of the branches in $B(d)$ whose leaf situations and no other situation in those branches are pertinent to Φ. That is, b is a $\Phi-$minimal branch if the leaf of that branch b, and no other situation in b, supports at least one type in the set Φ. This set is denoted $B(d, \Phi)$.

4.4.2.3 Relevance degree

The formulation of the relevance degree is redefined on the set of types $\Re : S \times 2^T \rightarrow [0,1]$ as follows:

$$\Re(d,\Phi) = \sum_{b \in B(d,\Phi)} \partial(b)$$

This formulation is the same as that used for single-type queries, but it applies to complex queries. Partial relevance is captured because any situation that supports part of the query is considered pertinent.

4.4.2.4 Properties of the formulation of the relevance degree

As for the formulation of the relevance degree for single-type query, the expression of the relevance for a complex query cannot express the extent to which a document is specific to a query because a pertinent situation can support information that does not concern the query. As it will be shown in the next chapter, if the information is organized into semantic entities (for example, a group of synonymous terms), specificity can be captured in the evaluation of $\Re(d,\Phi)$. What can be concluded thus far is that if there are branches whose leaf situation does not contain any information relevant to the query, the document is not specific to the query.

The exhaustivity of the document cannot be expressed with the formulation of the relevance degree proposed in 4.4.2.3. Indeed, if all extensions of the situation that models a document leads to pertinent situations, the relevance degree of the document to the query will be of value 1. However, this does not imply that all the information requested is indeed part of the document, either explicitly or implicitly; that is, the document may still not be exhaustive in relation to the information being sought. In the next chapter, it is shown that by allowing extensions to be maximally extended (until they cannot be further extended), the exhaustivity of the document is represented.

It may be that the consequent of a constraint which can be applied to a situation is a type already supported by that situation. Let s be a situation which supports two types ϕ and χ (i.e., $s \models \{\phi, \chi\}$). Several cases occur:

(i) $\phi \rightarrow \chi \in K_1$; the application of the unconditional constraint has no effect

(ii) $\phi \rightarrow \chi | B \in K_2$ and s satisfies the background conditions of the constraint ($s \models B$); the application of the constraint has no effect.

(iii) $\phi \rightarrow \chi | B \in K_2$ and s does not satisfy the background conditions of the constraint ($s \not\models B$); the constraint cannot be applied to the situation s.

(iv) $\phi \rightarrow \chi | B \in K_2$ and it is not known whether s satisfies the background conditions of the constraint; the constraint is not applied, for it does not bring any additional information. However, the uncertainty associated to the constraint is not divided between the uncertainty associated to other uncertain condition constraints with antecedent ϕ. This is explained below.

(v) ϕ and χ are not related: all constraints with antecedents ϕ or χ can be used. The only restriction depends on the satisfaction of their background conditions.

In case (iv), suppose that the following two constraints $\phi \rightarrow \chi_1|B_1$ and $\phi \rightarrow \chi_2|B_2$ are uncertain with respect to the situation s and that B, B_1 and B_2 are mutually incompatible. If s did not support χ, the situation s would have been extended as follows[55]:

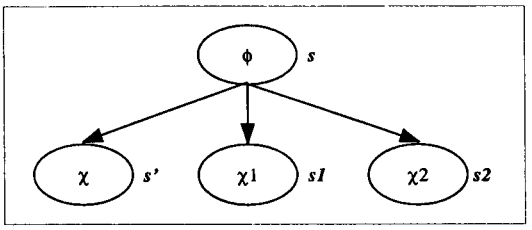


Figure 4.2: Case of extension that brings additional information

In the above case, normalization means that $cert(s, B) + cert(s, B_1) + cert(s, B_2) = 1$.

The fact that $s \models \chi$ implies that $s' = s$. The situation s is extended into two situations, as shown in the following schema:

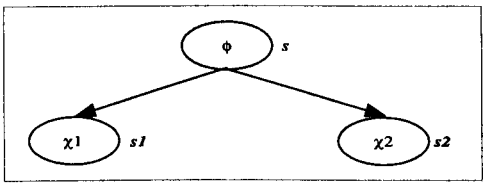


Figure 4.3: Example of an extension that does not bring additional information

As discussed in section 4.4.1.9, the constraints that lead to alternative extensions of a situation must be normalized. In the above example, this means that $cert(s, B_1) + cert(s, B_2) = 1$. However, although s is extended to two situations, there are indeed three possible ways to extend s, one being to itself. To represent this fact, the uncertainty values attached to $\phi \rightarrow \chi_1|B_1$ and $\phi \rightarrow \chi_2|B_2$ remains the same and $cert(s, B) + cert(s, B_1) + cert(s, B_2) = 1$ still holds.

4.5 Example

Let a set of types be defined by $T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}, t_{11}, t_{12}\}$. Let the set of unconditional constraints and the set of conditional constraints be given, respectively, by

$$K_1 = \{t_5 \rightarrow t_{10}, t_6 \rightarrow t_8, t_8 \rightarrow t_9\}$$
$$K_2 = \{t_1 \rightarrow t_2, t_1 \rightarrow t_3, t_2 \rightarrow t_4, t_2 \rightarrow t_5, t_2 \rightarrow t_6, t_4 \rightarrow t_{11}, t_4 \rightarrow t_{12}\}$$

For simplicity, the background conditions of the conditional constraints is not shown, but the uncertainty values associated to their satisfaction are shown in the figure below. Let d be the situation that models the document's initial information content. Suppose that the types t_1 and t_6 represent explicit information in the document. Let the query be represented by the set of types $\{t_5\}$. The extensions of d, together with the propagation and the aggregation of the uncertainty, are shown in the following figure:

[55] In this example, it is assumed that no other constraint is used to extend the situation s.

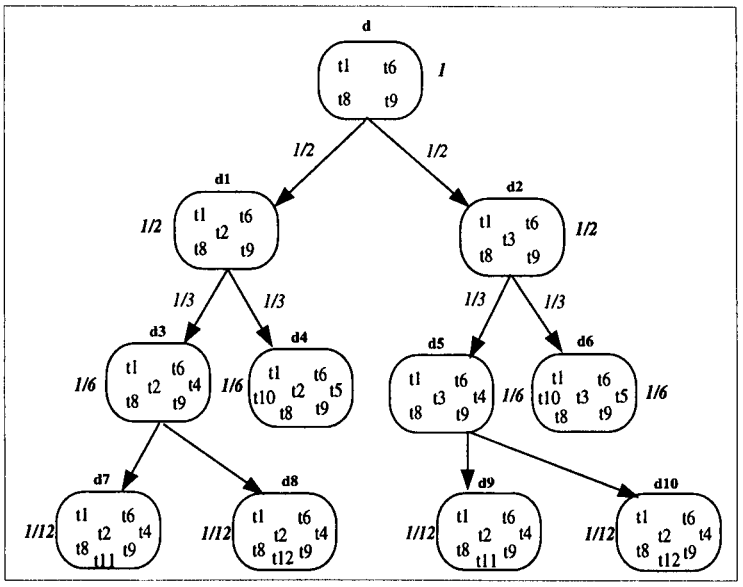


Figure 4.4: Example of the computation of the relevance in the unstructured model

Two minimal branches are obtained; one leading to the leaf situation d_4, and the other leading to the leaf situation d_6. These minimal branches are $d \triangleright d_1 \triangleright d_4$ and $d \triangleright d_2 \triangleright d_6$. The branches leading to d_7, d_8, d_9 and d_{10} cannot be further extended. The uncertainty values attached to the minimal branches are both equal to $1 * 1/2 * 1/3 = 1/6$. The relevance degree of the document to the query is $1/6 + 1/6 = 1/3$.

4.6 Discussion

Three issues raised in this chapter are discussed in this section: the relationships between the background conditions (section 4.6.1), the assumption of independent information (section 4.6.2) and the generalization of the unstructured model to one that deals with transformation in general (section 4.6.3).

4.6.1 Background Conditions

The relationship between the background conditions of constraints have not been considered in the extension of a situation. Indeed, let $\phi \rightarrow \phi_1 | B_1$ and $\phi \rightarrow \phi_2 | B_2$ be two constraints that can be applied with uncertainty to a situation s. The backgrounds conditions of these two constraints may be related. Indeed, there are four possible relationships between B_1 and B_2:

(i) $B_1 = B_2$; the two sets of background conditions are identical.

(ii) $B_1 \perp B_2$; the two sets of background conditions are incompatible.

(iii) The two sets of background conditions B_1 and B_2 are independent of each other. That is, the fact that a situation satisfies the background conditions B_1 has no effect on whether the situation satisfies the background conditions B_2

(iv) The two set of background conditions B_1 and B_2 are dependent.

Case (i) is already considered in the model. The application of the two constraints leads to one situation s' and the uncertainty associated with this extension is

$$\partial(s \triangleright s') = cert(s, B_1)$$

In case (ii), the application of the two constraints definitely leads to alternative situations, since the background conditions of these constraints are incompatible. In the last two cases, (iii) and (iv), the application of the two constraints leads to one situation s', because the background conditions are compatible. The uncertainty associated to the extension is

$$\partial(s \triangleright s') = cert(s, B_1) * cert(s, B_2)$$

This expression does not differentiate the two cases. In the context of probability theory, it assumes independence of knowledge, which is not always the case. The representation of dependent background conditions is not captured by the model. The same problem arises with respect to the relationships between background conditions of the two constraints $\phi_1 \rightarrow \varphi_1|B_1$ and $\phi_2 \rightarrow \varphi_2|B_2$. The model ignores the fact that the two set of background conditions may be dependent.

The unstructured model must be enhanced to resolve the problem mentioned above. These are known problems in the world of uncertainty theory [KC93]. The determination of a more accurate manipulation of the uncertainty of the background conditions will be the object of further research. In the implementation of the unstructured model (Chapter 6), the independent manipulation of the background conditions is, however, appropriate.

4.6.2 Modelling of the uncertainty

Besides the independent manipulation of the background conditions, independence is also assumed at other levels. For example, the formulation of the propagation of the uncertainty is

$$\partial(s \triangleright s') = cert(s, B_1) * cert(s, B_2)$$

where $\phi \rightarrow \phi_1|B_1$ and $\varphi \rightarrow \varphi_1|B_2$ are two conditional constraints that can be applied with uncertainty to a situation s, and s' is the extended situation. Such a formulation does not take into account the fact that ϕ and φ may be related (for example, they define a constraint).

In the formulation of the aggregation of the uncertainty, it is not expressed that two situations extended into one situation may share common information. The two situations are treated independently, as are their uncertainty values. More adequate formulations of both the propagation and the aggregation of uncertainty should be investigated to capture dependent knowledge.

4.6.3 From addition to transformation

The unstructured model may be extended to deal with other types of transformation than addition; that is, modification and deletion of information. Some indications towards this direction are briefly discussed in this section.

To represent the modification of information, it is necessary to model the transformation of a document on a basis other than the extension of that document. An approach was suggested in

[vRL96] with the use of channels instead of extensions. The concept of channels was introduced in [Bar92] to model the systematic link between situations. A channel carries the flow of information between two situations. The two situations support information which is determined by the flow of information carried in the channel. One of the situations can be viewed as the transformation of the other situation. The use of channels to model that transformation of a document was discussed in Chapter 2.

As discussed in Chapter 1, the deletion of information needs outside intervention, for example, from a user. A situation may represent a state of the IR system (obtained by the flow of information). The deletion of information could reflect a change in a user's beliefs. In that case, the IR system must go back to a state (a situation) that is compatible with that user's beliefs. As discussed in Chapter 2, a belief system seems the appropriate framework to model this type of system. There is some current work on the expression of a default logic (which is one example of a belief system) within Situation theory [Cav93], thus indicating that the deletion of information may be represented within Situation Theory.

4.7 Conclusion

A model based on the Transformation Principle has been presented. This model, called the unstructured model, accounts for an unstructured representation of a document. The qualitative components of the model are represented with Situation Theory [Bar89, Dev91, BE87, BE90, Fer90, Mos91] and the quantitative components are represented with a general uncertainty mechanism. Although transformation was restricted to an addition of information (an extension process), the model can be easily extended to include modification of information.

The use of Situation Theory provides an appropriate representation of information and its flow, explicit and implicit information, the partiality of information, intensionality, and so forth. Another important advantage in using Situation Theory is that the natural language processing can be performed, or at least formally modelled, with Situation Semantics [BP83]; thus leading to an uniform framework that deals with situations, types and constraints.

The uncertainty mechanism adopted allows the generalization of the unstructured model to a formalism that accounts for a structured representation of a document. This model is described in the next chapter.

Chapter 5
Description of the Model for a Structured Representation of Information

5.1 Introduction

The information contained in a document is often structured. For example, a document may consist of a title, a set of authors, an abstract, a text, figures and tables. A multimedia document may contain a mixture of text, image, and video. The structure of a document can also be implicit. For example, a structure may consist of the information (e.g., terms) contained in the document, which defines a document topic. Such types of structures are based on semantics because they take into account the fact that information can be semantically related. For reasons of simplicity, only semantic-based structures are considered in this thesis. A model that takes into account this type of structures is proposed in this chapter. The model is a generalization of the unstructured model developed in the previous chapter.

5.2 Semantic-based structures

Information that is part of a document's information content can be *semantically* related. An example of semantically related information is equivalent pieces of information. For example, the representation of a document's information content could be a set of terms. In a document, many terms can be used to refer to the same person or object. For instance, "the Canadian Prime Minister" and "J. Chretien" are two *equivalent* terms[56]. A document should not be more relevant to a query that uses "the Canadian Prime Minister" and "J. Chretien", than to a query that uses only one of these terms, because the information need is the same. Indeed, the first query uses two different terms to refer to the same piece of information (here a person), whereas the second query uses one term only. This equivalence of information can be taken into account by grouping equivalent terms together, and treating the groups of equivalent terms as entities. The comparison between the information sought by the query and the information contained in the document will then consist of matching the terms used in the query to the groups of equivalent terms that compose the document.

[56] As described in Chapter 2, these two terms are intensional because they do not refer to the same person in every context. This issue is ignored in that example.

Another example of semantically related information is that which is *nested* within other items of information. This can be demonstrated by the following documents (for simplicity, the explicit information content of a document is represented by a set of terms):

$$d_1 = \{dog, animal\}$$
$$d_2 = \{dog, animal, cat\}$$

Let $q = animal$ be a query. The meaning of the term "animal" is nested within the meaning of both "dog" and "cat" because dogs and cats are animals. The document d_2 is more relevant to the query q that is the document d_1. Indeed, d_2 is concerned with both dogs and cats, which are animals, whereas d_1 is only concerned with dogs. This observation can be reflected if the following representations of d_1 and d_2 are adopted:

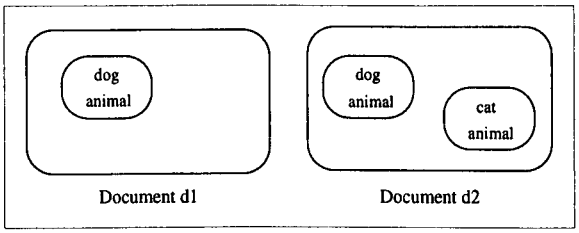


Figure 5.1: Example of a structured representation of a document

Here, two groups of terms are defined, one related to "dog" and the other related to "cat". The term "animal" is part of each group because "animal" is nested in both "dog" and "cat". If the two groups of terms are treated as entities, it is possible to assert that the document d_2 is more relevant to the query q than is d_1. The reason is that d_2 contains two entities that concern the query, whereas d_1 contains only one.

The preceding examples show that the representation of the information content of a document should take into account the fact that information can be semantically related. Such a representation is possible by structuring the information content of the document into a set of situations, as defined in Situation Theory, each of them supporting semantically related information. This approach leads to a semantic-based structured representation of the information content of a document.

Two advantages result from this representation. First, the fact that the information content of a document may be inconsistent can be represented clearly. For example, a document may report opposing views of a topic which, when translated into types, lead to contradictory types. If the document is initially modelled by a single situation, then this situation can be inconsistent[57]. If several situations model that document, this problem can be avoided. Second, the specificity of the document to the query can be captured. For example, if terms are grouped into situations according to their equivalent meaning, a situation can be viewed as delimiting one of the subjects covered by the document. The more situations that are formed, the less specific the document is to the subject defined by each situation. Similarly, the less situations that are formed, the more specific the document is to each subject defined by each situation.

[57] A situation is inconsistent if it supports contradictory information, for example, "the hat is red" and "the hat is not red", or "the hat is red" and "the hat is blue".

In this chapter, a model of an IR system is advanced, based on the Transformation Principle, and which accounts for a semantic-based structured representation of a document's information content. The model is called the *structured model.*

5.3 The components of the structured model

An IR model based on the Transformation Principle possesses both qualitative and quantitative components. In the unstructured model, the qualitative components were expressed according to Situation Theory. A situation, for instance d, models the document's initial information content, that is, the explicit, and the implicit and certain information content of the document. The flow of information transforms this situation into fictitious situations which capture the implicit and uncertain information content. The transformation is defined as an extension process, and the nature of the flow is determined by the constraints used to perform the extension. This is illustrated in the figure below:

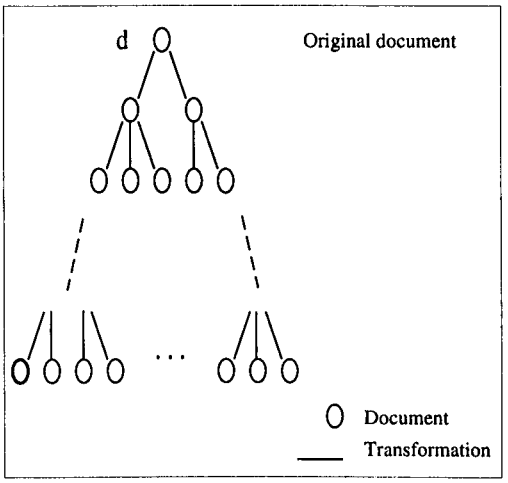


Figure 5.2: Transformation of a document in the unstructured model

The concepts defined to model the qualitative components in the unstructured model can be expanded to provide for a semantic-based structured representation of the document. This is illustrated in Figure 5.3.

In the structured model, the information content of the document is modelled as a set of situations, for instance D. Let s be a situation in D, and suppose that the information supported by s is semantically related. The flow of information may extend the situation s into a set of fictitious situations which support both the information supported by s and the information that is derived from the flow. The information supported by any of these fictitious situations is also semantically related. The reason is that the situation s originally supports semantically related information, and that situations are extended from the application of constraints, which symbolize semantic relationships. The same observation applies for all the situations in D with respect to their extended situations. Therefore, if the information in each situation of D is semantically related, then the information supported by any of the fictitious situations that result from the extension of that situation is also semantically related. If the situations in D initially form a semantic-based

structured representation of a document's initial information content, then the extensions of these situations also form a semantic-based structured representation of the fictitious document, that is, the document's implicit information content.

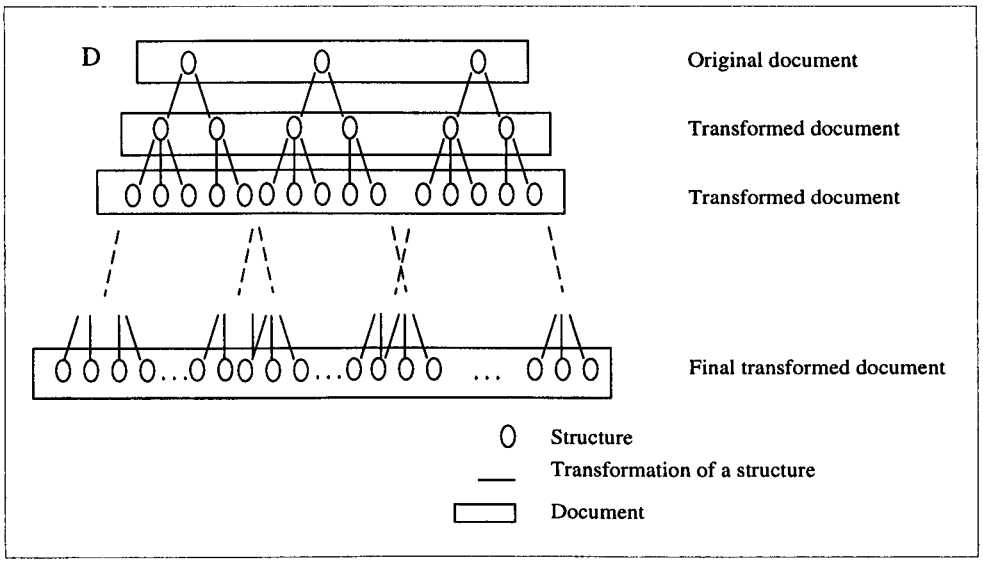


Figure 5.3: Transformation of a document in the structured model

The expressions of the qualitative components in the structured model require the definitions of the following concepts:

(i) a situation that supports semantically related information. Such a situation is called a *basic situation.*

(ii) a semantic-based representation of the document's information content as a *set of basic situations.*

(iii) the transformation (extension) process applied to a *set of basic situations* and yielding a *set of basic situations.*

The expression of these qualitative components uses the concepts defined in the unstructured model. Two qualitative components not mentioned previously are representations of the query and the knowledge set. In the structured model, only the document's information content is structured. Therefore, the representation of queries is the same as for the unstructured model; a query is represented as a set of types. The knowledge captures semantic relationships which are the same for the structured and the unstructured model. Thus, the knowledge set is the same as for the unstructured model; it is modelled as a set of unconditional and conditional constraints.

An IR model based on the Transformation Principle also possesses quantitative components. The representation of the quantitative components in the unstructured model is based on a general uncertainty mechanism. The representation of these components must be re-defined to accommodate a semantic-based structured representation of documents. These components are

(iv) propagation and aggregation of the uncertainty.

(v) expression of the degree of relevance of the document to the query.

As explained in section 1, one advantage in formalizing a semantic-based structured representation of a document's information content is to capture specificity. This is possible by assigning weights to groups of semantically related information items. This means that a weight is associated to each basic situation, thus reflecting its significance with respect to the document's overall information content. Therefore, the definition of the following concept is also required:

(vi) the significance of a basic situation.

The expression of the structured model requires the definitions of the items (i) to (vi) listed above. The items (i) to (iii) correspond to the qualitative components of the model, whereas items (iv) to (vi) correspond to its quantitative components. Unless otherwise stated, the concepts defined in the unstructured model remain the same in the present chapter. First, basic situations are formally defined within Situation Theory ontology.

5.4 Basic situations

A basic situation is a situation that supports semantically related information. The definition of basic situation is based on Situation Semantics [BP83, Coo] and some of the notions defined by Drestke [Dre81]. Note that the former framework is based on Situation Theory and is used for natural language processing, and that the latter framework presents many of the foundations of Situation Theory.

According to Situation Semantics, the representation of a sentence[58] involves two entities:

(i) a situation s that is described by the sentence, and
(ii) a type φ that represents its information content.

The representation of this sentence is denoted $s \models \varphi$ and is referred to as the *propositional content* of the sentence.

According to Drestke [Dre81], the cognitive activity that leads to the knowledge of an agent results in *intentional mental states* [Cum89, Zal88][59]. Indeed, the propositional content of an agent's knowledge exhibits intentional characteristics because this knowledge must be distinguished even when it involves a number of inter-dependent propositional contents. Moreover, the agent can know that $s \models \varphi$ but not that $s \models \varphi'$ (or $s' \models \varphi'$) although $s \models \varphi$ could imply $s \models \varphi'$ (or $s' \models \varphi'$). Meaning also exhibits intentionality; a sentence has a specific meaning, although it might carry information that goes beyond that meaning.

Propositional contents that exhibit intentionality are called *semantic contents*[60]. According to Drestke, a document's sentence is not intentional with respect to its propositional content. Although a sentence has, for instance, the propositional content $s \models \varphi$, the knowledge has this as its exclusive content (which then constitutes a semantic content). This not the case for the sentence. Indeed, a sentence has a meaning which usually corresponds to its propositional content, but it also carries

[58] Although sentence is mentioned, the discussion applies to any other type of syntactic structure, or a set of them.
[59] In [Dev91] and [BP83], the adequacy of Situation Theory to model intentional states, such as knowledge, meaning or belief is thoroughly demonstrated.
[60] In reality Drestke defines three levels of intentionality. These are ignored for simplicity sake. Here it is assumed that a propositional content either exhibits intentionality or does not.

additional information which generates further propositional contents. As a result, Dretske proposes a refined definition of semantic content for sentences. A sentence has the propositional content $s \models \varphi$ as its semantic content if this information is carried in digital form by the expression of that sentence. If $s \models \varphi'$ (or $s' \models \varphi'$) is nested[61] within $s \models \varphi$, then both constitute propositional contents of the sentence, but only $s \models \varphi$ corresponds to its semantic content.

The expression of a sentence may carry several propositional contents in digital form, but not all of them constitute a semantic content. For example, if the expression of a sentence carries the information "the object is square" (i.e., $s \models [\dot{s} \models\ll Shape, \dot{x}, square; 1 \gg]$) and the information "the object is rectangular" (i.e., $s \models [\dot{s} \models\ll Shape, \dot{x}, rectangular; 1 \gg]$), only the former propositional content leads to a semantic content, because "being square" implies "being rectangular" (the latter is nested within the former). Therefore, defining the semantic content as a propositional content carried in digital form is still inappropriate. Consequently, Dretske states that the propositional content $s \models \varphi$ constitutes a semantic content of a sentence if and only if

(i) indeed $s \models \varphi$, and

(ii) the sentence has no other propositional content $s \models \varphi'$ (or $s' \models \varphi'$) such that $s \models \varphi$ is nested into it.

Semantic contents are used as the basis of the definition of basic situations. The analysis of document's sentences results in a collection of propositional contents. Those not nested in other propositional contents (whether or not they are carried in digital form), constitute the semantic content of the document. For example, let s be a situation such that $s \models cat$, $s \models dog$ and $s \models animal$[62]. The propositional contents $s \models cat$ and $s \models dog$ constitute semantic contents because "dog" is not nested within "cat" and vice versa. The propositional content $s \models animal$ does not determine a semantic content because $s \models animal$ is nested within both $s \models cat$ and $s \models dog$.

The analysis of the document's sentences may lead to a number of propositional contents which involve a number of situations. For example, let a document be about the flooding which occurred in the United States in 1994 and in Europe in 1995. Two situations are described by this document, one referring to the United States and the other referring to Europe. If these two situations correspond to the real situations related to flooding, then the information supported by these situations extends beyond the information content of the document. This information may not be identifiable from the document's information content, and is thus difficult, if not impossible, to capture. These kinds of situations are not considered in this thesis. Only situations that result from a semantic-based analysis of document's sentences are considered. These are referred to as semantic-based situations, and support types that represent information explicitly or implicitly contained in the document's information content (the latter comes from the application of certain constraints).

To enable the expression of the specificity of a document to a query, semantic contents are restricted as follows:

> If $s \models \varphi$ constitutes a semantic content of the document, then no other type constitutes a semantic content with respect to s.

[61] That is, the proposition content $s \models \varphi'$ (or $s' \models \varphi'$) comes from the fact that $s \models \varphi$.
[62] For simplicity, types are directly represented as terms.

That is, if $s \models \varphi'$ constitutes a propositional content of a document, then the information represented by the type φ' is explicitly or implicitly contained in the document's information content, and it is semantically related to the information represented by the type φ. The latter means that there exists a constraint $\varphi \rightarrow \varphi'$ or a set of constraints that lead φ to φ' that can be applied to s (i.e., the constraints are either unconditional, or certain and conditional with respect to s). Otherwise, two different situations are defined, leading to two propositional contents $s \models \varphi$ and $s' \models \varphi'$.

To recap: the semantic-based analysis of document's sentences yields a set of semantic contents, which involve a set of situations such that each of these situations has a unique semantic content and supports semantically related information. These situations constitute the *basic situations* of the document.

In the previous example of dogs and cats, two basic situations are defined, one related to "dog' and the other related to "cat". Let s' and s'' be these situations, the semantic contents of which are $s' \models dog$ and $s'' \models cat$. Since dogs and cats are both animals, $s' \models animal$ and $s'' \models animal$. Both s' and s'' constitute basic situations because they both have a unique semantic content, $s' \models dog$ and $s'' \models cat$, and because each of them supports semantically related information.

The basic situations are the basis of a model of an IR system that accounts for a semantic-based structured representation of documents. Situation Theory does not provide a concept for a set of situations, although such a concept can be defined within the theory. However, recent work shows the analogy between Situation Theory and the framework of Scott Domains [Sco82] from which the concept of a set of situations can be derived.

5.5 Scott Domains for Information Retrieval

Many concepts defined in Scott Domains present an overall similar behavior to those necessary to construct the structured model. Indeed, [Bar91] and [Sel90] have demonstrated the analogy between Scott Domains and Situation Theory[63]. They also show the superiority of Situation Theory to Scott Domains for an information theory perspective. Neither the proofs nor the arguments are given here since they are complex and unnecessary in understanding the model. Only the terminology related to the development of the structured model is listed. Scott [Scott82] states:

> "Intuitively, an information system is a set of propositions that can be made about possible elements of the desired domain [...]; as a consequence, an element can be constructed abstractly as the set of all propositions that are true of it. Partial elements have small sets; while total elements have large sets ..."

The concept of *element* can be compared to the notion of situation. This is shown with the (incomplete and simplified) definitions of *information system* and *domain* given below.

An information system is a tuple $A = \langle D_A, \vdash_A \rangle$ where D_A is a set of *propositions* and $\vdash_A$ is an *entailment relation* defined on $2^{D_A} \rightarrow 2^{D_A}$. There are properties between D_A and $\vdash_A$ which represent typical properties of an entailment relation. An element of an information system

[63] The comparison is between Scott Domains and Channel Theory, an extension of Situation Theory. However, for the purpose of the discussion, the reference to Situation Theory is sufficient.

$A = \langle D_A, \vdash_A \rangle$ is any set of propositions x such that

(i) all subsets of x are consistent, and
(ii) x is closed under entailment.

Given an element x and a proposition X, $X \in x$ indicates that the property described by X is true of x. An element is an intentional object that is described by some propositions. Situations can be compared to elements. Indeed, $X \in x$ can be rewritten in Situation Theory as $x \models X$, where x and X can be viewed as a situation and a type, respectively, such that the former supports the latter. The closure under entailment in (ii) means that whatever can be entailed from the initial properties of an element also constitute a property of the element. A property true of x entails other propositions which describe properties that are also true of x. This can be compared to the application of the certain constraints in Situation Theory. That is, $x \models X$, and $X \vdash_A Y$ (X entails Y) which can be rewritten in Situation Theory ontology as the constraint $X \rightarrow Y$ implies $x \models Y$. This property is intrinsic to the nature of situations and certain constraints.

The set of elements in an information system $A = \langle D_A, \vdash_A \rangle$ constitutes a *domain*, denoted $|A|$. In the next section, a domain is re-defined as a set of situations instead of elements. These situations correspond to the basic situations that are identified from the semantic-based analysis of the text document. This new definition of a domain is used to model a structured representation of a document. Only the explicit and the implicit and certain information of the document is embodied in the domain.

The encapsulation of the implicit information, which results from the flow of information, is possible with the notion of *approximate mapping*. Two information systems $A = \langle D_A, \vdash_A \rangle$ and $B = \langle D_B, \vdash_B \rangle$ can be mapped together by an approximate mapping, which is a binary relation $f : A \rightarrow B$ between consistent propositions of D_A and D_B. Some properties of f are too restrictive for an information theory perspective, but f is still comparable to extensions which model the flow of information in this thesis. Indeed, the approximate mapping f can be defined so that the image of the elements that constitute the domain $|A|$ yields the elements of the domain $|B|$. That is, if domains are re-defined in terms of situations, the set of situations of the first domain are mapped (extended) to the set of situations of the second domain. This is formally defined in the next section. Furthermore, it was proven in [Sco82] that the composition of approximate mappings is also an approximate mapping, thus allowing the modelling of sequential extensions.

Thus, concepts of Scott Domains and Situation Theory are comparable. A detailed and formal comparison can be found in [Bar91], but a brief summary appears in the following table:

Scott Domain Theory	Situation Theory
Proposition	Type
Entailment	Unconditional Constraints
Element	(Basic) Situation
Approximate Mapping	Conditional Constraints (Extension)

Table 5.1: Scott Domains Theory vs. Situation Theory

5.6 The qualitative components of the structured model

The structured representation of a document involves a set of basic situations, which were formally defined in section 5.3. As explained in section 5.4, the concept of a domain is re-defined within Situation Theory to refer to sets of basic situations.

5.6.1 Information domain

Let T_D be a set of types. Let S_D be the set of basic situations that are identified from the semantic-based analysis of a document's information content. The situations in S_D support the types in T_D. T_D and S_D form an *information domain* denoted $D = \langle T_D, S_D \rangle$. The information domain is a structured representation of the explicit, and the implicit and certain information content of the document. The latter comes from the application of the certain constraints (i.e., unconditional constraints, and certain and conditional constraints).

The notion of information domain is illustrated with the following example. Suppose that types are represented by terms and situations are represented by group of terms that are semantically related. Let

$$\{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}, t_{11}\}$$

be the set of types that are explicitly extracted from the text document. Let

$$\{t_1 \rightarrow x_1, t_3 \rightarrow x_3, t_9 \rightarrow x_9, t_{11} \rightarrow x_{11}\}$$
$$\{t_1 \rightarrow t_2, t_3 \rightarrow t_4, t_5 \rightarrow t_4, t_5 \rightarrow t_6, t_7 \rightarrow t_8, t_7 \rightarrow t_9, t_{11} \rightarrow t_8, t_{11} \rightarrow t_9, t_{11} \rightarrow t_{10}\}$$

be two sets of certain constraints. In the first set, the application of the constraints leads to information that is not explicit in the document, whereas in the second set, it leads to information that is explicit in the document[64]. Assume that the analysis of the text document leads to the following semantic contents: $s_1 \models t_1$, $s_2 \models t_3$, $s_3 \models t_5$, $s_4 \models t_7$ and $s_5 \models t_{11}$. The terms t_1, t_3, t_5, t_7 and t_{11} are not semantically related because none of the above constraints links any of these terms together. Therefore, the document can be modelled by the information domain $D = \langle T_D, S_D \rangle$, where

$$T_D = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}, t_{11}, x_1, x_3, x_9, x_{11}\}$$
$$S_D = \{s_1, s_2, s_3, s_4, s_5\}$$

This is shown in the figure below:

[64] For simplicity, the background conditions of the conditional constraints are not represented.

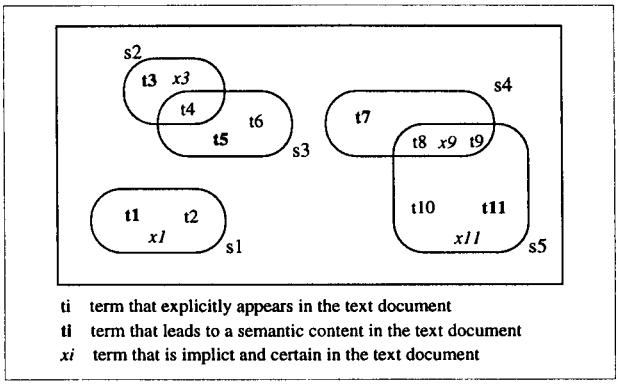


Figure 5.4: Representation of an information domain

In the domain D, $s_1 \models t_1$ because it is the semantic content associated with s_1, and $s_1 \models t_2$ because $t_1 \rightarrow t_2$ is a certain constraint. Also, $s_1 \models x_1$ because $t_1 \rightarrow x_1$ is a certain constraint. In the basic situation s_1, there is one type, t_1, in which all the others types are nested. The correct terminology is that the propositional contents related to the other types supported by s_1 are nested within $s_1 \models t_1$. The same terminology is used to refer to types when the same situation is involved (i.e. s_1).

Types can be shared by two or more basic situations, but such types do not constitute semantic content. Otherwise, only a basic situation that includes all the others would be involved, because all the other types will be nested within that type.

Information domains are based on Scott Domains, but are not identical to them. First, the certain constraints which concur with Scott's entailment relation are not defined for each information domain. The certain constraints are common to all information domains. Second, the basic situations which correspond to Scott's elements are primarily considered. Scott defines first an information system, upon which the set of elements that constitute the domain is constructed.

In the unstructured model, a document is modelled by a single situation. The information contained in the document can sometimes be contradictory. In that case, the situation modelling the document may support contradictory information; that is the situation may be inconsistent. If a document is modelled by an information domain, the situations that compose that information domain are consistent because they support semantically related information.

The construction of information domains from the analysis of documents is discussed in the next chapter. In the remainder of this chapter, it is assumed that documents are represented by information domains. In further references, the terms domain and information domain are both used to refer to the same concept.

5.6.2 Refinement of an information domain

As explained in Chapter 4, the transformation of a document into a fictitious one is an extension process. In the unstructured model, the extension process results from the application of uncertain conditional constraints[65]. This also applies to the structured model. However, the extension process

[65] These refer to the conditional constraints where the satisfaction of the background conditions is unknown.

starts from the set of basic situations that define the information domain modelling the document's initial information content. If the extensions of the set of the basic situations lead to a set of basic situations, these situations can define the information domain that models the fictitious document. This connection between information domains models the transformation process.

In [Sco82], Scott Domains are connected by an approximate mapping. In the structured model, the connection between two information domains is modelled by a *refinement function* which is analogous to the approximate mapping. The term refinement is used instead of approximate mapping because it is more appropriate to the ontology of extensions as defined in this thesis. Also, it is used in the Dempster-Shafer's framework to represent the connection between two bodies of evidence (this is explained when the quantitative components of the structured model are described in section 5.7).

Let $D_1 = \langle T_1, S_1 \rangle$ and $D_2 = \langle T_2, S_2 \rangle$ be two information domains. A refinement is a function $\omega : D_1 \rightarrow D_2$ defined on the two domains D_2 and D_1 as follows:

$$\text{for all situations } s \in S_1,\ \omega(s) = \begin{cases} E_d(s) & \text{if } E_d(s) \neq \emptyset \\ \{s\} & \text{if } E_d(s) = \emptyset \end{cases}$$

$E_d(s)$ is the set of direct extensions from s (i.e., for all $s' \in E_d(s)$, there is no $s'' \in E(s)$ such that $s \triangleright s'' \triangleright s'$, where $E(s)$ is the set of extensions of s). The construction of D_2 ensures that $\omega(s) \neq \emptyset$ because a situation that is not extended is maintained in D_2. The refinement function models the simultaneous extensions of the situations that constitute D_1, the *coarse* domain, into the situations that compose D_2, the *refined* domain. The refinement of the basic situations of D_1 constitutes the basic situations of D_2:

$$\bigcup_{s \in S_1} \omega(s) = S_2$$

The information supported by a basic situation of D_2 is also semantically related because it is the extension of a basic situation of D_1, and the extension is based on constraints which model semantic relationships. A basic situation s_2 of D_2 has a semantic content. If s_2 is the extension of one situation s_1 of D_1 with semantic content $s_1 \models \phi$, then the semantic content of s_2 is $s_2 \models \phi$, because the types supported by that basic situation are semantically related, directly or indirectly, to ϕ. The relationships are certain within s_2. Indeed, it is the refinement of s_1 into s_2 that is uncertain, not the information supported by s_2; this information is uncertain *with respect* to s_1. If two situations of D_1 are refined into s_2, this means that the semantic content of one of these situations becomes nested into the other one. Therefore, s_2 is also a basic situation since it has a unique semantic content.

The definition of the refinement function shows that a situation s in S_1 can be extended to a situation s' that supports types in T_1 that are not caused by the fact that s' is an extension of s. This means that an item of information can be both explicit (occurring in D_1) and implicit (found in D_2) in a document's information content.

The refinement function is extended to any subset of $A \subseteq S_1$ as follows:

$$\omega(\{s\}_{s \in A}) = \bigcup_{s \in A} \omega(s)$$

5.6.3 Conclusion

A document's initial information content is modelled as an initial information domain. The basic situations of the domain are determined by the semantic contents that are identified from a semantic analysis of the document's information content. The application of the uncertain constraints delivers the implicit and uncertain information. This information is represented in the refined domain, which is constructed from the initial domain. This process is defined as a refinement. It continues until no more unused uncertain constraint can be applied.

Thus far, there has been no mention of any quantitative component. The representation of the significance of information, the propagation and the aggregation of uncertainty, and the computation of the relevance degree have not been discussed. The representation of quantitative components in the structured model is achieved with the use of Dempster-Shafer's Theory of Evidence [Dem68, Sha76].

5.7 Dempster-Shafer's Theory of Evidence for Information Retrieval

Dempster-Shafer's Theory of Evidence defines the concepts *frame of discernment, focal element, basic probability assignment, belief function* and *refinement function.* The relevance of these concepts to the structured model is briefly discussed.

A *frame of discernment* is a set of propositions organized into subsets, each constituting a focal element. The *focal elements* can be compared to basic situations of an information domain. The propositions can be matched to the types in that domain.

A body of evidence is attached to a frame of discernment under the form of a basic probability assignment. This ascribes beliefs that are exactly committed to the focal elements. Similarly, the use of a *basic probability assignment* in an information domain can measure the significance of the information content of each basic situation with respect to the overall information content. The definition of the basic probability is altered to conform to the definition of the information domain.

A *belief function* computed on the basic probability assignment measures the amount of belief allocated to any set of propositions. Similarly, a belief function can be used to measure the extent to which the information need, as phrased in a query, is contained in an information domain. If a query is represented by a set of types, a belief function can express the degree to which these types are supported by the basic situations of the domain. The definition of the belief function is also modified to comply to the definition of an information domain.

A *refinement function* is defined between frames of discernment. Given two frames of discernment, a refinement consists of splitting the propositions of the coarse frame to obtain the refined frame. This function captures the propagation and aggregation of uncertainty from the coarse frame to the refined frame. These are reflected in the properties between the basic probability assignments of the two frames. This refinement function is similar to the refinement function defined in the previous section. A comparison of the two refinement functions is given in section 5.7.4.

The analogies between information domain and Dempster-Shafer's framework are summarized in the following table:

Dempster-Shafer's theory	Information Domain
Proposition	Type
Focal Element	Basic Situation
Basic Probability Assignment	Missing
Belief Function	Missing
Refinement	Refinement

Table 5.2: The Dempster-Shafer's Theory of Evidence vs. information domain

5.8 The quantitative components of the structured model

The definition of an information domain is augmented with concepts from Dempster-Shafer's Theory of Evidence to obtain a framework where information is structured and its quantitative features are represented. The additional notions are basic probability assignments and belief functions. They are redefined to accommodate the formalization of information domains.

5.8.1 Basic probability assignment

A *basic probability assignment* (BPA) defined on the information domain is a function $m_D : D \rightarrow [0,1]$ such that

$$\sum_{s \in S_D} m_D(s) = 1$$

The value of $m_D(s)$ represents the significance of the situation s with respect to the overall information content of the document. In a basic situation constructed with a set of synonymous terms, the frequency of the terms that constitute this basic situation can be used to compute its BPA. $m_D(s)$ is also referred to as the *weight* of the situation s in the domain D. The value of this weight grows with the significance attached to the information supported by s.

An information domain coupled with a BPA constitutes a structured and weighted representation of the information that is explicit and implicit with certainty in a document. In the remainder of this chapter, it is assumed that the weight (the BPA) of the basic situations of an information domain has been determined. The computation of m_D is discussed in the next chapter.

5.8.2 Belief function

Given an information domain $D = \langle T_D, S_D \rangle$, a belief function $Bel_D : D \rightarrow [0,1]$ is used to measure the amount of relevant information contained in that domain to a query. For a query represented by a set of types Φ, this measure depends on the existence of the basic situations of

that domain that are pertinent to Φ[66]:

$$Bel_D(\Phi) = \sum_{s \models \varphi \text{ and } \varphi \in \Phi}^{s \in S_D} m_D(s)$$

In Dempster-Shafer's framework, the belief of a set is based on the focal elements included in that set. Here, the belief of a set (of types) is based on the basic situations pertinent to that set. Some of the types supported by a pertinent basic situation may not belong to the set of types representing the query. However, the types supported by a basic situation are semantically related. A basic situation that is pertinent to a given set of types can be viewed as being included in that set of types. Suppose that a basic situation is a group of synonyms; for example, it supports the two terms "dog" and "barking animals"[67]. This basic situation is pertinent to a query, for example "information about dogs", if a term that is part of the query belongs to (is supported by) that situation. In that case, the other terms supported by that situation, for example "barking animals", are relevant to the query since they are synonymous to that term.

The formulation of $Bel_D(\Phi)$ uses the BPA of the situations that are pertinent to the query represented by Φ. Even if a situation supports several types of the query, its BPA is only included once in the summation. The reason for this is that a basic situation supports types which are semantically related. Suppose that a basic situation represents synonymous terms. In that case, if two terms used in the query are supported by the same situation, then the two terms are semantically related. The document should not be more relevant to a query that uses two synonymous terms in its expression than to a query that uses only one of these terms, since the information need is the same in both circumstances. Furthermore, if two terms are used to refer to the same piece of information in a document, then this piece of information is more significant in the document if only one term is used. This should already be captured in the weight associated to the basic situation that supports these two terms (for they are semantically related).

5.8.3 Weighted information domain

A weighting mechanism can be mapped onto an information domain by using a basic probability assignment and a belief function. The former measures the significance of the basic situations of the domain and the latter expresses the relevance of the information represented in that domain to a query. This mapping generates a *weighted information domain* which is denoted as $D = \langle T_D, S_D, m_D, Bel_D \rangle$.

5.8.4 Refinement of a weighted information domain

The refinement of an information domain models the transformation of the document symbolized by that information domain. Shafer also defines a refinement function, which is expressed from the propositions[68] of a (coarse) frame of discernment. The refinement of the propositions of the coarse frame constitutes the propositions of the refined frame. All the propositions of the coarse frame of discernment are refined. In the structured model, the refinement of a (coarse) information domain

[66] A situation s is pertinent to a set of type Φ if there exists at least one $\varphi \in \Phi$ such that $s \models \varphi$.
[67] For simplicity, types are represented directly by terms.
[68] In reality, the refinement function is defined on singletons (sets constituted of one proposition) to allow a generalization to sets of propositions.

applies to the set of basic situations that constitute that information domain. The refinement of the basic situations of the coarse domain leads to the basic situations of the refined domain. All the situations of the coarse domain are refined.

Although the refinement of a frame of discernment and the refinement of an information domain present an overall similar behavior, several characteristics of the refinement of a frame discernment are not followed by the refinement of an information domain. First, the refinement of a domain is defined in terms of situations, although it depends on the types supported by these situations. Shafer's refinement function is applied to propositions and then generalized into sets. Second, the structure of the refined domain depends on the structure of the coarse domain. In Shafer's framework, the set of focal elements of the refined frame is not constructed from the set of focal elements of the coarse frame[69]. Third, the refined domain contains the types of the coarse domain, plus those obtained by refinement. These types are grouped into situations. With Shafer's refinement, only the propositions which result from the refinement are kept in the refined frame. Fourth, it is possible that two or more situations are refined in the same situation because the refinement of a situation is caused by the application of one or several conditional constraints which may converge to the same piece of information. In Shafer's refinement function, two propositions cannot be refined into the same proposition[70].

Let $D = \langle T_D, S_D, m_D, Bel_D \rangle$ and $D' = \langle T_{D'}, S_{D'}, m_{D'}, Bel_{D'} \rangle$ be two weighted information domains, the former being refined into the latter. The qualitative characteristics of the refinement of an information domain, that is, the construction of T'_D and S'_D from T_D and S_D, were defined in section 5.5.2. The quantitative characteristics of the refinement of an information domain, which have not been discussed, model the propagation and the aggregation of the uncertainty. The quantitative characteristics of Shafer's refinement function are given by the properties between the basic probability assignments of the coarse frame and the refined frame. The same holds true for the quantitative characteristics of the refinement function defined for information domains. The BPA of the two weighted information domains must be defined to model the propagation and aggregation of the uncertainty.

5.8.5 Computation of the basic probability assignment of the refined domain

Let $D_i = \langle T_i, S_i, m_i, Bel_i \rangle$ and $D_{i+1} = \langle T_{i+1}, S_{i+1}, m_{i+1}, Bel_{i+1} \rangle$ be two weighted information domains related by the refinement function $\omega_i : D_i \rightarrow D_{i+1}$. Both properties below are required for m_i and m_{i+1} to be BPAs:

$$\sum_{s \in S_i} m_i(s) = 1 \quad \text{and} \quad \sum_{s \in S_{i+1}} m_{i+1}(s) = 1$$

Suppose that the BPA m_i is already known, then the BPA m_{i+1} must be determined. Let $s \in S_i$ and let $s \triangleright s'$ be a branch such that $s' \in \omega_i(s)$ (s is refined into s'). The value of $m_{i+1}(s')$ is given by

$$m_{i+1}(s') = \partial(s \triangleright s') * m_i(s)$$

[69] However, there are properties imposed on the BPAs of the coarse and the refined frames which, when used in a certain way, can lead to the construction of the focal elements of the refined frame based on the focal elements of the coarse frame (this was discussed in Chapter 3).

[70] It would be interesting to see what the characteristics of the refinement of an information domain engenders in Shafer's framework. This theoretical work is perhaps the subject of further research, for its result is not relevant to this particular thesis.

This is analogous to the formulation used in the unstructured model. At s the uncertainty propagated so far is $m_i(s)$. If $s_0 \triangleright s_1 \triangleright \ldots \triangleright s$ is the only branch that leads to s, then

$$m_i(s) = \partial(s_0 \triangleright \ldots s) * m_0(s_0)$$

which is the uncertainty of the branch $\partial(s_0 \triangleright \ldots \triangleright s)$ multiplied by a factor. This is defined as the weight $m_0(s_0)$. In the unstructured model, this factor equated to a value of 1 because the information was not weighted.

Several basic situations of the information domain D_i can be refined into one situation s'. The formula above is generalized as follows:

$$\sum_{s \in \omega_i^{-1}(s')} \partial(s \triangleright s') * m_i(s) = m_{i+1}(s')$$

$\omega_i^{-1}(s')$ is the set of the situations in S_i of which s' is an extension. This formulation captures the fact that the more situations are refined into one situation, the more significant is that situation. It is easy to verify that m_{i+1} is a BPA. This is due to the normalization of the constraints.

In the refinement process some situations may be refined into themselves (they cannot be further extended). Let s' be such a situation. In such case, $m_{i+1}(s') = m_i(s')$ thus indicating that the weight attached to the situation s remains the same. If another situation in D_i is extended to s', then $m_{i+1}(s')$ must take into account that situation (as defined in the generalized formula).

The formulation used to evaluate $m_{i+1}(s')$ is compatible with many models of the propagation and aggregation of uncertainty; $m_i(s)$ corresponds to an uncertain fact and $\partial(s \triangleright s')$ can be viewed as an uncertain rule. The choice of the product and summation is debatable; other combinations may be more accurate. This issue was discussed in the previous chapter in section 4.6.

In Shafer's framework, some properties relate the coarse and refined frames. One property states that the BPA of a set A is always greater or equal to the BPA associated to its refined set $\omega_i(A)$. That is,

$$m_i(A) \geq m_{i+1}(\omega_i(A))$$

This implies two facts. First, if A is not a focal element (i.e., $m_i(A) = 0$), the refinement of A cannot be a focal element. Second, if A is a focal element (i.e., $m_i(A) > 0$) and $\omega_i(A)$ is a focal element (i.e., $m_{i+1}(\omega_i(A)) > 0$), the BPA of this focal element cannot be greater than the BPA of A. In Shafer's framework, refinement increases uncertainty. Applied to an information domain, the analogous inequality would be

$$m_i(s) \geq m_{i+1}(\omega_i(s))$$

where s is a situation. The interpretation of $m_{i+1}(\omega_i(s))$ is different because $\omega_i(s)$ is a set of situations. This inequality could mean that the BPA of the sum of all $m_{i+1}(s')$ such that $s' \in \omega_i(s)$ should not be greater than $m_i(s)$, thus

$$\sum_{s' \in \omega_i(s)} m_{i+1}(s') \leq m_i(s)$$

Situations can be refined into the same situation, so this property is not satisfied. On the contrary, it can be proven that (the proof is left as an exercise)

$$\sum_{s' \in \omega_i(s)} m_{i+1}(s') \geq m_i(s)$$

The corresponding inequality is then

$$m_i(s) \leq m_{i+1}(\omega(s))$$

Equality is obtained whenever the situation s shares no extensions with other situations of the domain D_i. The difference between the inequalities in Dempster-Shafer's refinement function and the one proposed in this chapter occurs because an information item may be implicit in several information items[71].

In summary, the document is initially modelled by a weighted information domain $D_0 = \langle T_0, S_0, m_0, Bel_0 \rangle$. The extensions of the situations in S_0 lead to the construction of successive refined domains $D_i = \langle T_i, S_i, m_i, Bel_i \rangle$, for $i > 0$.

There is a difference in the interpretation of m_0 and m_i. For $s \in S_i$, the BPA $m_i(s)$ represents the uncertainty that s is obtained from the initial domain D_0 after i successive refinements. For $s \in S_0$, the quantity $m_0(s)$ measures the significance of the situation s in the initial domain. The quantities $m_i(.)$ are computed from m_0 and the uncertainty of the constraints used in the refinement process. m_0 is established when processing the document.

5.8.6 Formulation of the relevance degree

A document's initial information content is modelled by a weighted information domain $D_0 = \langle T_0, S_0, m_0, Bel_0 \rangle$, which captures the information that is explicit, and implicit and certain in the document. The implicit and uncertain information is represented in the different weighted information domains that result from the successive refinements of D_0. The refinement process continues until a weighted information domain is obtained in which all the basic situations cannot be further extended. The last information domain corresponds to a structured and weighted representation of the explicit, implicit and certain, and implicit and uncertain information of the document. The belief function associated with that domain can express the relevance of the document to the query, because it evaluates the extent to which the information supported by the basic situations of that domain, which is all that can be obtained from the document, concerns the query. If $D_n = \langle T_n, S_n, m_n, Bel_n \rangle$ is the last domain and Φ is the set of types representing the query then $Bel_n(\Phi)$ represents the relevance degree of the document to the query represented by Φ.

This computation is inefficient because obtaining the final information domain is often unnecessary. Indeed, many situations can be kept as such in the refinement process without affecting the final value of the relevance degree. A more efficient formula that leads to the same result is proposed.

During the extension process of the unstructured model, a situation pertinent to the query was not further extended because it did not affect the value of the relevance degree of the document. This also applies to the structured model. Indeed, let s a basic situation of a domain D_i be refined into

[71] See [Eva82] for a philosophical discussion on that matter.

the situations $s_1, \ldots, s_k$ of the domain D_{i+1}. If no other situation of the domain D_i is refined to any of the situations of $s_1, \ldots, s_k$, the following equality holds[72]:

$$m_i(s) = \sum_{j=1,k} m_{i+1}(s_j)$$

Assume that s is a pertinent situation with respect to the query, and that D_{i+1} is the final domain (i.e., no situation in D_{i+1} can be further extended). The fact that s is pertinent implies that all the situations $s_1, \ldots, s_k$ are also pertinent. Therefore, the value

$$\sum_{j=1,k} m_{i+1}(s_j)$$

is included in the summation that formulates the belief function associated with the domain D_{i+1}. This value is exactly $m_i(s)$. This shows there is no point in extending s, for the extension of a pertinent situation does not change the value of the relevance degree with respect to that situation. Moreover, if there is a situation s' in D_i that is refined to one of the situations $s_1, \ldots, s_k$, this latter situation is represented in D_{i+1} and the associated weight comes from the situation s'. The weight that comes from the situation s is already included in $m_i(s)$.

This shows that the pertinent basic situations of an information domain do not need to be extended. However, in order to preserve the initial structured representation of the document, the refinement of a pertinent situation of a domain is set to itself. Therefore, a situation s from a domain D_i is refined into the following situation(s) in the domain D_{i+1}:

(i) s if it cannot be extended (i.e., $E_d(s) = \emptyset$) or if all constraints that could have been applied have already been used,

(ii) s if it is a pertinent situation with respect to the query, or

(iii) $E_d(s)$ if (i) and (ii) do not apply.

In the definition of the refinement function, case (ii) was not mentioned because the refinement was performed without regard to the query.

The weight of a pertinent situation in the refined domain remains the same, unless it is also the extension of another situation of D_i. In that case, the weight that comes from the extension is added to the weight of the pertinent situation.

The new definition of refinement leads to a more efficient computation of the relevance degree. Let the initial representation of the document be the domain $D_0 = \langle T_0, S_0, m_0, Bel_0 \rangle$. This domain is refined into successive domains. During the refinement process, the pertinent and non-extendible situations in a domain are kept as such in its refined domain. The refinement process ceases when a domain, for instance $D_n = \langle T_n, S_n, m_n, Bel_n \rangle$, in which the situations are either pertinent or non-extendible is reached.

Let Φ be the representation of the query. For all $D_i = \langle T_i, S_i, m_i, Bel_i \rangle$, $i \geq 0$, $Bel_i(\Phi)$ is the degree of relevance of the document after i refinements. This value is the summation of the BPA of the pertinent basic situations of D_i. If the summation was restricted to those pertinent basic

[72] This equality arises from the fact that, first, constraints that lead to extensions are normalized, and second, the summation of the BPA of the basic situation of an information domain is, by definition, equal to 1.

situations that become pertinent in D_i, the formulation of the relevance can take into account that the refinement of a pertinent situation does not affect the value of the relevance degree. To restrict the summation, it is therefore necessary to distinguish between pertinent, extendible and non-extendible situations. For this purpose, the two following sets are defined:

$$P(A,\Phi) = \{s \in A | \text{there exists } \varphi \in \Phi, s \models \varphi\}$$
$$NE(A) = \{s \in A | E_d(s) = \emptyset\}$$

$P(A,\Phi)$ is the set of pertinent situations with respect to Φ in A. $NE(A)$ is the set of non-extendible situations in A. No type supported by these situations appears as the antecedent of an uncertain conditional constraint. If it does, the corresponding constraint has already been used to arrive at these situations. The definition of a belief function is also modified so that only the basic pertinent situations that become so in a domain are considered. Therefore, a belief function $Bel_i : 2^{S_i} \times D_i \rightarrow [0,1]$ is defined for each domain D_i with respect to a given set of situations in S_i:

$$for\ A \subseteq S_i \qquad Bel_i(A,\Phi) = \sum_{s \models \varphi \text{ and } \varphi \in \Phi}^{s \in A} m_i(s)$$

$Bel_i(A,\Phi)$ represents the belief that Φ is supported only by the situations in A in the domain D_i. Based on the new definition of belief functions, and the definitions of two sets, it is possible to formulate the relevance degree.

Let $\Re : S \times 2^T \rightarrow [0,1]$ be the function that measures the relevance degree. Although the representation of a document involves a number of situations, $\Re$ is defined in terms of the situation which models the document's initial information content. Let d be this situation (each basic situation of the domain D_0 is included in the sense of $\subseteq$ in d). The formulation of the relevance degree of the document to the query represented by the set of types Φ is defined as

$$\Re(d,\Phi) = \sum_{i=0}^{n} Bel_i(\Gamma_i,\Phi)$$

where $$\Gamma_i = \begin{cases} S_0 & \text{if } i = 0 \\ \omega_{i-1}(\Gamma_{i-1} - P(\Gamma_{i-1},\Phi) - NE(\Gamma_{i-1})) & otherwise \end{cases}$$

Γ_i is the set of situations that are extensions of situations in Γ_{i-1} (with respect to ω_i) in which all pertinent situations with respect to Φ (i.e., $P(\Gamma_{i-1},\Phi)$) and all non-extendible situations (i.e., $NE(\Gamma_{i-1})$) have been removed. This formulation expresses the relevance degree of a document to a query with respect to its explicit and implicit information content. It can be easily proven that $\Re(d,\Phi)$ is the same as $Bel_n(\Phi)$. The difference is that the determination of the former is more efficient.

5.8.7 Example

Consider the domain $D_0 = \langle T_0, S_0, m_0, Bel_0 \rangle$ which can be extended into $D_1 = \langle T_1, S_1, m_1, Bel_1 \rangle$

which can then be extended to $D_2 = \langle T_2, S_2, m_2, Bel_2 \rangle$ as follows:

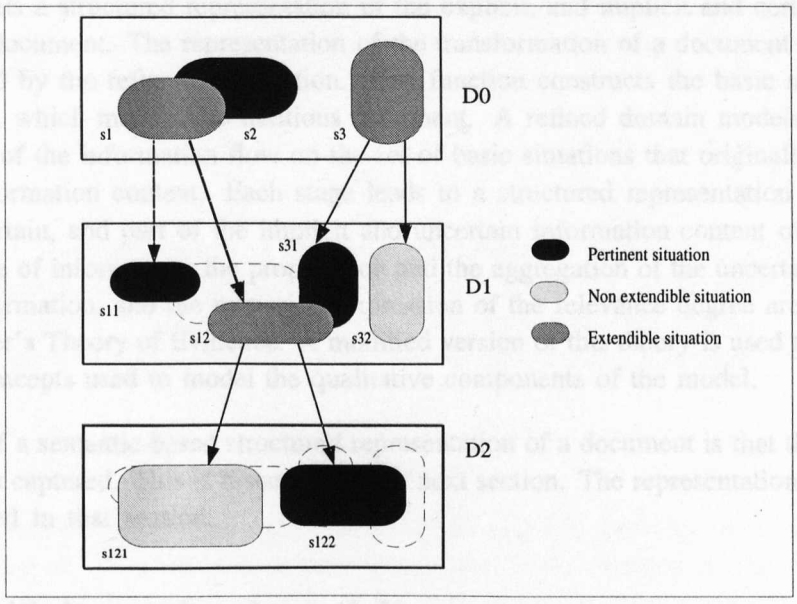


Figure 5.5: Example of the refinement process

In this figure, for example $\omega_0(s_1) = \{s_{11}, s_{12}\}$ and $\omega_1(s_{12}) = \{s_{121}, s_{122}\}$. The different values of the set of basic situations S_i, the set of extended situations Γ_i, the set of pertinent situations $P(\Gamma_i, \Phi)$ and the set of non-extendible situations $NE(\Gamma_i)$ are given in the table below:

	S_i	Γ_i	$P(\Gamma_i, \Phi)$	$NE(\Gamma_i)$
$i = 0$	$\{s_1, s_2, s_3\}$	$\{s_1, s_2, s_3\}$	$\{s_2\}$	$\emptyset$
$i = 1$	$\{s_{11}, s_{12}, s_2, s_{31}, s_{32}\}$	$\{s_{11}, s_{12}, s_{31}, s_{32}\}$	$\{s_{11}, s_{31}\}$	$\{s_{32}\}$
$i = 2$	$\{s_{11}, s_{121}, s_{122}, s_2, s_{31}, s_{32}\}$	$\{s_{121}, s_{122}\}$	$\{s_{122}\}$	$\{s_{121}\}$

Table 5.3: The different steps of the refinement process

From the domain D_2, the set Γ_3 is computed as follows:

$$\begin{aligned}\Gamma_3 &= \omega_2(\Gamma_2 - P(\Gamma_2, \Phi) - NE(\Gamma_2)) \\ &= \omega_2(\{s_{121}, s_{122}\} - \{s_{122}\} - \{s_{121}\}) \\ &= \omega_2(\emptyset) \\ &= \emptyset\end{aligned}$$

Therefore, no more refinement is possible, so

$$\begin{aligned}\Re(d, \Phi) &= Bel_0(\Gamma_0, \Phi) + Bel_1(\Gamma_1, \Phi) + Bel_2(\Gamma_2, \Phi) \\ &= m_0(s_2) + m_1(s_{11}) + m_1(s_{31}) + m_2(s_{122})\end{aligned}$$

5.8.8 Conclusion

The model presented in this chapter caters for a structured representation of documents. Situation Theory is used to model the qualitative components. Information is symbolized by types. The flow of information is represented by constraints and the situations affected by the application of these

constraints. Information is organized into basic situations which constitute an information domain. The latter models a structured representation of the explicit, and implicit and certain information content of the document. The representation of the transformation of a document into a fictitious one is modelled by the refinement function. This function constructs the basic situations of the refined domain, which models the fictitious document. A refined domain models some stage in the application of the information flow on the set of basic situations that originally constitute the document's information content. Each stage leads to a structured representation of the explicit, implicit and certain, and part of the implicit and uncertain information content of the document. The significance of information, the propagation and the aggregation of the uncertainty inherent to the flow of information, and the numerical expression of the relevance degree are modelled with Dempster-Shafer's Theory of Evidence. A modified version of this theory is used to conform with the different concepts used to model the qualitative components of the model.

One outcome of a semantic-based structured representation of a document is that the specificity of the document is captured. This is discussed in the next section. The representation of exhaustivity is also discussed in that section.

5.9 Specificity and exhaustivity

The *specificity* of a document to a query is the extent to which the information in the document relates to the query. The *exhaustivity* of a document to a query is the extent to which all the information sought by the query is contained in the document. For example, a document represented by the set $\{dog, cat\}$[73] is specific to a query requiring information about "dog and cat", but no to a query looking for information about "dog" because the document contains information that is not related to "dog". The document is exhaustive with respect to a query seeking information about "dog", but not to a query requiring information about "dog and horse" because the document does not contain information about "horse".

It was shown in Chapter 4 that both the specificity and the exhaustivity of a document to a query were not captured in the unstructured model. The specificity of a document could not be captured because a situation pertinent to a query could support information that is not related to the query. It was not possible to express how much information supported by a pertinent situation was relevant to the query. The exhaustivity of a document could not be represented because a situation is pertinent to a query if it supports at least one of the information items contained in the query. It was not possible to verify whether all the information sought by the query was contained, either explicitly or implicitly, in the document. In this section, an expression of each measure is proposed, followed by a method of combining both measures.

5.9.1 Specificity

In the structured model, a document's initial information content is modelled by a weighted information domain $D = \langle T_D, S_D, m_D, Bel_D \rangle$. The information is structured into basic situations (the situations in S_D). The basic situations support semantically related information (the types in

[73] For simplicity, a document is represented by a set of terms.

T_D). Let Φ be the set of types modelling the query. Two concepts defined in the unstructured model used in tandem with the structured model are used to express a measure of the specificity of the document. These concepts are $B(d, \Phi)$, the set of minimal branches with root d, and whose leaves are situations that are pertinent to Φ, and $\partial(.)$ the uncertainty attached to branches. The value

$$\sum_{b \in B(d,\Phi)} \partial(b)$$

is the summation of the uncertainty of all $\Phi-$minimal branches with root d. This value measures the extent to which a situation d contains, explicitly and implicitly, information that concerns the query. If the situation d is a basic situation of the information domain D (i.e., $d \in S_D$), a weight $m_D(d)$ is assigned to that basic situation reflecting its significance. Therefore, the value

$$m_D(d) * \sum_{b \in B(d,\Phi)} \partial(b)$$

reflects the uncertainty associated with obtaining $\Phi-$minimal branches that originate from a weighted situation d. This value is calculated for each basic situation of the domain D. The values obtained for all the basic situations of the domain can be combined to express a measure of specificity as follows:

$$Sp(D, \Phi) = \sum_{d \in S_D} \left(m_D(d) * \sum_{b \in B(d,\Phi)} \partial(b) \right)$$

$Sp(D, \Phi)$ reflects the specificity of the document modelled by the information domain D to the query symbolized by the set of types Φ for the following reasons:

(i) if all the basic situations of the domain D are pertinent, then $Sp(D, \Phi) = 1$. Since the information supported by these situations is semantically related, the entire information content of the document concerns the query, thus indicating that the document is specific to the query.

(ii) if all the extensions of the basic situations of the domain D lead to pertinent situations, then $Sp(D, \Phi) = 1$. This means that the flow of information originating from the document always leads to situations pertinent to the query. The document is specific to the query because information remains structured during the extension process.

(iii) if $Sp(D, \Phi) < 1$, some of the basic situations of the domain D or one of the refined domains do not concern the query. The document contains information that is not relevant to the query. The higher the value of $Sp(D, \Phi)$, the more specific the document is to the query.

(iv) if $Sp(D, \Phi) = 0$, the document is irrelevant to the query.

The value of $Sp(D, \Phi)$ coincides with the degree of relevance that is computed in the structured model (this can easily be proven). Therefore, the relevance degree of a document to a query, as computed by the structured model, constitutes a measure of specificity of the document to the query.

An example illustrating the difference between the values of the relevance degree of a document in the unstructured model and the structured model is given. Let a query be modelled by the set of types $\{\alpha_{22}, \gamma_{22}\}$. Suppose that the two types α and γ are explicitly identified in a document

information content. It is assumed that the constraints in (for simplicity, the background conditions are not represented)

$$\left\{\begin{matrix}\alpha \rightarrow \alpha_1, & \alpha \rightarrow \alpha_2, & \alpha_2 \rightarrow \alpha_{21}, & \alpha_2 \rightarrow \alpha_{22} \\ \gamma \rightarrow \gamma_1, & \gamma \rightarrow \gamma_2, & \gamma_2 \rightarrow \gamma_{21}, & \gamma_2 \rightarrow \gamma_{22}\end{matrix}\right\}$$

lead to extensions, and that they can be consistently applied together. The values of the relevance degree are calculated for both an unstructured and a structured representations of the document. The figure below illustrates both representations, together with the extension process involved in each representation. Only the extensions that lead to situations pertinent to the query are represented. The values of the uncertainty that propagates along these extensions (for the example, the values are arbitrarily defined, but they still satisfy the normalization criteria discussed in section 4.4.1.9) are indicated in the figure.

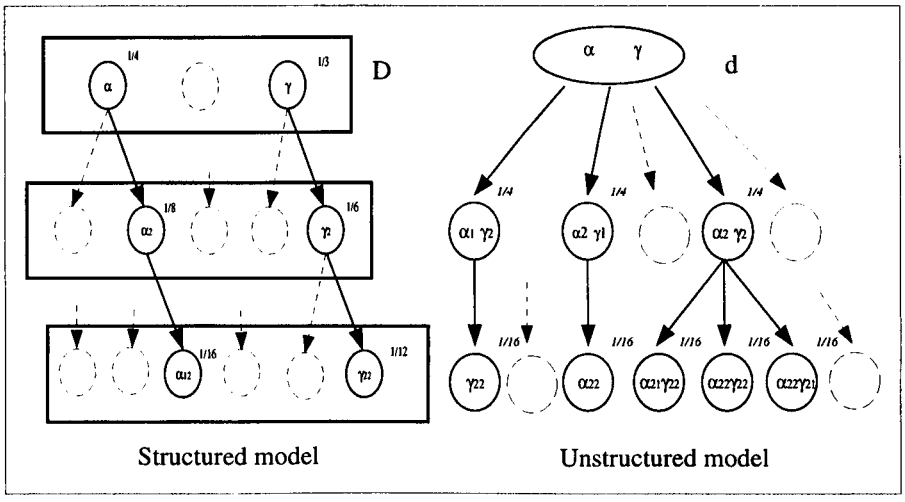


Figure 5.6: Specificity in the unstructured model and the structured model

In the structured model, the value of the relevance degree is $1/16 + 1/12 = 0.145$ whereas in the unstructured model the value is $1/16 + 1/16 + 1/16 + 1/16 + 1/16 = 0.312$. The value is lower in the first case because information is structured according to its semantics. Consequently, there are less extensions that lead to situations that support α_{22} or γ_{22}. The information supported by these situations is also semantically related, if not equal to α_{22} or γ_{22}. These situations are specific to α_{22} or γ_{22}. This is not the case in the unstructured model because a situation that supports α_{22} or γ_{22} may support information that is unrelated to α_{22} or γ_{22}.

5.9.2 Exhaustivity

The relevance degree expressed by the structured model corresponds to a measure of specificity. The exhaustivity is to some extent captured in the structured model by the fact that the situations are extended until some relevant information is found. However, it is not possible to tell with this approach whether all the information being sought is found.

Exhaustivity was not captured in the unstructured model because the computation of the relevance degree is based on minimal branches. In order to capture the exhaustivity of a document, it is necessary to identify all the information that is contained explicitly or implicitly in the document.

This is possible if the extension process is carried out as far as possible (i.e., it does not cease when a pertinent situation to the query is obtained).

Let a document be modelled by a single situation d. Let $B_t(d)$ be the set of branches that originate from the situation d and whose leaves are situations that cannot be further extended. This set is referred to as the set of *maximal branches*. The leaf of each of these maximal branches is a situation that supports all the information contained or derived from a document, with respect to a particular application of uncertain constraints. Therefore, the set $B_t(d)$ represents all the alternative maximal extensions of the situation d.

Let Φ be the set of types representing the query. Let $B_t(d, \Phi)$ be the set of all maximal branches that originate from the situation d and whose leaves support *all* the types in Φ. This set can be used to reflect the exhaustivity of the document to a query. The size of $B_t(d, \Phi)$ is a first indication of the exhaustivity of the document. Indeed, if the document is exhaustive to the query then there should be at least one maximal branch whose leaf supports all the information being sought by the query. If there are many such branches, the document can be considered as highly exhaustive with respect to the query. If there are no such branches, then the document is not exhaustive to the query because there is no extension of the original situation that leads to a situation that supports all of Φ. Even if the information items that constitute the query may be contained, one by one, in different leaves of the maximal branches, the document is not exhaustive because maximal branches model alternative extensions of the initial situation.

The uncertainty value associated to each of the branches in $B_t(d, \Phi)$ is a second indication of the exhaustivity of the document. Indeed, the more uncertain is a maximal branch, the more uncertain is the information supported by the leaf situation of that branch; that is, the document is less exhaustive with respect to that branch. One method that combines both the size of the set of maximal branches and the uncertainty associated to these branches is as follows:

$$Ex(d, \Phi) = \sum_{b \in B_t(d,\Phi)} \partial(b)$$

$Ex(d, \Phi)$ measures the exhaustivity of the document represented by the situation d with respect to the query symbolized by a set of types Φ. If $Ex(d, \Phi) = 0$, the document is not exhaustive because no branch leads to a situation that supports all of Φ. If $Ex(d, \Phi) > 0$ then the document is exhaustive because at least one of the maximal branches satisfies all the information need. The uncertainty of these branches reflects the degree of exhaustivity. The more maximal branches that lead to situations that support all of Φ, the more exhaustive is the document to the query. If $Ex(d, \Phi) = 1$, either the situation that is used to model the document (i.e., d) supports all the types of the query, or all the maximal extensions of that situation lead to situations that supports all the types of the query. In that case, the document is exhaustive to the query.

5.9.3 Combination of specificity and exhaustivity

The initial information content of a document is represented by a situation in the unstructured model and by a weighted information domain in the structured model. Let d be that situation and D be that domain. Let Φ be the set of types that represent the query. $Ex(d, \Phi)$ and $Sp(D, \Phi)$ measure the exhaustivity and the specificity of the document to the query, respectively. The following formula can be used to account for both exhaustivity and specificity in the expression of relevance

degree of the document to the query:

$$\frac{a * Ex(d, \Phi) + b * Sp(D, \Phi)}{a + b}$$

a and b represent the importance attached to exhaustivity and specificity, respectively. a and b are real numbers in the interval $[0, 1]$. The higher a, the more importance is attached to exhaustivity. The same applies with respect to b and specificity.

The interpretation of $Ex(d, \Phi)$ and $Sp(D, \Phi)$ are speculative so far and experiments are necessary to ascertain empirically how well, if at all, they reflect the exhaustivity and specificity of the document. The validity of these measures is investigated in Chapter 7.

5.10 Possible extensions of the structured model

The structured model accounts for a semantic-based structured representation of document. The model can be expanded to incorporate additional, better or different features. An example of each type of features are discussed in this section.

An additional feature is the representation of a transformation in general (i.e., addition, modification or deletion of information). As for the unstructured model, the use of channels [Bar92] instead of extensions can lead to a model that captures transformation in general. The refinement of a domain will then be defined as a set of channels which link the basic situations of an information domain to the basic situations of its refined domain. The flow of information determines the nature of these channels.

A better handling of the uncertainty should be incorporated in the structured model. As for the unstructured model, the formulations of propagation and the aggregation of uncertainty in the structured model assume the independence of information an the background conditions (this issue was discussed in the previous chapter in section 4.6). However, formulations of the propagation and the aggregation of uncertainty that do not assume this independence can be incorporated in the structured model. Indeed, the propagation and the aggregation of uncertainty are expressed by the relationships between the BPA associated to a weighted information domain and the BPA associated to its refined weighted information domain. To capture the dependence of information, it is then only necessary to reformulate the relationships between the two BPAs. The rest of the model should remain the same.

Different features can be incorporated in the structured model. For example, the structured model can take into account types of structures other than those that are semantic-based. Structures can be pragmatic-based. Examples of pragmatic-based structures are found in a document collection built by [Lid91]. The collection consists of abstracts organized into pragmatic-based structures. These structures are referred to as discourses. Examples of discourses are "purpose" "methodology", "result", etc. The structured model will represent each discourse by a basic situation. The definition of a basic situation will have to be modified, for it will not be semantically based. The weight attached to the basic situations will take into account the importance attached to each type of discourses, as well as the importance of the information in the discourses.

5.11 Conclusion

This chapter proposed a model of an IR system for a structured and weighted representation of the document. The model follows the Transformation Principle, and models transformation as an extension process. The qualitative components of the model are represented by Situation Theory. In the model, information is structured in a set of basic situations, where a basic situation is a situation that supports semantically related information. The quantitative components are represented by a modified version of Dempster-Shafer's Theory of Evidence. Some simplifications were made about the propagation and the aggregation of the uncertainty. Finally, it was possible to express a measure of specificity and exhaustivity. A combination of these measures may be used to express the relevance degree.

In summary, Chapter 4 and 5 presented a model of an IR system for each of the following cases:

(i) an unstructured representation of a document's information content
(ii) a structured representation of a document's information content

The two models together with their properties have been formally introduced. The next stage of the work is to study the empirical behavior of these models. An implementation of each model is carried out in order to determine its performance. These implementations are described in the next chapter.

Chapter 6
The Implementation of the Models

6.1 Introduction

In this thesis, two new models of an IR system are proposed, which are based on the Transformation Principle. The first model, outlined in Chapter 4, caters for an unstructured representation of a document's information content. It is referred to as the *unstructured model.* The second model, outlined in Chapter 5, caters for a structured representation of a document's information content. It is referred to as the *structured model.* The implementation of these models is described in this chapter.

The components of an IR model based on the Transformation Principle are classified as qualitative or quantitative. The representations of the qualitative components of the unstructured and the structured models are based on Situation Theory [Bar89, Dev91]. The qualitative components and their representations in both models are shown in the following table:

Qualitative components	Unstructured model	Structured model
Information item	Type	
Knowledge set (Semantic relationship)	Unconditional constraints Conditional constraints with their uncertainty degree	
Query	Set of types	
Document	Situation	Weighted information domain
Transformation (Flow of information)	Branch (Extension)	Refinement
Structure (Semantic)	N/A	Basic situation

Table 6.1: The qualitative components

The representations of the quantitative components of the two models are based on two different theories. The quantitative components of the unstructured model are represented by a general uncertainty mechanism, and the quantitative components of the structured model are represented by Dempster-Shafer's Theory of Evidence [Dem68, Sha76]. The quantitative components and their

representations in both models are shown in the following table:

Quantitative components	Unstructured model	Structured model
Significance of information (Weight)	N/A	Basic probability assignment
Propagation of uncertainty	*	Refinement
Aggregation of uncertainty	+	
Relevance degree	+	Belief functions

Table 6.2: The quantitative components

The discussion of the implementation of both the unstructured and the structured models highlights those components common to both models, and those components differentiating each model. The common components are types (section 6.2) and constraints (section 6.3). The implementation of the unstructured model (section 6.4) requires the implementation of situations and extensions. The implementation of the structured model (section 6.5) requires the implementation of weighted information domains, refinements, basic situations, basic probability assignments, and belief functions. The representation of a query is common to both models, but the implementation of queries is discussed in section 6.4 because it requires an understanding of some of the concepts introduced in that particular section.

6.2 Implementation of types

Types model information items. The translation of the information items into types is a complex process. For example, the sentence "the dog runs" should be transformed into the following type:

$$\left[\dot{d}|\dot{d} \models \ll Run, dog; 1 \gg\right]$$

In [FLV87], an intermediary representation of the sentence, referred to as schemata[74], is first used to represent the sentence. The schemata is then transformed into the type. The schemata associated with the sentence "the dog runs" is

$$\begin{bmatrix} REL\ "run" & & \\ ARG1 & \begin{bmatrix} IND\ "ind1" & \\ SPEC\ "the" & \\ COND & \begin{bmatrix} REL\ "dog" \\ ARG1\ "ind1" \\ POL\ 1 \end{bmatrix} \end{bmatrix} \\ POL\ 1 & \end{bmatrix}$$

[74] In [FLV87], schematas are used as a trade-off between Situation Semantics [BP83] and Discourse Representation Theory [Kam91] (the latter being used to model intentional states). The use of Discourse Representation Theory as an alternative to Situation Semantics is discussed in [Coo91a, Coo].

This example shows that the translation of more complex sentences into schematas, and the translation of schematas into types can be very difficult. The translation of information items into types can also be difficult due to the problem of capturing the full meaning of natural language. As a result, often only a restricted area of natural language can be covered by the translation process (see [Bla92]). Even then a correct translation cannot be ensured[75]. Another issue is that such a rich representation of an information item may not be useful in the context of information retrieval, unless it is carefully determined.

Aside from the complexity involved in converting a document's information content into types, the determination of the semantic relationships between information items is itself problematic. The transformation of a document depends on these relationships. Therefore, the appropriate capturing of these relationships is crucial to the implementation of the models proposed in this thesis.

There are available systems from which appropriate semantic relationships can be extracted, namely *thesauri*. However, the semantic relationships stored in most of these thesauri are only related to terms (i.e., single words or groupings of words). Although the use of thesauri limits the full capturing of natural language, the use of existing thesauri is preferable to the onerous task of determining each semantic relationship. Therefore, in this implementation, information items correspond to terms. If w is a term, then its corresponding type is denoted $'w'$, where

$$'w' = \left[\dot{d}|\dot{d} \models \ll present, w; 1 \gg\right]$$

Similar simplifications were followed in [HB94]. In future references, unless otherwise stated, a type is represented by the term to which it corresponds. For example, the type $'mathematics'$ represents the term "mathematics", where $'mathematics'$ represents $[\dot{d}|\dot{d} \models \ll present, mathematics; 1 \gg]$[76].

An advantage which results from the representation of information items as terms is that the performance of an IR system will not be compromised because of the complexity of the implementation of the information items or the inappropriateness of the semantic relationships. Indeed, with a more complex representation of information items, poor results would not necessarily be due to the models, but rather to the complexity of the models' implementation. Other implementations of the models are discussed in Chapter 8.

In the remainder of this chapter, when types are implemented, they represent information items that correspond to terms. The implementation of the constraints is discussed next. The outcome of this discussion is applicable to both models because constraints are common to the unstructured and the structured models.

6.3 Implementation of the constraints

The knowledge set is modelled by a set of unconditional constraints K_1 and a set of conditional constraints K_2, respectively:

$$K_1 = \{\varphi \rightarrow \varphi'\}$$

[75] These problems are not new and enter the area of natural language processing [Win83, DKZ85, Fro86].

[76] Types based on negative infons are not considered. Indeed, the interpretation of $[\dot{d}|\dot{d} \models \ll present, mathematics; 0 \gg]$ in IR has been and is still a problem because it is not known wether it means that the term "mathematics" is explicitly not in the document, or is that there is another term that contradicts it.

$$K_2 = \{\varphi \rightarrow \varphi' | B\}$$

In $\varphi \rightarrow \varphi'$, φ and φ' are known as the antecedent and the consequent of the constraint, respectively. An uncertainty function $cert : S \times BC \rightarrow [0,1]$ measures the uncertainty attached to the use of conditional constraints. The purpose of this section is to describe the implementation of the sets K_1 and K_2, and the uncertainty function *cert*.

As explained in the previous section, types model information items which correspond to terms; consequently, constraints are semantic relationships between terms. These unconditional and conditional constraints, together with the background conditions and the uncertainty function associated to the conditional constraints, are extracted from thesauri. The type of thesaurus required is described in 6.3.1. The actual thesaurus used in this implementation is described in 6.3.2. The construction of the constraints from that thesaurus is explained in 6.3.3.

6.3.1 Thesauri

In the English Oxford encyclopedia, thesauri are defined as "*a 'treasury' or 'storehouse' of knowledge, as a dictionary, encyclopedia, or the like*" or as "*a collection of concepts or words arranged according to sense*". Therefore, in general, thesauri aim to store semantic relationships that associate terms with one another.

There are two kinds of associations between terms: *first order* and *second order*. First order associations are defined as terms that often co-exist within some predefined boundaries (e.g., text, paragraph, or sentence). Second order associations are defined as terms that are semantically related (e.g., synonyms, broader terms, narrower terms or related terms).

The use of first order associations to capture semantic relationships has been proven unsatisfactory in IR (see [Cro90, Rug92, MMN83, Kra91]). Indeed, first order associations are usually derived from statistical methods or statistical methods coupled with syntactic analysis performed on the document collection itself. Although the outcomes allow the determination of quantified relationships, poor results are often obtained because terms are judged to be related when they are not. Therefore, in this thesis, the implementation of the semantic relationships is carried out using second order associations.

The determination of second order associations is usually carried out manually. Indeed, with statistics-based methods, the relationships are identified on the basis of frequent co-occurrence of terms (examples of methods are discussed in [Kuh64, SB64]). However, it is rare that two synonyms are used in the same documents, so the relationship between these synonyms cannot be identified based on their frequency of co-occurrence. The benefit that comes from a manually-built thesaurus is that very few false relationships are produced since the process is done by human experts (a discussion on thesaurus construction and some examples can be found in [Sri91, Bla90, Pac91, Bru89, BGLY86, Den64]).

In this thesis, an on-line thesaurus that stores manually-built second order associations is used to derive the constraints which constitute the knowledge set. The thesaurus is known as WordNet™ (Version 1.5) [Mil90]. In this way, correct relationships are provided; the remaining tasks are to select those which form constraints, to determine their background conditions, and to quantify the uncertainty of their use.

6.3.2 The WordNet thesaurus

WordNet is a general thesaurus that covers conventional English and a wide range of technical terms. WordNet only stores the base form of terms. For example the base form of "augmenting" is "augment". The WordNet library contains functions for searching the WordNet database and for applying a morphological process to the search of strings. The purpose of these functions is to find a base form that is present in WordNet only if that word does not exist as such. There are exception lists which contain the morphological transformations for words that are not regular and therefore cannot be found. For example, the base form of "children" is "child".

An entry in WordNet can be either a word or a *collation*. A collation is a string of two or more words connected by spaces or hyphens. In further references, a "term" refers to a "word" or a "collation", unless otherwise specified.

WordNet stores four parts of speech: *nouns*, *verbs*, *adjectives* and *adverbs*. As with many IR systems, only *nouns* are used in the implementation because they usually contain most of the information expressed in a sentence.

WordNet encapsulates different categories of semantic relationships. Five of them are considered in this implementation: *synonymy, hypernymy, hyponymy, holonymy and meronymy*. The terms defined by these categories are respectively:

(i) *synonym*;

(ii) *hypernym*, which is a generic term used for a whole class of specific instances;

(iii) *hyponym*, which is a specific term used to designate a member of a class;

(iv) *meronym*, which is a name of a constituent part of, the substance of, or a member of something;

(v) *holonym*, which is a name of a whole of which the meronym is a part.

WordNet takes into account the *polysemic* nature of terms by organizing them into logical groupings of terms called *synsets*. For example, the WordNet synonym entry of the term "horse" is

Synonym
Sense 1 : sawhorse, horse, sawbuck, buck
Sense 2 : knight, horse
Sense 3 : horse
Sense 4 : horse, Equus caballus

Figure 6.1: Example of synonyms in WordNet

In WordNet, "horse" has four senses (meanings). The synonym entry of "horse" has four synsets, one for each of the four senses. Below, an example of a relationship for each of the other categories is given with respect to the term "horse".

Hypernym	
Sense 1 :	framework, frame
Sense 2 :	chessman, chess piece
Sense 3 :	gymnatsic apparatus, exerciser
Sense 4 :	equine, equid

Hyponym	
Sense 1 :	trestle
Sense 3 :	pommel horse, side horse
	vaulting horse, long horse, buck
Sense 4 :	foal
	stallion, entire
	gelding
	saddle horse, riding horse, mount

Figure 6.2: Example of hypernyms and hyponyms in WordNet

Meronym	
Sense 4 :	(MEMBER OF) Euquus, genus Equus

Holonym	
Sense 4 :	(HAS PART) mane
	(HAS PART) withers

Figure 6.3: Example of meronyms and holonyms in WordNet

For example, the hypernyms of the term "horse" is given as a synset for each of the four senses of that term.

Aside from the various senses the entry term may possess, each term in a synset possess a sense[77]. Although this sense is not displayed, it can be determined from WordNet.

6.3.3 Construction of constraints

Relationships of each of the five categories (i.e., synonymy, hypernymy, hyponymy, holonymy and meronymy) are used to implement the constraints and the uncertainty function attached to the conditional constraints. Relationships of all categories can be amalgamated to model the constraints and the uncertainty function.

6.3.3.1 Synonym-based constraints

The synonyms in WordNet are organized into synsets. Let t be a term of WordNet. Let $\{t_1, \ldots, t_k\}$ be a synset of the synonym entry of the term t. For any term t' in that synset, $t \rightarrow t'$ constitutes a constraint. Whether that constraint is unconditional or conditional depends on the number of senses the term t has in WordNet. Several cases occur:

(i) t has a single sense; that is, it is not polysemic. In that case, the constraint $t \rightarrow t'$ always holds; thus, it is unconditional.

(ii) Otherwise, the constraint is conditional, for its application depends on the sense of t. Let S_t be the sense of t in the constraint $t \rightarrow t'$. In that case, the constraint holds with respect to a document if the sense of t in the document is S_t.

Therefore, constraints are conditional if their application depends on the senses attached to their

[77] To my knowledge, this sense is unique.

term antecedents. These senses act as the background conditions of the constraints. Therefore, in (ii), the complete representation of the constraint $t \rightarrow t'$ is

$$t \rightarrow t' | \{S_t\}$$

$\{S_t\}$ constitutes the background conditions of the constraint $t \rightarrow t'$, meaning that the constraint can be applied with certainty to a situation only if that situation supports t such that the sense of t in that situation is S_t.

The sense of t' in the constraint $t \rightarrow t'$ must be represented because a constraint with antecedent t' may be applied later. Let $S_{t'}$ be the sense of the term t' in the constraint $t \rightarrow t'$. To represent the sense of $t \rightarrow t'$, the constraint is written as

$$t \rightarrow [t', \{S_{t'}\}] | \{S_t\}$$

For simplicity, the above constraint is written (bearing in mind that $\{S_t\}$ constitutes the background conditions of the constraint)

$$[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$$

Also, for the sake of uniformity, an unconditional constraint is written

$$[t, \{\}] \rightarrow [t', \{S_{t'}\}]$$

The empty set is used because t is not polysemic. In the above two constraints, the sense associated to t' is also $\{\}$ if this term is non-polysemic in WordNet.

$[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$ and $[t, \{\}] \rightarrow [t', \{S_{t'}\}]$ implement a conditional constraint and an unconditional constraint, respectively. Therefore, $[t, \{S_t\}]$, $[t, \{\}]$ and $[t', S_{t'}]$ must correspond to types. The initial implementation of types given in section 6.2 is modified to reflect this fact. Indeed, section 6.2 defines an information item as representing a term, for instance t, and the type corresponding to the term t is denoted $'t'$ where

$$'t' = \left[\dot{d} | \dot{d} \models \ll present, t; 1 \gg\right]$$

With the above implementation of constraints, an item of information does not represent only a term only, but instead represents a term and its associated senses[78]. Let t be a term and let lS_t be the set of senses associated to t, either in a situation (this is explained in section 6.4.2) or in a constraint (in that case, lS_t is a singleton). The type representing this item of information is then implemented as

$$[t, lS_t] = \left[\dot{d} | \dot{d} \models \{\ll present, t; 1 \gg, \ll Sense, t, lS_t; 1 \gg\}\right]$$

If the term is not polysemic, then its corresponding type is implemented as

$$[t, \{\}] = \left[\dot{d} | \dot{d} \models \ll present, t; 1 \gg\right]$$

Uncertainty arises when it is not known whether the background conditions of a conditional constraint are satisfied by a situation. Since background conditions reflect senses, this uncertainty

[78] This captures, to a limited extent, intensionality.

represents the probability of a sense being the one referred to by the use of a term. Indeed, not all possible senses of a polysemic term are equally likely. If some information on the relative probability of the various senses could be obtained, the uncertainty could be divided to reflect this. Methods based on statistics [Hoe66, Klu74] or numerical taxonomy [CS57, SS73] could be used for that effect. Unfortunately, WordNet does not provide information to determine which sense of a term is used most frequently[79]. Other factors could be taken into account in the uncertainty (e.g., the number of common hyponyms). Since there is no obvious solution to this problem other than empirical, the following approach is adopted.

Let $\sharp t$ be the number of senses of the term t in a situation s. If $\sharp t > 1$, and S_t is among the set of senses of the term t in the situation s, then the uncertainty associated with the use of the constraint $[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$ with respect to the situation s is set to

$$cert(s, \{S_t\}) = 1/\sharp t$$

$1/\sharp t$ is used because it reflects the fact that the situation s may be extended to $\sharp t$ alternative situations (if no additional information is available). With this formulation of the function $cert$, the constraints (more precisely, their background conditions) with antecedent t that are used to extend the situation s are already normalized.

The determination of constraints and the uncertainty function based on the other types of relationships is discussed in the following sections. The method adopted for their determination is similar to that described for the determination of synonym-based constraints.

6.3.3.2 Hypernym-based constraints

Each relationship between a term and its hypernym in WordNet constitutes a hypernym-based constraint. Let $t \rightarrow t'$ be one of them. If the term t is not polysemic, then $t \rightarrow t'$ constitutes an unconditional constraint. It is then denoted as $[t, \{\}] \rightarrow [t', \{S_{t'}\}]$. Otherwise, $t \rightarrow t'$ constitutes a conditional constraint, and is then written $[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$. $\{S_t\}$ are the background conditions of the constraint.

The method which determines the uncertainty of hypernym-based conditional constraints is the same as the one used for synonym-based constraints. It is also based on the number of senses a term has in a situation[80].

6.3.3.3 Hyponym-based constraints

A hyponym is a specific term which designates an instance of a class. For example, some of the hyponyms of sense 1 of "car" in WordNet are:

[79] In WordNet (Version 1.5), the synsets of a term are displayed in increasing order of the frequency of their senses. However, there is no information telling how more often a sense of a term is used instead of another. The quantification of this ordering is not an obvious task. Also, the ordering may not be appropriate for all document collections. For this reason, this implementation ignores this feature of WordNet.

[80] In WordNet, the number of senses of a term is the same for each category of relationships, although a term may not have, for example, a hypernym for each of its possible senses.

Sense 1 :	cab, hack, taxi, taxicab jeep, landrover limousine, limo

Figure 6.4: Hyponyms of "car"

Each relationship between a term and its hyponym constitutes a constraint. However, from a document which contains the term "car", it cannot be automatically inferred that the hyponym referred to by "car" is, for example, "cab" and not "limousine". This indetermination is not due to the polysemic nature of "car", but it may be solved from the document itself; for example, if the latter mentions "cab". However, it is erroneous to assume that if "cab" appears in the document then no other hyponym can be referred to by "car". Therefore, the existence of a hyponym in a document does not preclude the existence of other hyponyms in that document. For this reason, in a document which mentions "car", both the hyponyms (with respect to sense 1) "cab" and "limousine" are implicit. Therefore, the use of hyponyms to construct constraints and their uncertainty is the same as for synonyms.

6.3.3.4 Holonym-based constraints

A holonym is the name of a whole to which a term is a part, a member, or a substance. The uncertainty in a holonym relationship is due only to the polysemic nature of terms. The types of holonyms (members of, parts of, substance of) in this implementation are not distinguished because this is beneficial only if a rigorous linguistic process is performed. The determination of holonym-based constraints and their uncertainty is the same process as described for synonym-based constraints.

6.3.3.5 Meronym-based constraints

A meronym is a part of something. It follows that mentioning a term implies that its meronyms are implicitly mentioned. Uncertainty arises when the term is polysemic. The treatment of meronyms is the same as for synonyms. As for holonyms, the types of meronyms are not distinguished.

6.3.3.6 Combined constraints

Different categories of relationships of WordNet have been used independently to define the knowledge set. Relationships of the different categories can be used jointly to define the knowledge set. A constraint that was originally unconditional or conditional stays unconditional or conditional, respectively. It may be that some relationships are only defined for some senses of a term. For example, the meronym entry of "horse" is only defined for sense 4 (see section 6.3.2). Although one meronym synset is involved, the resulting constraint is conditioned with the sense 4 of "horse".

An approach in which no distinction is made between the different categories of relationships is proposed in [RSM94]. There, the uncertainty associated between any two terms is based on the number of hypernyms the two terms have in common, and the depth between these hypernyms and the two terms. This approach, however, does not provide an implementation of the background conditions, so it is not followed in this thesis.

6.3.4 Conclusion

The implementation of the constraints is based on an existing on-line thesaurus, known as WordNet. The appropriateness of the constraints is ensured because WordNet stores (hopefully) correct relationships. The polysemic nature of WordNet terms is used to define whether a constraint is unconditional or conditional. An unconditional constraint has the general form $[t, \{\}] \rightarrow [t', \{S_{t'}\}]$ and a conditional constraint has the general form $[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$. $\{S_{t'}\}$ corresponds to the sense of t' in the two constraints. $\{S_t\}$ constitutes the background conditions of the latter constraint. The uncertainty attached to the conditional constraint in a situation is based on the possible senses of the term t in that situation.

In the next section, the implementation of the unstructured model is described. The knowledge set of the unstructured model can be implemented by any of the processes described in the six previous sections. In the remainder of this chapter, the knowledge set is referred to simply as the set of unconditional constraints K_1 and the set of conditional constraints K_2. That is, it is assumed any category of relationships can be used, or indeed a mixture of them.

6.4 Implementation of the unstructured model

The unstructured model involves types, constraints and situations. Types model the information items, which correspond to terms. Not all possible terms are considered in this implementation. The selection of terms is described in 6.4.1. The implementation of the constraints was discussed in section 6.3. The unstructured model defines a root situation and extended situations. The root situation models the document's initial information content and the extended situations model the document's extended information content. The latter situations result from the extension of the root situation, where extension models the flow of information. The implementation of a root situation and the implementation of an extended situation are depicted in 6.4.2 and 6.4.3, respectively. The section 6.4.3 also describes the extension process. Examples illustrating the implementation of situations are given in section 6.4.4. Queries are modelled as sets of types. The handling of queries is dealt with in section 6.4.5. The implementation of the remaining components of the unstructured model is discussed in 6.4.6.

6.4.1 Selection of terms

In WordNet, a noun, or more correctly *noun-phrases*, is a word or a collation (i.e., a set of words), for example, "information" and "abdominal nerve plexus", respectively. Although it may have been advantageous to use the fact that WordNet stores collations, it is not obvious that doing so would enhance the IR system performance. Indeed, the use of noun-phrases in a document's representation has not yet been proven very successful in IR (see [LL93, Fag87, Sme88]). Furthermore, the determination of appropriate noun-phrases would require a robust syntactic and semantic analysis because WordNet does not store all the different forms of a noun-phrase. For example, "art exhibition" is stored in WordNet whereas "exhibition of art" is not (the problem in obtaining the canonical form of a noun-phrase is comprehensively discussed in [Sme92]). Finally, even if the determination of the appropriate noun-phrases was possible, it is not known what maximum length a noun-phrase should be. Indeed, a WordNet collation can contain as many as 4 words,

so finding the appropriate collation is not an easy task. Suppose that the noun-phrase "amygdalus communis amara" was extracted from a text document. It is not known whether "amygdalus communis amara" leads to an improved document's representation then "amygdalus communis", because both "amygdalus communis" and "amygdalus communis amara" are WordNet entries (this issue was extensively discussed in [Fag87]). For this reason, only words are considered in this implementation. In the above example, this means that the three words forming the noun-phrase "amygdalus communis amara" are used individually as the document's representation. In further references, words are simply referred to as terms.

6.4.2 Implementation of a root situation

In the unstructured model, the representation of a document's initial information content is a situation that supports the types which model the information items that are

(i) explicitly extracted from the document's information content,
(ii) derived from the application of unconditional constraints, or
(iii) derived from the application of conditional and certain constraints.

The implementations of these three processes are described in the following sections.

6.4.2.1 Types extracted from the text document

Single (noun) terms are extracted from the text document, and are submitted to a process that removes stop words (e.g., "about", "in", "best", etc). The removal of stop words uses the stop list given in [vR79].

After the removal of stop words, the remaining terms are submitted to a stemming process based on WordNet. The stemming process used in most IR systems is the Porter algorithm [Por80]. This algorithm cannot be used in this implementation because it may output stems that are not WordNet terms. For example, with the Porter algorithm, "connection", "connect" and "connections" are all stemmed into "connect", which is not a noun in WordNet. With the WordNet stemming process, a term is transformed into its base form only if it does not appear in WordNet. For example, "accounts" is transformed into "account" in WordNet, but "accounting" is not. Exceptions are also captured by the WordNet stemming. For example, "children" is stemmed into "child"[81].

A term extracted in the document may not exist in WordNet. That is, the term is neither a noun, a verb, an adjective nor an adverb. The term is kept because WordNet does not cover all possible nouns. Such terms will be referred to as *proper nouns.*

A term which results from the stemming process may be ambiguous; that is, the term is polysemic (it has several senses) and the sense of that term in the document is unknown. Knowing the sense of a polysemic term is important because the conditional constraints which hold with respect to that term can then be identified. However, disambiguating a term is not always possible, and it has not yet been proven beneficial in IR (see [KC92, Voo93, San94]). For this reason, no disambiguation

[81] There is, however, a disadvantage with the use of the WordNet stemming. In WordNet, many variations of the same term can appear. For example, the terms "account", "accounting" and "accountant" all appear in WordNet, so the latter two are not stemmed into "account" as they would have been with the Porter algorithm. It is not certain whether the distinction between the three terms is always necessary. This problem is more obvious in the following example. Both "follower" and "followers" are stored in WordNet. It is not sure whether the occurrence of "followers" in a document is as the plural of "follower", or as "followers" itself.

is attempted in this implementation; a term is assigned all the different senses that it possesses in WordNet. Let t be a term that results from the WordNet stemming process. Let lS_t be the senses of the term t in WordNet. The type which models this term is implemented as

$$[t, lS_t]$$

If the term t is not polysemic (the term t has a unique sense), or it is not a WordNet term, then the corresponding type is implemented as

$$[t, \{\}]$$

The types as above implemented are supported by the root situation. Let s be that situation. Then

$$s \models [t, lS_t] \quad \text{or} \quad s \models [t, \{\}]$$

A situation is implemented as the union of types, which are implemented as terms with their associated senses in that situation[82]. For simplicity, in future references, the implementation of the types and the types themselves are not distinguished. The same applies for a situation and its implementation. In addition, any term t such that $s \models [t, lS_t]$ is said to be contained in the situation s, where lS_t can be the empty set.

6.4.2.2 Types coming from unconditional constraints

Unconditional constraints are concerned with non-polysemic terms contained in a situation. Let s be a situation. Let t be a non-polysemic term such that $s \models [t, \{\}]$. If $[t, \{\}] \rightarrow [t', \{S_{t'}\}]$ is an unconditional constraint of the knowledge set, then the constraint can be applied to s and therefore

$$s \models [t', \{S_{t'}\}]$$

If the term t' is contained in the situation s, then there exists a set of senses $s_{t'}$ associated with t' such that $s \models [t', s_{t'}]$. If the sense of t' in the constraint is among the senses of t' in the situation, then the application of the constraint $[t, \{\}] \rightarrow [t', \{S_{t'}\}]$ to the situation s has no effect[83]. Otherwise, the application of the constraint to the situation s brings an additional sense to t' in s; $s \models [t', s_{t'} \cup \{S_{t'}\}]$. The same reasoning applies for the application of conditional constraints that lead to terms already contained in a situation.

6.4.2.3 Types coming from conditional and certain constraints

Conditional and certain constraints are concerned with unambiguous polysemic terms contained in a situation. Let $s \models [t, \{s_t\}]$ where t is a term and s_t is the sense of the term t in the situation s (an unambiguous term has a single sense). Let $[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$ be a conditional constraint of the knowledge set. The application of this constraint to the situation s is certain if $S_t = s_t$; that is, the sense of t in the constraint and in the situation is the same. The effect of the application of the constraint is that $s \models [t', \{S_{t'}\}]$.

The union of the types which

(i) model information items extracted from the text document,
(ii) are derived from unconditional constraints, or
(iii) are derived from conditional and certain constraints,

[82] The representation of a situation by a set was discussed in [Bar89, Dev91], in which this set was referred to as an abstract situation. As often done in mathematics, abstraction was necessary to study situations in general.

[83] This conforms to the discussion carried out in Chapter 4, section 4.4.2.4. The application of an unconditional constraint $\phi \rightarrow \chi \in K_1$ or a conditional constraint $\phi \rightarrow \chi | B \in K_2$ to a situation s has no effect when $s \models \{\phi, \chi\}$.

is used to implement the root situation.

6.4.3 Implementation of a situation that results from an extension

A situation can be extended into another situation from the application of conditional and uncertain constraints. The extension of a situation results from the application of one or several conditional and uncertain constraints. The use of a single constraint is discussed in section 6.4.3.1. The use of a group of constraints is discussed in 6.4.3.2. The uncertainty attached to the extended situation is discussed in section 6.4.3.3.

6.4.3.1 Use of a single constraint

A situation can be extended to another situation if a conditional and uncertain constraint can be applied to that situation. Conditional and uncertain constraints are concerned with ambiguous terms. Let t be an ambiguous term that is contained in a situation s, and let s_t be the set of senses associated to the term t in the situation s (i.e., $s \models [t, s_t]$). t is ambiguous in s, so the set s_t is not a singleton nor the empty set. Let $[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$ be a conditional constraint of the knowledge set. The application of this constraint to the situation s is uncertain if the sense of the term t in the constraint is among the possible senses of the term t in the situation s (i.e., $S_t \in s_t$); that is, the background conditions of the constraint (i.e., $\{S_t\}$) may or may not be satisfied by the situation s. The application of the constraint to the situation s leads to the extension of that situation.

Let s' be the situation that results from the application of the constraint $[t, \{S_t\}] \rightarrow [t', lS_{t'}]$. This situation supports

(i) the type $[t', \{S_{t'}\}]$[84].

(ii) the types that were originally supported by s,

(iii) the fact that the sense of t in s' is S_t,

(iv) the types derived from the application of unconditional constraints and the application of conditional and certain constraints in s' (as described in sections 6.4.2).

Case (iii) means that the extension process as implemented in this thesis is also viewed as a reduction of ambiguity; during the extension, a term become unambiguous.

6.4.3.2 Use of a group of constraints

The application of two or more constraints to a situation s can result into one situation or several alternative situations, depending on whether or not their background conditions are compatible. Let $[t_1, \{S_1\}] \rightarrow [t'_1, \{S_{t'_1}\}]$ and $[t_2, \{S_2\}] \rightarrow [t'_2, \{S_{t'_2}\}]$ be two conditional constraints. In this implementation[85], the fact that the two terms t_1 and t_2 may be semantically related is not taken into account. As a result, the sense of the term t_1 is independent to the sense of t_2. The applications of the two constraints to the situation s, and their background conditions, are independent from each other. Consequently, only the following two cases occur regarding the applications of the constraints:

[84] If the term t' is already contained in the situation s, the application of $[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$ to s will add the sense $S_{t'}$ to the senses of t' in s.

[85] This issue was discussed in Chapter 4, section 4.6.

(i) If the antecedents of these constraints refer to the same term (i.e., $t_1 = t_2$), then the background conditions of these constraints is incompatible if $S_1 \neq S_2$. In that case, the application of the constraint leads to alternative situations.

(ii) Otherwise, the background conditions of these constraints are compatible, and the constraints are used conjointly to extend the situation s into one situation.

The effect of the application of the these constraints is the same as described in the previous section.

6.4.3.3 Uncertainty of extension

Let a situation s be extended into a situation s'. The uncertainty attached to the extended situation s' is computed from the uncertainty of the constraints leading to that situation and the uncertainty that is attached to the situation s. The formula was given in Chapter 4. The computation of the uncertainty attached to the extended situation is a direct outcome of its formulation. Therefore, it does not need further discussion. What remains to be discussed is the normalization of the constraints. As described in section 6.3.3.1, the normalization is already captured in the expression of *cert* in this implementation.

6.4.4 Examples

Suppose that the WordNet synonym entries of the terms "dog" and "horse" are respectively[86]

Sense 1 : pawl (1), bounder (1) **Sense 2** : firedog (1)

Figure 6.5: WordNet synonyms of "dog"

Sense 1 : sawbuck (1) **Sense 2** : knight (2)

Figure 6.6: WordNet synonyms of "horse"

The senses of the different synonyms are displayed between brackets. The following five synonym-based constraints are extracted from these entries:

(1) $['dog', \{1\}] \rightarrow ['pawl', \{1\}]$

(2) $['dog', \{1\}] \rightarrow ['bounder', \{1\}]$

(3) $['dog', \{2\}] \rightarrow ['firedog', \{1\}]$

(4) $['horse', \{1\}] \rightarrow ['sawbuck', \{1\}]$

(5) $['horse', \{2\}] \rightarrow ['knight', \{2\}]$

The result of the application of some of these constraints to a situation s are shown in the following table (the uncertainty associated with each extended situation is shown between brackets):

[86] These entries are not complete, but are sufficient for the purpose of these examples.

	The types supported by s	Constraints	The results
I	$['horse', \{2\}]$	(5)	$s \models ['knight', \{2\}]$
II	$['dog', \{1\}]$	(3)	Cannot be applied
III	$['dog', \{1,2\}]$	(1) (2)	$s' \models \{['pawl', \{1\}], ['bounder', \{1\}]\}$ (0.5)
IV	$['dog', \{1,2\}]$	(1) (3)	$s' \models ['pawl', \{1\}]$ (0.5) $s'' \models ['firedog', \{1\}]$ (0.5)
V	$['dog', \{1\}]$	(1) (2)	$s \models \{['pawl', \{1\}], ['bounder', \{1\}]\}$
VI	$['horse', \{2\}], ['knight', \{2\}]$	(5)	No effect
VII	$['dog', \{1,2\}], ['horse', \{1,2\}]$	(3) (5)	$s' \models \{['firedog', \{1\}], ['knight', \{2\}]\}$ (0.25)
VIII	$['dog', \{1\}], ['pawl', \{2\}]$	(1)	$s \models ['pawl', \{1,2\}]$

Table 6.3: Examples of the results of the application of the implemented constraints

In row I, the constraint (5) is conditional and certain with respect to the situation s because the background conditions of the constraint are satisfied by the situation s. The application of the constraint leads to additional information about s. In row II, the constraint (3) cannot be applied because its background conditions are not satisfied by the situation s. In row III, the application of the constraints (1) and (2) lead to one situation s' because (i) the antecedents of these constraints are supported by s, (ii) the background conditions of these constraints may or may not be satisfied ("dog" is ambiguous in the situation s), and (iii) these background conditions are compatible. In row IV, the constraints (1) and (3) are uncertain with respect to s. The background conditions of these constraints are incompatible, so two situations result from the application of these constraints, one situation for each constraint. In row V, the constraints (1) and (2) are both certain with respect to the situation s. Their application delivers additional information about s. In row VI, the term "knight" with sense 2 is already supported by the situation s, so the application of the constraint (5) has no effect. In row VII, the two constraints are uncertain with respect to the situation s. Their application leads to one situation because their background conditions are compatible. Finally, in row VIII, the application of the constraint (1) adds an additional sense to "pawl".

6.4.5 Implementation of queries

In the unstructured model, a query is modelled by a set of types. Hence, it is implemented as a set of terms and their associated senses. The determination of these terms is the same process as that performed on documents. The terms used in a query can also be ambiguous. No disambiguation is performed, so a term is associated with all its WordNet senses. The empty set is associated with terms that are either not in WordNet, or not polysemic. With an interactive IR system, a user when entering his or her query, may be ask to specify the senses of some of the terms used in his or her query. In both cases, the analysis of a query results in a list of types of the form $[t, lS_t]$.

6.4.6 Remaining components of the unstructured model

The implementation of the remaining components of the unstructured model are discussed in this section. These are the sequential extension of situations, the propagation and aggregation of uncertainty, and the computation of the relevance degree.

6.4.6.1 Sequential extension of situations

The flow of information extends the root situation into other situations, which are then extended into other situations, and so forth. A sequential extension of situations is called a branch. The sequential extension of situations ceases when a situation (known as a leaf) is obtained such that

(i) it supports the information being sought, or
(ii) no more unused constraints can be employed to extend that situation.

In the first case, a minimal branch is defined; its leaf situation is pertinent to the query. In the second case, a maximal branch is constructed; its leaf situation is non-extendible. The implementation of minimal and maximal branches requires the implementation of the extension of a situation (this was discussed is section 6.4.3), a pertinent situation and a non-extendible situation.

6.4.6.1.1 Pertinent situation

A situation s is pertinent to a set of types Φ if the situation supports at least one type in the set. In this implementation, a situation s is pertinent to a set of types Φ if

(i) there exists a type $[t, s_t]$ such that $s \models [t, s_t]$, and
(ii) there exists $[t, q_t]$ in Φ such that at least one of the senses of t in the situation s is compatible with one of the senses of t in the query (i.e., $s_t \cap q_t \neq \emptyset$).

The reason for this is because a type $[t, s_t]$ can be viewed as $\sharp s_t$ types of the form $[t, \{S_t\}]$ for each sense S_t in s_t, where $\sharp s_t$ is the number of senses of t in s. The same applies for the types representing the query. Therefore, the fact that $s_t \cap q_t \neq \emptyset$ means that one of the types supported by the situation s is among one of the types modelling the query.

6.4.6.1.2 Non-extendible situation

A situation s is non-extendible if for all types $[t, s_t]$ supported by s and for all conditional constraints $[t, \{S_t\}] \rightarrow [t', \{S_{t'}\}]$, any of the following occur:

(i) the constraint has already been applied, or the application of the constraint has no more effect (this means in both cases that there is some $s_{t'}$ such that $s \models [t', s_{t'}]$ and $S_{t'} \in s_{t'}$).
(ii) the background conditions of the constraint are not satisfied by the situation (i.e., $S_t \notin s_t$).

6.4.6.2 Propagation and aggregation of uncertainty

The implementation of the propagation and the aggregation of uncertainty is a direct result of their formulations. The uncertainty of an extended situation is defined as the multiplication of the uncertainty of the situation that is extended to that situation with the uncertainty of the normalized constraints used in the extension. If several situations are extended to one situation, the uncertainty that results from each extension is aggregated. The uncertainty of the extended situation is defined as the summation of the uncertainty attached to the obtainment of that situation from each of the other situations.

6.4.6.3 Computation of the relevance degree

The relevance degree of a document to a query is the summation of the uncertainty attached to the obtainment of minimal branches from the root situation. The implementation of the root situation, the minimal branches and their uncertainty was discussed in sections 6.4.2, section 6.4.6.1 and section 6.4.6.2, respectively. Therefore, the implementation of the computation of the relevance degree of a document to a query does not need further discussion.

6.5 The implementation of the structured model

The structured model caters to a semantic-based structured representation of a document. The implementation of the structured model requires the implementation of the weighted information domain which models the representation of a document's information content (section 6.5.1), and the implementation of the refinement function which models the transformation of a document (section 6.5.2). The implementation of the remaining components of the structured model is discussed in section 6.5.3.

The structured model uses many concepts defined in the unstructured model, such as types, situations, the extension of a situation, the representation of queries and the representation of constraints. The implementation of these concepts was discussed in sections 6.4 and 6.3.

6.5.1 Implementation of the weighted information domain

A weighted information domain is defined as $D = \langle T_D, S_D, m_D, Bel_D \rangle$. The implementation of the weighted information domain requires the implementation of T_D, the set of types; S_D, the set of basic situations; m_D, the basic probability assignment (BPA); and Bel_D, the belief function. The implementation of the set of basic situations is discussed in section 6.5.1.1. T_D is the set of types that are supported by the basic situations. The implementation of the set of types is derived from the implementation of the basic situations, and is also discussed in section 6.5.1.1. The BPA attaches a weight to each basic situation, and reflects the significance of the information supported by that basic situation with respect to the document's overall information content. The implementation of the BPA is discussed in section 6.5.1.2. The belief function Bel_D is a measure of the relevance of the information supported by the basic situations to a query, the implementation of which is discussed in section 6.5.1.3.

6.5.1.1 Basic situations

In the unstructured model, the representation of a document's initial information content is modelled by a root situation, for instance d. The types supported by this situation represent the explicit, and the implicit and certain information content of a document. In the structured model, the representation of a document's initial information content is modelled by a set of basic situations. A basic situation s is a semantic structure with a single semantic content, for instance, $s \models \varphi$. That is, for all types ψ supported by s, there is a constraint $\varphi \rightarrow \psi$ either unconditional, or certain and conditional. The determination of the basic situations requires the determination of the semantic contents of the document.

The determination of the semantic contents of a document necessitates the identification of those types supported by d that cannot be obtained from the certain application of a constraint to d. Such types are those that do not appear as a consequent of a constraint that can be applied with certainty to the situation d. Therefore, only types that model terms explicitly extracted from the document may lead to semantic content.

Let t be a term extracted from the document and let d_t be its associated senses in d (i.e., $d \models [t, d_t]$). Let $[t', \{S_{t'}\}] \rightarrow [t, \{S_t\}]$ be a constraint which can be applied to d. Whether $[t, d_t]$ can be obtained from the application of the constraint $[t', \{S_{t'}\}] \rightarrow [t, \{S_t\}]$ depends on the following conditions:

(i) the sense of t in the situation is S_t (i.e., $d_t = \{S_t\}$). In that case, $[t, d_t]$ can be obtained from the application of $[t', \{S_{t'}\}] \rightarrow [t, \{S_t\}]$ to d.

(ii) S_t is not among the possible senses of t in d (i.e., $S_t \notin d_t$). In that case, the type $[t, d_t]$ cannot be obtained from the application of $[t', \{S_{t'}\}] \rightarrow [t, \{S_t\}]$ to d.

(iii) S_t is among the possible senses of t in d but t has other senses in d. Let the set $s_t = d_t \backslash \{S_t\}$ be these senses. The type $[t, s_t]$ cannot be obtained from the application of the constraint $[t', \{S_{t'}\}] \rightarrow [t, \{S_t\}]$ to d, whereas the type $[t, \{S_t\}]$ can be obtained from this application.

In case (iii), to determine the types that cannot be obtained from the application of a constraint, it may be necessary to decompose a type into two *sub-types*. These two sub-types include one that can be obtained from the application of a constraint, and one that cannot be obtained from the application of a constraint. The decomposition is done with respect to the senses attached to the term in that type.

Let $C(d)$ be the set of the types in the situation d that cannot be obtained from the application of a constraint to d. This set contains the types which lead to semantic contents. Let $NC(d)$ be the set of types that are supported by d but that are not in the set $C(d)$. This set contains the types which can be obtained from the application of a constraint to d[87]. The sets $C(d)$ and $NC(d)$ are used to construct the basic situations. The following algorithm is used:

Step 1— Let $[t, d_t]$ be a term of $C(d)$ (remove it from $C(d)$).

Step 2— If d_t contains several senses, a basic situation s is created such that $s \models [t, d_t]$. $s \models [t, d_t]$ constitutes a semantic content and s does not support other types (since none can be derived with certainty). Goto Step 7.

Step 3— If d_t is the empty set, then a basic situation s is created such that $s \models [t, \{\}]$. $s \models [t, \{\}]$ constitutes a semantic content. Goto Step 5.

Step 4— If d_t contains one sense D_t, a basic situation s is created such that $s \models [t, \{D_t\}]$. $s \models [t, \{D_t\}]$ constitutes a semantic content.

Step 5— Let $[t', d_{t'}]$ be a term of $NC(d)$.

5.1—No constraint with consequent of the form $[t', \{S_{t'}\}]$ can be applied to s. In that case, $[t', d_{t'}]$ is not included in the basic situation s.

[87] If the type $[t, lS_t^1]$ is in $C(d)$ and the type $[t, lS_t^2]$ is in $NC(d)$ then $lS_t^1 \cap lS_t^2 = \emptyset$ and $lS_t^1 \cup lS_t^2 = d_t$, where d_t is the set of senses of t in d.

5.2—A constraint with consequent of the form $[t', \{S_{t'}\}]$ can be applied to s. Whether $[t', d_{t'}]$ is included in the basic situation s depends on the relationships between $d_{t'}$ and $S_{t'}$. Two relationships exist:

5.2.1— $S_{t'}$ is not among the senses of t' in $d_{t'}$. In that case, $[t', d_{t'}]$ is not supported by s.

5.2.2— The sense of t' in the constraint is among the senses of t' in $d_{t'}$. If the term t' is not already contained in s, then $s \models [t', \{S_{t'}\}]$. Otherwise, the sense $S_{t'}$ is added to the senses t' already contains in s.

Step 6— Repeat Step 5 for all types in $NC(d)$.

Step 7— If $C(d)$ is not empty, Goto Step 1. Otherwise, exit.

The set of types T_D is the union of all the types supported by the basic situations.

6.5.1.2 Basic probability assignment

In the weighted information domain D, the BPA m_D is associated with the set of basic situations S_D to reflect the significance of information. Let s be a basic situation. $m_D(s)$ measures the significance of the information supported by the situation s with respect to the document's overall information content. The information supported by a situation is modelled as types, implemented as terms with their associated senses. Therefore, the degree of significance $m_D(s)$ is dependent on the significance of the terms contained in the situation s.

The significance of a term can be computed from the frequency of that term in the document. Let $F(t)$ be the frequency of a term t in the document. Only the terms that are explicit in the document are considered. Otherwise, $F(t)$ would have to take into account the frequency of the terms in which t is implicit. Suppose that t is implicit in t' (i.e., the two terms constitute a constraint). If the term t' appears n times in the document, then the term t appears at least n times in the document, although eventually implicitly. Such an approach leads to higher frequency of a term that is implicit in many terms. It is not sure that this approach reflects correctly the significance of a term in the document. Therefore, it is not adopted.

The significance of a term contained in a situation should also take into account the number of senses associated to that term in that situation. Indeed, two situations can contain the same term, but the senses associated with the term are different in the two situations. If the term has more senses in one situation than it has in the other, then the term can be viewed as more significant in the first situation than in the second.

Let $T(s)$ be the set of terms explicitly extracted from the text document and contained in the situation s. A formulation of $m_D(s)$ which takes into account both the frequency of terms contained in the situation s and the senses associated with these terms is

$$m_D(s) = \frac{\sum_{t \in T(s)} F(t) * \sharp s_t}{\sum_{s \in S_D} \left(\sum_{t \in T(s)} F(t) * \sharp s_t \right)}$$

$\sharp s_t$ is the number of senses of a term t in the situation s. This formula is similar to that given in [dSM93], where a Dempster-Shafer based IR model is described. The denominator ensures that

m_D is a BPA, that is

$$\sum_{s \in S_D} m_D(s) = 1$$

6.5.1.3 Belief function

Given a set of types Φ modelling the query, $Bel_D(\Phi)$ is defined as the summation of the BPA associated to the basic situations in S_D pertinent to the set Φ. The implementation of the belief function Bel_D requires the implementation of the basic situations, the BPA associated with these basic situations, pertinent situations and queries. The implementation of the basic situations and the BPA was discussed in the previous two sections, and the implementation of pertinent situations and queries was discussed in section 6.4.

6.5.2 Refinement

The transformation of a document is modelled as the refinement of the weighted information domain that constitutes a semantic-based structured representation of that document. Let $D_1 = \langle T_1, S_1, m_1, Bel_1 \rangle$ and $D_2 = \langle T_2, S_2, m_2, Bel_2 \rangle$ be two weighted information domains such that the latter is the refinement of the former. The refinement corresponds to the simultaneous extensions of the basic situations of D_1 into the basic situations of D_2. The extension of a basic situation is the same process as for the extension of a situation. This implementation of this process was described in section 6.4.3.

The BPA of a basic situation of the refined domain D_2 is defined in terms of both the BPA of the basic situations in D_1 that are extended into that situation, and the uncertainty of the conditional constraints leading to that situation. The computation of the BPA of a refined situation is the same process as for the computation of the uncertainty of an extended situation, which was described in 6.4.3.

6.5.3 The remaining component of the structured model

The last component of the structured model that has not yet been discussed is the computation of the relevance degree. The implementation of the relevance degree of a document to a query is based on the weighted information domains that result from the refinement process. The relevance degree is defined as the summation of the BPA of the pertinent situations in each information domain such that the situations become pertinent in that domain. The implementations of weighted information domains, refinement, and the BPA associated to information domains were discussed in the previous sections.

6.6 Conclusion

The implementations of two IR system models were described in this chapter. More specifically, both the unstructured and the structured models were provided with methods for implementing their related components. Critical to this chapter was the discussion of the implementation of the constraints. The different experiments and the evaluation of the models are discussed in the next chapter.

Chapter 7
Experiments and Evaluation

7.1 Introduction

This chapter describes the experiments and the evaluation of the unstructured and the structured models, which cater to an unstructured and a structured representation of information, respectively. The implementation of these two models was described in Chapter 6. The present chapter also investigates the appropriateness of the measure of exhaustivity and the measure of specificity defined in Chapter 5. It also examines whether the measure combining specificity and exhaustivity, also defined in Chapter 5, is an adequate measure of relevance.

The chapter contains 4 sections. The set up of the experiments is described in section 7.2. The results of the experiments and their analysis are discussed in section 7.3. Due to the results obtained, further experiments were carried out. Their set up, results and analysis are elaborated in section 7.4. The chapter finishes with a discussion in section 7.5.

7.2 Set up of the experiments

To perform experiments on the two models proposed in this thesis, the standard test collection originally gathered by Waswani and Cameron in 1970 at the National Physical Laboratory (NPL) is used. This collection contains 11429 documents, which are titles, and 93 queries, and comes with a relevance assessment (see [vRRP80] for a description of the collection and its construction). For computational factors, only the first 40 queries are used in the experiments.

The NPL collection was chosen to perform the experiments for two reasons. First, its documents are short; the maximal, minimal and average lengths of the documents are 105, 1 and 19.96 terms, respectively. This is essential because, in the models proposed in this thesis, a large amount of knowledge (i.e., the constraints are implemented as WordNet relationships) is used to determine a document's implicit information content, the computation of which increases with the size of the document. Second, the English language used in the NPL documents is not too technical. This is important because the WordNet thesaurus, used to implement the thesaurus, does not provide the kind of relationships imperative for technical collections.

Four sets of experiments were initially performed:

(i) one to evaluate the unstructured model.

(ii) one to evaluate the structured model, which as demonstrated in Chapter 5, indicates a measure of specificity.

(iii) one to evaluate the exhaustivity measure, described in section 5.8.2, Chapter 5. This measure will be said to be formulated by the *Exhaustive Model.*

(iv) one to evaluate the combination of the exhaustivity measure and the specificity measure, described in section 5.8.3, Chapter 5. This measure will be said to be given by the *Combined Model.*

The result of each experiment consists of a ranked set of documents for each of the 40 queries. An effective model should place the relevant documents at the top of the ranking and those less relevant lower in the ranking.

The set up of the four experiments is described in sections 7.2.1 through 7.2.4, respectively. The benchmarks used to compare the results are described in section 7.2.5. Finally, the evaluation method used to analyze the results of the experiments is explained in section 7.2.6.

7.2.1 The Unstructured Model

The aim of this first set of experiments was to determine whether more relevant documents were retrieved with the unstructured model than with conventional IR models. Most IR models only take into account the information explicitly extracted from documents. As a result, if a document and a query have no common terms, the document is not retrieved for that query. The unstructured model tries to remedy this problem by also taking into account the information implicit in documents.

The results of this set of experiments depend on the knowledge base used, here WordNet, to implement the constraints. That is, they depend on whether the WordNet thesaurus provides appropriate relationships to compute the NPL documents' implicit information content. An additional goal of this set of experiments was to determine which type of relationships (e.g., synonymy or hypernymy) determines best the documents' information content.

Five experiments were carried out, one for each type of WordNet relationships, which were synonymy, hypernymy, hyponymy, meronymy and holonymy (see Chapter 6). As explained in Chapter 6, the relationships were used to extend a document's initial representation, implemented as a set of terms with their associated senses, until information relevant to (terms in) a query was (were) found. Each term was used separately as the basis of an extension. Hence, the more terms a document had in its initial representation, the bigger its number of alternative extensions (see section 4.4.1.4 in Chapter 4 for an example of the extensions of a document). The terms in an extended representation of the document could also be the basis of further extensions, and so forth. This indicates that the number of alternative extensions with respect to the document's initial representation could be very large. This problem is known as the combinatorial explosion in the Artificial Intelligence world [RN95, Wat85]. To overcome it, a maximal depth was imposed on the number of extensions. This depth was arbitrarily set to 5.

7.2.2 The Structured model

The experiments to evaluate the structured model were performed in two phases: first, the

documents were structured and, second, the relevance degrees of these documents were computed for the 40 queries. The two phases were performed with each type of WordNet relationships (except for hyponyms for reasons given in section 7.3.2). For the same reasons explained in the previous section, a maximum depth of 5 extensions was imposed.

7.2.3 The Exhaustive Model

In Chapter 5, section 5.8, two measures were proposed, one that represents the degree of specificity of the document, referred to as $Spe(d, q)$, and one that represents its degree of exhaustivity, referred to as $Exh(d, q)$, where d is the situation representing the document and q is the set of types representing the query. The measure $Spe(d, q)$ is that calculated by the structured model. The aim of this set of experiments was to investigate the measure of exhaustivity $Exh(d, q)$. As for the previous two sets of experiments, each type of WordNet relationships were used, and a maximal depth of 5 extensions was imposed.

7.2.4 The Combined Model

The measures representing the degree of specificity of the document (i.e., $Spe(d, q)$) and the degree of exhaustivity of the document (i.e., $Exh(d, q)$) are combined together to express the degree of relevance of a document, the situation d, to a query, the set of types q, taking into account both specificity and exhaustivity. The formula is (see section 5.8.3, Chapter 5)

$$\frac{a * Spe(d, q) + b * Exh(d, q)}{a + b}$$

where a and b are factors reflecting the importance attached to specificity and exhaustivity, respectively. The higher a and b, the more importance is attached to specificity and exhaustivity, respectively. In this set of experiments, $a = 1$ and $b = 1$; both the specificity and the exhaustivity were equally important. The combination was made with respect to each type of WordNet relationships.

7.2.5 Benchmarks

To compare the results obtained with the different sets of experiments described in the four previous sections, two benchmarks were used:

(i) *Benchmark B1*: to compare with the unstructured model.

(ii) *Benchmark B2*: to compare with the structured model, and more importantly, the combined model.

With the benchmark B1, a document was established relevant to a query if the former contained a term that appeared in the query. That is, relevance was affirmed on the basis of the information explicit in the document. The comparison of the unstructured model to the benchmark B1 established whether relevant documents to a query that have no common terms with that query could also be retrieved. Note that a document with at least one common term with a query was also retrieved by the unstructured model (see section 4.4.1, Chapter 4). With the benchmark B1, a document and a query were represented by the set of terms appearing in them, d and q,

respectively. The relevance degree was given by

$$B_1(d,q) = \begin{cases} 1 & \text{if } d \cap q \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

The benchmark B2 was the standard vector space model [Sal71, SM80]. There, all documents and queries were N-dimensions vectors, where N was the number of terms in the document collection. The ith component of a vector was the weight of a term t_i in the document (or the query) modelled by that vector. For example, a document was represented by the vector $d = \langle w_{1,d}, \ldots, w_{N,d} \rangle$ where

$$w_{i,d} = freq(t_i, d) * idf(t_i)$$

$freq(t_i, d)$ was the occurrence frequency of term t_i in the document, and $idf(t_i)$, referred to as the inverse document frequency of the term t_i in the document collection, was computed by

$$idf(t_i) = \log \frac{D}{freq(t_i)}$$

$freq(t_i)$ was the number of documents in which the term t_i occurred and D was the number of documents in the collection. The same representation was adopted for a query, but a term occurrence frequency was within the query. The relevance of the document to the query was given by

$$B_2(d,q) = \frac{\sum_{i=1}^{N} w_{i,d} * w_{i,q}}{\sqrt{\sum_{i=1}^{N} (w_{i,d})^2} * \sqrt{\sum_{i=1}^{N} (w_{i,q})^2}}$$

The benchmark B2 was used because it captured both specificity, via each of the $w_{i,d}$s, and exhaustivity, via each of the $w_{i,q}$s, of the document to the query.

In the two benchmarks B1 and B2, as for the unstructured and the structured models, singles terms were extracted from documents, stop words were removed (the same stop list mentioned in section 6.4.2.1, Chapter 6, was used), and only nouns and proper nouns (terms that were neither nouns, verbs, adverbs or adjectives in WordNet) were taken into account. The stemming process was also based on WordNet (as described in section 6.4.2.1, Chapter 6). The reason being that a benchmark based on another stemming process (e.g., the Porter algorithm [Por80]) may have led to better/worse results due to the fact that the stemming process was different in the two methods. Here, it was the methods of retrieval that were compared.

7.2.6 Evaluation

The result of an experiment or a benchmark was a set of ranked documents for each query. A result was good if the documents relevant to a query were highly ranked and those less relevant were lower in the ranking. This was evaluated by computing the so-called *recall and precision* values[88] [vR79]. Different computations of these values have been defined (see [vR79]). In this

[88] The recall expresses the proportion of retrieved relevant documents with respect to all relevant documents. The precision expresses the proportion of retrieved relevant documents with respect to all retrieved documents (the formulations were given in Chapter 1, in section 1.1)

thesis, average precision values were calculated at standard recall values $10, \ldots, 100$ of percentage of relevant documents that were retrieved.

More precisely, for each query, pairs of precision-recall values were computed. The first pair of values was computed when the first relevant document was retrieved. Suppose that this document appeared at rank n_1, and that for the query, Q documents were assessed relevant (this information came from the relevance assessment provided by the test collection). In that case, the recall and the precision values were, respectively, $R_1 = 1/Q$ and $P_1 = 1/n_1$. For the second retrieved document, at position $n_2 > n_1$, the recall and the precision values were, respectively, $R_2 = 2/Q$ and $P_2 = 2/n_2$. A set of such pairs was obtained for each query, and was denoted $\{(R_i^k, P_i^k)\}$ for query number k.

To obtain the average precision values given a set of queries, the precision values must be given for the same values of recall. For this purpose, the above set was interpolated to standard recall values $10, \ldots, 100$ (%). For any point (r,p) of the set, the closest standard point value on the right of r was assigned the precision value p. If several of these points had the same closest standard recall value, the highest of the precision values was assigned to that closest recall value. The result of the interpolation was a set of precision values $\{P_{10}^k, P_{20}^k, \ldots, P_{100}^k\}$. The same process was applied for each query, and the overall precision values of the whole experiment at standard recall value $i = 10, \ldots, 100$ became

$$\widetilde{P}_i = \frac{\sum_{k=1}^{K} P_i^k}{K}$$

where K is the number of queries in the test collection.

7.2.7 Summary

The experiments carried out in this thesis are summarized in the table 7.1. For purpose of clarity, the experiments are named by a letter followed by a number. The letters U, S, E, C refer to the unstructured, the structured, the exhaustive and the combined models, respectively. The numbers refer to the type of WordNet relationships or the combination of types used in the experiment. For example E4 is the experiment done on the exhaustive model using holonym type relationships. The number 6 is used for the case of mixed use of types.

	Unstructured model	Structured Model	Exhaustive Model	Combined Model
Synonym	U1	S1	E1	C1
Hypernym	U2	S2	E2	C2
Hyponym	U3	S3	E3	C3
Holonym	U4	S4	E4	C4
Meronym	U5	S5	E5	C5
Combined Relationships	U6	S6	E6	C6

Table 7.1: Summary of the different experiments

Not all these experiments were carried out for reasons stated later. The experiments were performed on a Sun Sparc running Solaris 2.4 with the programming language Tcl 7.4 [Ous94] and C [HJ91] (via an interface to Tcl) to access the information in the WordNet thesaurus.

7.3 Results and analysis

The results of the different experiments are displayed and analyzed in this section. The results consist of precision and recall values displayed in tables or graphs. Unless otherwise specified, the values are shown in percentages. The tables from where the graphs are based are listed in the appendix of this chapter.

7.3.1 The benchmarks

The results of the benchmarks B1 (to be compared with the unstructured model) and the benchmark B2 (to be compared with the combined model) are shown in the following table:

		B1	B2
R E C A L L	10	2.23	46.33
	20	1.57	35.92
	30	1.42	26.72
	40	1.37	21.73
	50	1.26	19.47
	60	1.22	15.58
	70	1.19	12.47
	80	1.20	9.28
	90	1.13	7.12
	100	0.54	4.20
Average Precision		1.32	19.89

Table 7.2: Precision and recall values for the two benchmark models

The precision values obtained with the benchmark B2, i.e., the Vector Space Model, were as expected (see [vRRP80] for a comparison). The precision values obtained with the benchmark B1 were very low, which is not surprising since a document was relevant to a query if the document and the query had at least one common term. As explained in section 7.2.5, the benchmark B1 was used to identify for a query the documents that have at least one common term to that query. Statistics on the documents retrieved by the benchmark B1 are given in Table 7.3.

The table shows that the average number of non-retrieved relevant documents was much lower than that of the retrieved relevant documents. However, the average number of irrelevant retrieved documents was very high. These two observations imply that the overall recall is high, whereas the overall precision is very low[89]. One conclusion is that the use of WordNet relationships to extend a document's initial representation seems questionable since the overall recall obtained with the benchmark B1 is already high. That is, the information explicit in the documents usually allows retrieval of most relevant documents. However, the use of WordNet relationships, that is, the capturing of the information implicit in the documents, may help placing the relevant documents higher in the ranking. This is investigated in later sections.

[89] Here, the recall and the precision values are with respect to all the documents retrieved, i.e., with relevance greater than 0.0. The ranking is not taken into account.

Query number	Number of relevant documents	Number of relevant documents not retrieved	Number of retrieved documents not relevant	Query number	Number of relevant documents	Number of relevant documents not retrieved	Number of retrieved documents not relevant
1	19	0	2637	21	50	1	2465
2	15	1	2835	22	65	3	2301
3	33	0	3994	23	20	0	2623
4	5	0	1564	24	39	1	2529
5	4	3	1071	25	73	4	1769
6	10	1	566	26	53	3	1427
7	75	1	1944	27	28	0	974
8	1	0	3917	28	8	0	1687
9	2	0	2347	29	11	0	2525
10	11	0	3279	30	7	0	3003
11	4	0	2494	31	11	0	4479
12	39	0	4417	32	13	0	1147
13	59	3	2398	33	8	0	4855
14	56	1	3173	34	9	1	4413
15	32	6	2214	35	30	1	1571
16	26	0	4454	36	8	2	2644
17	23	0	3684	37	25	0	2452
18	13	0	2611	38	12	0	3195
19	26	1	2753	39	5	1	975
20	23	3	961	40	29	1	920

Average number of relevant documents per query: **24.5**
Average number of relevant documents retrieved per query: **23.55**
Average number of relevant documents not retrieved per query: **0.95**
Average number of irrelevant documents retrieved per query: **2531.67**

Table 7.3: Some statistics about the benchmark B1

7.3.2 The Unstructured Model

Five experiments were carried out, U1 to U5, using each type of WordNet relationships to compute a document's implicit information content. The results of these experiments are shown in the Table 7.4[90]. Only queries for which relevant documents were not retrieved with the benchmark B1 are shown.

The table shows that the unstructured model retrieves more relevant documents that the benchmark B1. Hence, the use of WordNet relationships allow retrieval of additional relevant documents, which could not be retrieved when only the explicit information of these documents was taken into account.

The table shows that hyponyms (U3) retrieve the highest number of relevant documents. The number of retrieved relevant documents decreases then with, respectively, synonyms (U1), hypernyms (U2), holonyms (U4) and meronyms (U5). However, as the last column shows, the relevant documents retrieved by the different types of relationships were often the same. This was due to several reasons. First, in WordNet, a term can be associated to the same term by different types of WordNet relationships. For example, the term "process" in WordNet has the term "act" as both a synonym, in sense 1, and a hypernym, although indirectly, in sense 2. Second, the WordNet relationships led to many general terms frequently used in queries. Examples of such terms include "method", "determination", and "expression". Finally, there was not all that many more relevant documents to be retrieved, so any difference that may rise with using various types

[90] The experiment U6 which uses several types of relationships was not performed since, at this stage, the aim was to ascertain the appropriateness of WordNet relationships.

of relationships was already limited.

Query number	Additional number of relevant documents retrieved					Number of additional documents retrieved for each query by all the relationships
	U1	U2	U3	U4	U5	
2	-	-	1	-	-	1
5	1	2	2	1	-	2
6	-	1	1	-	-	1
7	-	-	1	1	-	1
13	3	-	2	2	-	3
14	-	1	1	-	-	1
15	-	1	2	-	-	2
19	-	-	-	1	-	1
20	1	-	3	1	2	3
21	-	-	-	-	-	0
22	1	-	2	-	-	2
24	-	-	-	-	-	0
25	-	1	-	-	-	1
26	-	-	-	-	-	0
34	-	-	-	-	-	0
35	-	-	-	-	1	0
36	2	2	-	-	-	2
39	-	-	1	-	-	0
40	1	-	-	-	-	1
Number of additional documents retrieved for all queries	9	8	16	5	3	21
Average number of additional relevant documents retrieved per query	0.47	0.42	0.84	0.26	0.16	1.11
Average number of relevant documents not retrieved (this number was 2 for B1)	1.53	1.58	1.16	1.74	1.84	

Table 7.4: Comparison of the number of additional documents retrieved by the unstructured model

Statistics on the irrelevant documents retrieved by the unstructured model is given in the following table:

	U1	U2	U3	U4	U5
Average of irrelevant documents retrieved per query	3312.6	4867.3	4169.32	2981	2784.92
Increase with respect to B1	+780.92	+2335.62	+1637.65	+449.32	+253.25

Table 7.5: Irrelevant documents retrieved by the unstructured model

The highest number of retrieved irrelevant documents comes with the use of hypernyms (U2). This was because the use of hypernyms to extend a document's initial representation leads to much less specific representations. In these, terms such as "entity" or "class" were obtained, which can be used in the queries, although they do not best describe the information need expressed in a query.

Something not previously mentioned is that the use of hyponyms (U3), which retrieve the second highest number of irrelevant documents, presented computation problems because WordNet terms often possess a large number of hyponyms. For example, the term "human", for only one of its senses, had more than 100 hyponyms (synsets). As a result, the time required to compute the documents implicit information content using hyponyms was lengthy. To overcome this problem, the maximal depth initially fixed to 5 extensions was reduced to 2.

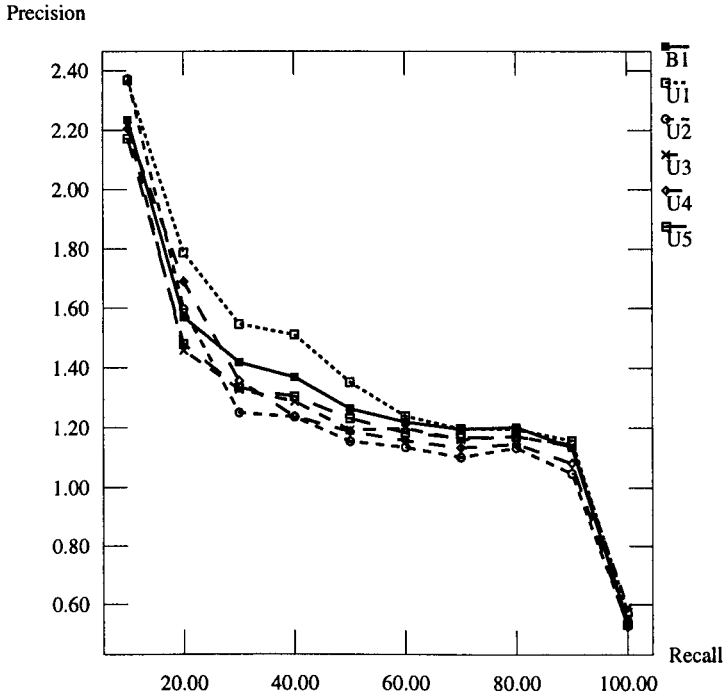


Figure 7.1: Precision and recall values obtained with the unstructured model

Obviously, with a depth of 5, many more irrelevant documents would have been retrieved. However, even with a depth of 2, the computation time took longer than when the other types of relationships were used.

The precision and recall values obtained with the unstructured model are shown in the Figure 7.1 (the benchmark B1 is also shown). The graph shows that the use of synonyms (U1) always improves precision, whereas the use of the other types of relationships often decreases precision with respect to the benchmark B1. In particular, the graph shows that the precision values obtained with hyponyms (U3) were much lower than those obtained with synonyms. Therefore, since the computation length required by the use of hyponyms was very high, hyponyms were not used subsequently to extend documents representations.

Further experiments could be performed on the unstructured model, for example to find out the maximal depth to impose on the number of extensions to obtain a better result (i.e., a trade-off between the number of relevant documents and non-relevant documents retrieved). This, however, goes beyond the scope of this thesis. Here, the aims of the experiments were to observe the behavior of the models proposed in this thesis.

To conclude, the experiments performed on the unstructured model led to positive results, since more relevant documents were retrieved by taking into account the implicit information in the documents. However, except for synonyms, the precision of the output deteriorated.

7.3.3 The Structured Model

The different types of WordNet relationships, except for hyponyms (see previous section), were used separately to structure (e.g., to build the basic situations of) the NPL documents. Some statistics

on the output of the process are shown in the following table (recall that the NPL collection has 11429 documents). In the table, a basic situation is said to be non-singleton if it contains more than one term.

	S1	S2	S4	S5
Number of documents with non-singleton basic situations	368	2190	548	240
Percentage of documents with non-singleton basic situations	3.2	19.16	4.79	2.09
Average length of the basic situations in those documents	1.04	1.33	1.025	1.186

Table 7.6: Results of structuring documents using the different WordNet types relationships

Except for hypernyms (S2), the number of structured documents with non-singleton basic situations was very low. One reason was that NPL documents were short (an average of 19.96 terms per document), so the chance that two terms in a document were semantically related by a WordNet relationship was low. The use of hypernyms to structure documents gave the highest number of documents with non-singleton basic situations because in WordNet, terms like "entity" or "place", which appear in a number of documents, are (directly or indirectly) hypernyms of a large number of terms.

Next, an example of a NPL document, and its structured representations using the different types of WordNet relationships is given. The original document was

apparent observation of solar corpuscular clouds by direct continuous wave reflexion a report of observations in ohio of doppler signals centred on mcs which were first recorded at ut on april the observations are discussed in relation to a solar flare of importance which reached a maximum at about ut on april

Figure 7.2: Example of a NPL document

Structuring this document with hypernyms led to the representation below. The basic situations are delimited with "{" and "}". The weight of the basic situation is first given, then the terms, their tags ("n" for noun and "p" for proper nouns[91]) and associated senses (between "(" and ")") are shown:

{0.095 april n (1) } {0.046 centred n (1) } {0.061 mc n (1) , relation n (1)} { 0.061 signal n (1), relation n (1) } {0.107 ut n (1), relation n (1) } {0.138 observation n (1 2 3 4 5) } {0.046 cloud n (1 2) } {0.046 wave n (1 2 3 4 5 6 7) } {0.046 ref lexion n (1 2 3 4 5 6) } {0.046 report n (1 2 3 4 5) } {0.046 ohio n (1 2) } {0.03 relation n (2 3) } {0.046 flare n (1 2 3) } {0.046 importance n (1 2) } {0.046 maximum n (1 2 3) } 0.046 co rpuscular p () } {0.046 doppler p ()}

Figure 7.3: Structured representation of a NPL document using hypernyms

The basic situation { 0.061 signal n (1), relation n (1) } was built because in WordNet, "relation" is a hypernym, although indirectly, of "signal". The structured representation of the document using the other types of WordNet relationships had no singleton basic situations:

[91] Proper nouns usually refer to name of country, people, city, river. Here, a proper noun is any term that is unknown from WordNet (see Chapter 6). Tags are used to differentiate between nouns and proper nouns, since proper nouns will never be part of a relationship between terms.

{0.095 april n (1) } {0.047 centred n (1) } {0.047 mc n (1) } {0.047 signal n (1) } {0.095 ut n (1) } {0.142 observation n (1 2 3 4 5) } {0.047 cloud n (1 2) } {0.047 wave n (1 2 3 4 5 6 7) } {0.047 reflexion n (1 2 3 4 5 6) } {0.047 report n (1 2 3 4 5) } {0.047 ohio n (1 2) } {0.047 relation n (1 2 3) } {0.047 flare n (1 2 3) } {0.047 importance n (1 2) } {0.047 maximum n (1 2 3) } {0.047 corpuscular p () } {0.047 doppler p ()}

Figure 7.4: Structured representation of a NPL document using synonyms, holonyms or meronyms

The few occurrences of non-singleton basic situations was also due to the fact that the WordNet relationships were not specific to the NPL collection. Many documents terms had no WordNet entry (these terms are tagged with "p" in the above examples). For terms with WordNet entries, often none of the senses in those entries were appurtenant. Consider the following NPL document:

the coaxial system amplifiers

Figure 7.5: Example of a NPL document

The synonyms and hypernyms of the terms "system" and "amplifier" are given Figures 7.6 and 7.7.

Sense 1
synonym: system, unit
hypernym: instrumentality, instrumentation
Sense 2
synonym: system
hypernym: substance, matter
Sense 3
synonym: system
hypernym: group, grouping
Sense 4
synonym: arrangement, organization, system
hypernym: structure
Sense 5
synonym: system, system of rule
hypernym: method
Sence 6
synonym: system
hypernym: body part
Sense 7
synonym: system
hypernym: plan of action
Sense 8
synonym: system
hypernym: live body

Figure 7.6: WordNet entries of the term "system"

The most appropriate senses of "system" for the above document are 4 and 5. In many of its senses, the term "system" has no synonym.

Sense 1
synonym: amplifier
hypernym: electronic equipment

Figure 7.7: WordNet entries of the term "amplifier"

The term 'amplifier" has no synonym. The above two entries illustrate that WordNet relationships was ineffective in structuring documents (and hence, in computing the implicit information content of the documents) in the NPL collection because many terms that could be viewed as related were not assessed so by WordNet. Therefore, it was not possible to construct the basic situations as being semantic structures. This was a problem because the computation of the relevance of a document to a query, as defined by the structured model, depended strongly on that semantically related information items were grouped into basic situations (semantic structures).

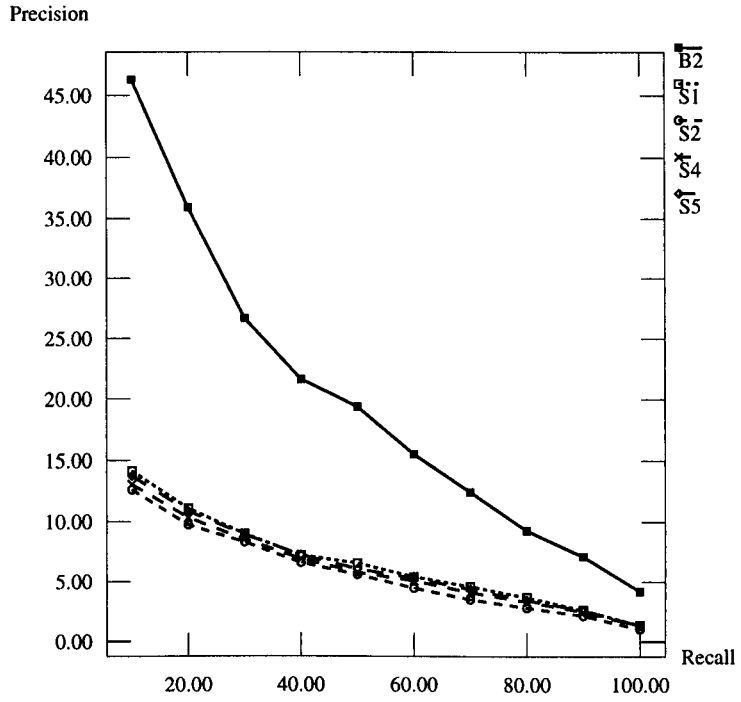


Figure 7.8: Precision and recall values obtained with the structured model

Relationships extracted from a thesaurus or a knowledge base specific to the NPL documents would have certainly led to more accurate structured representations of the NPL documents. Unfortunately, such a thesaurus or knowledge base was not available. This is a main drawback and will certainly lead to impoverished performance of the structured model. This should be taken into account when further investigating the results obtained with the structured model.

The relevance of documents to queries as defined by the structured model were then computed. The precision and recall values obtained are shown in Figure 7.8 (the benchmark B2 is also shown in the graph). The graph shows that the structured model performed best with synonyms (S1), then with meronyms (S5), holonyms (S4), and finally hypernyms (S2). However, the difference in the overall precision was very small, and could mainly be explained by the fact that no significant difference in the relevant documents retrieved by the various types of relationships was observed (see Table 7.4).

The fact that the best results (although not significantly better) were obtained with synonyms is not surprising because extending a document's representation to one containing synonymous terms seems an intuitive approach. The worst results were obtained with hypernyms because they often yield document descriptions that contain general terms frequently used in queries. The results obtained with holonyms and meronyms can be explained by the fact that less relevant documents were retrieved than with synonyms, but less irrelevant documents were retrieved than with hypernyms.

The precision and recall values obtained with the experiments performed with the structured model were much lower than that obtained with the vector space model (the benchmark B2). However, it should be stated that the structured model captured only the specificity of documents to queries, whereas the Vector Space Model captured both specificity and exhaustivity. It is then difficult (yet) to speculate on the results because the relevance assessment provided by the NPL collection

takes into account both exhaustivity and specificity of document to queries. That is, no assessment on either the documents exhaustive or specific to queries is available. To impetrate on the extent to which the structured model indicates a specificity measure, the four most relevant documents established by S1 (synonyms) for query 13 are examined:

Document	Relevance Degree
4079	0.75
4354	0.666666
4626	0.666666
2085	0.500001

Table 7.7: The four most relevant documents as established by S1 for query 13

The query 13 and some of the above documents are shown below:

Query 13
mathematical expressions and graphs for the design of transistorised tuned pass band amplifiers

Document 4079
design of unsymmetrical band pass filters

Document 4354
etched wiring simplifies magnetic amplifier design

Document 4626
magnetic amplifier design a practical approach

Figure 7.9: Query 13 and document numbers 4079, 4354 and 4626

Document 4079 was identified as most specific to the query. Looking closely at this document, although the terms "mathematical expressions" or "graph" were not mentioned in the document, the document can be viewed as specific to the query, although it is not exhaustive to the query.

Documents 4626 and 4354 were also determined as highly specific to the query, although this is less evident. For example, the document 4626 mentions "practical approach" whereas query 13 seems to refer to "theoretical approach". This happened because in both the design and the implementation of the models, terms were treated independently. This obviously should be refined as discussed in Chapters 4, 5 and 6. For example, if the representation of a document is extended to one that contains a term that "contradicts" one of the terms used in the query, the relevance of the document should be reduced.

Document 4354 has been qualified as specific to the query because the document is short and contains two terms that appear in the query. This was more obvious by looking at the following document which was assessed relevant to the query 13:

band pass amplifiers their synthesis and gain bandwidth factor seven types of band pass amplifier are investigated and compared and their design formulae are given

Figure 7.10: Document number 2458

The rank of document 2458 is 19 which shows it was determined as specific to the query. The document was, however, lower in the ranking than for all the documents listed in Table 7.7 mainly because it was a longer document, and as earlier explained, the semantic-based structured representation of documents, as obtained in this experiment, is poorly captured; in many cases, a basic situation was constituted of a single term. Moreover, some of these terms were not as informative as others in expressing a document's information content. This should then be reflected in the weight assigned to their corresponding basic situations so that, not only the weight expresses the significance of a basic situation in a document, but also its "informativeness". The weighting mechanism adopted here did not incorporate this feature.

Another reason for obtaining low precision values was that irrelevant documents were incorrectly retrieved due to the WordNet relationships. One reason for this was that no disambiguation was done on both queries and documents terms, and many terms were erroneously obtained in the extended representations of documents.

To conclude, the structured model offers a measure of the specificity of a document to a query. It, however, presented some problems. First, the basic situations constructed did not constitute adequate semantic-based structured representations of documents. Second, the weighting mechanism did not distinguish informative to non-informative terms, and third, too many irrelevant documents were retrieved by the use of WordNet relationships. The first problem, unfortunately, cannot be solved, unless other data are used, but solutions to the latter are investigated in section 7.4.

7.3.4 The Exhaustive Model

The experiments performed to evaluate the exhaustive model were not successful, because for nearly all queries, no documents were retrieved. That is, no extension of a document's representation led to a representation that contained all the terms in the query. This shows that the measure of exhaustivity defined in Chapter 5 is too strict, at least for the queries provided by the NPL collection. This measure may be more appropriate for shorter queries of 2 to 3 terms such as "operating systems", where the retrieved documents should be about "operating systems" and not "operating" or "systems" alone. In section 7.4.4, a less strict measure of exhaustivity of a document to a query was proposed.

7.3.5 The Combined Model

Since no documents (or very few) were retrieved by the exhaustive model, the combined model became

$$\frac{Spe(d,q)}{2}$$

where $Spe(d,q)$ was computed by the structured model. This meant that the ranking of documents was the same as for the structured model. As a result, the evaluation of the combined model yielded the same precision and recall values obtained with the structured model. Compared to the vector space model, the combined model performed poorly, the reason being that no information on the extent to which a document was exhaustive to a query was rendered. Based on a new formulation of the exhaustive model, the combined model was again experimented with in section 7.4.5.

7.4 Additional experiments, their set up, results and analysis

New experiments were performed to address some of the issues raised in the previous sections. These issues were:

(i) the differences between the precision values obtained with the various types of WordNet relationships were insignificant. This may change if several types of WordNet relationships are used together.

(ii) often terms that have nothing to do with the initial representation of a document were obtained with the use of WordNet relationships. Disambiguating terms may remedy this problem.

(iii) the initial weights of the basic situations did not allow the distinction between informative and non-informative information. A better weighting mechanism is required.

(iv) the measure of exhaustivity proposed in Chapter 5 was too strict. A new measure must be defined.

(v) due to (iv), the combined model collapsed into the structured model, which computes a measure of specificity. With a less strict exhaustivity measure, the combined model may lead to better precision values.

To investigate the above issues, new experiments were performed with 12 queries arbitrary selected in each of the following groups:

Group A : where all documents were retrieved without using WordNet relationships: 1, 8, 10 and 30.

Group B : where additional documents were retrieved using WordNet relationships: 5, 13, 19 and 37.

Group C : where no additional documents were retrieved using WordNet relationships: 7, 15, 24 and 39.

These groups reflect the three possible scenarios that arise with the use of WordNet relationships to compute a document's implicit information content. All the new experiments, with one exception, used one type of WordNet relationship, namely synonyms, since their use provided the best results (the highest precision values).

Before describing the new experiments, the precision and recall values were computed for the above 12 selected queries for the benchmark B2 and the structured model using synonyms (the experiment S1). These new values constituted the new benchmarks, and were referred to as B3 and Syn, respectively. These two new benchmarks were compared, respectively, to B2 and S1 to establish whether the structured model or/and the vector space model favor the selected 12 queries. The two comparisons are shown in Figure 7.11.

In overall, both Syn and B3 performed better than S1 and B2, respectively. However, the extent to which B3 was better than B2 was higher to that of Syn with respect to S1, thus showing that the selected 12 queries did not advantage the structured model with respect to the vector space model.

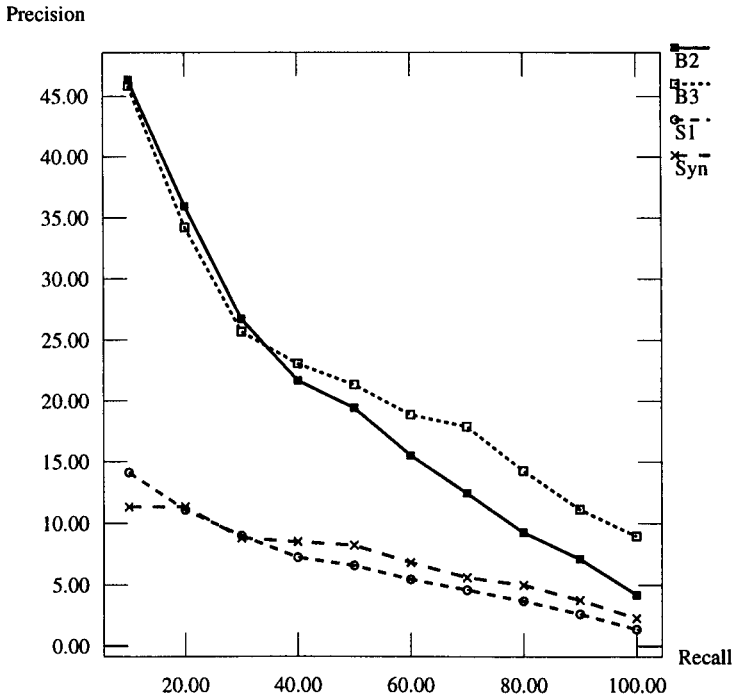


Figure 7.11: Comparison of the benchmarks for 12 and 40 queries

7.4.1 Use of synonyms and holonyms (Syn1)

The aim of the experiment, called *Syn1*, was to determine whether using several types of relationships in the structured model improved precision. Two types of WordNet relationships were used: synonyms and holonyms. Synonyms were used because they led to the best precision values. Holonyms were used because they allow retrieval of more relevant documents than meronyms, and less irrelevant documents than hypernyms (see Tables 7.4 and 7.5). The precision and recall values of the experiment Syn1 are shown in the graph in Figure 7.12.

No improvement on the precision values was obtained with using synonyms and holonym relationships in the structured model. This result was not surprising, since, as explained in section 7.3.3, no significant difference was obtained with the various relationships (see the precision and recall values in the Figure 7.8). Therefore, the decision to conduct all the new experiments only with synonyms seemed right.

Other combinations may have led to higher precision values. A particular combination would be with synonyms and hyponyms since as shown in section 7.3.2, the latter relationships retrieved the highest number of relevant documents. However, due to the problem explained in 7.3.2, this combination was not attempted.

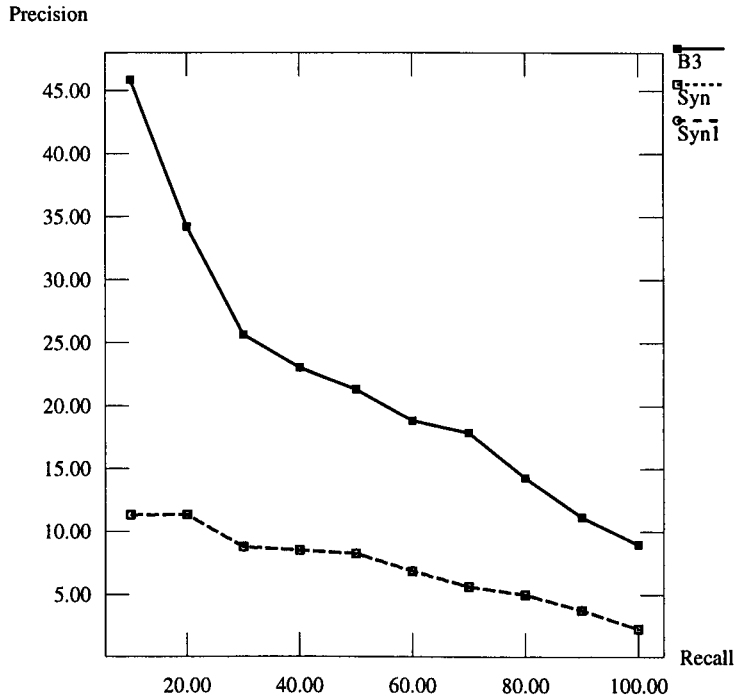


Figure 7.12: Precision and recall values obtained with the experiment Syn1

7.4.2 Limited number of term senses (Syn2)

As explained in section 7.3.3, the extended representation of a document could include terms, the meaning of which was unrelated to the meaning of the terms contained in the initial representation of that document. The main reason being that no disambiguation was performed on documents or queries; many senses of a term were alien to those referred to in a document. This experiment, referred to as *Syn2*, attempted to remedy this problem by restricting the senses of terms in a document's representation. The restriction was based on the fact that, in WordNet, the senses of a term are displayed in decreasing order of their use. This can be seen in the following example:

Sense 1	**Sense 4**
horse, Equus caballus	sawhorse, horse, sawbuck, buck
Sense 2	**Sense 5**
horse	knight, horse
Sense 3	
cavalry, horse cavalry, horse	

Figure 7.13: Synonyms of the term "horse" in WordNet displayed in decreasing order of their use

In the experiment Syn2, only the first two senses of a term were taken into account. That is, the computation of the relevance degree was as for the structured model with the difference that if a term had more that two senses, only the first two were used. This number was arbitrary chosen. Obviously, the first two senses of a term may not be those referred to in a document. The precision and recall values obtained with the experiment Syn2 are shown in the following graph:

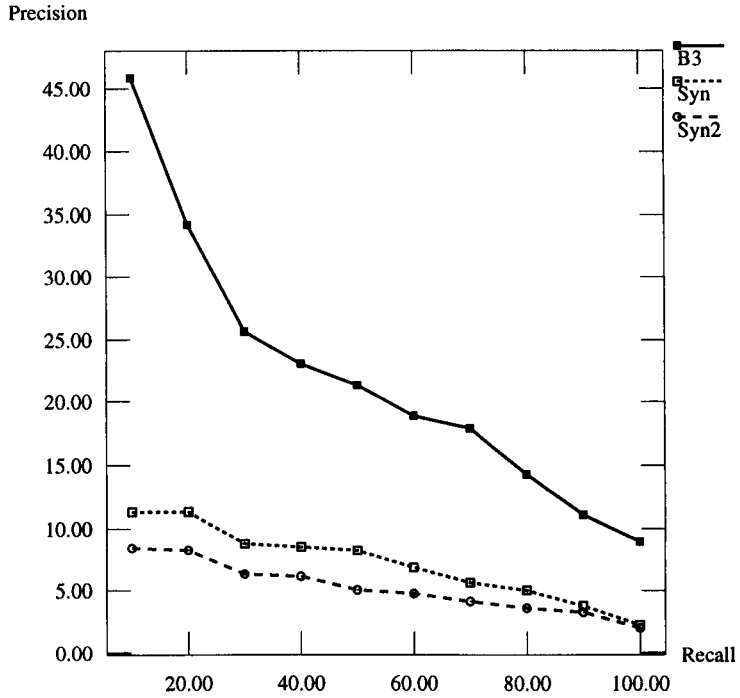


Figure 7.14: The precision and recall values obtained with the experiment Syn2

A decrease in the precision values was observed with Syn2. This was owed to the fact that the two most used senses of a term according to WordNet were often not those referred to in the documents of the NPL collection. For example, the term "pass" that appears in the document 4079 (see Figure 7.9) had the following WordNet synonyms (part of the entry is displayed):

Sense 1	**Sense 5**
base on balls, walk, pass	pass, laissez passer
Sense 2	**Sense 6**
pass	pass, strait, straits
Sense 3	**Sense 7**
pass, passing play, passing game, passing	pass, mountain pass, notch
Sense 4	
pass, passport	

Figure 7.15: WordNet synonym entries of the term "pass"

The order provided by WordNet was not suited to the NPL collection. To restrict the senses associated to terms, proper disambiguating is necessary.

7.4.3 A different weighting mechanism for the basic situations (Syn3)

The basic situations were usually constituted of single terms, some of which were more informative than others in describing the document's information content. The weighting mechanism, as implemented in Chapter 6, did not permit distinction between informative and non-informative terms in the basic situations.

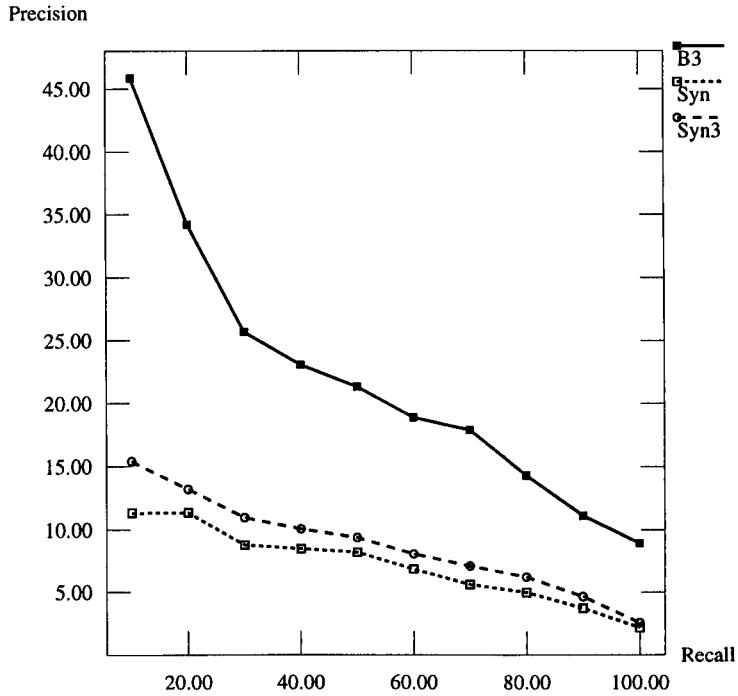


Figure 7.16: Precision and recall values obtained with the experiment Syn3

Non-informative terms are usually those that appear in many documents of a collection, and can be differentiated from the informative ones by associating to the terms weights that include the so-called inverse document frequency (this was used in the vector space model, the benchmark B2, described in section 7.2.5).

The same strategy is adopted in this experiment, called *Syn3*, to distinguish informative terms to those less informative in a basic situation. The weighting mechanism was redefined to include inverse document frequency, so that higher weights were assigned to those basic situations that contained more informative terms than others. For simplicity, since the basic situations were mainly singletons (see Table 7.6), for each term t in the document, a basic situation s was created. The weight (BPA) of the situation s was given by

$$m(s) = \frac{freq(t, D) * idf(t)}{\sum_{s \in D} m(s)}$$

D was the weighted information domain modelling the document (see Chapter 5), that is, the set of basic situations, one for each term explicitly extracted from the document. The denominator was the summation of weights of all the basic situations (terms) in the document. $freq(t, D)$ and $idf(t)$ were the term occurrence frequency in the document (as defined in Chapter 6) and the inverse document frequency (as computed by the vector space model), respectively. The implementation of the other components of the model is as described in Chapter 6.

The precision and recall values obtained with the experiment Syn3 are shown in Figure 7.16. The graph shows a significant increase in the precision values with respect to Syn. This indicates that including inverse document frequency in the weights attached to basic situations was effective. It also happened that the document 2458 ranked $19th$ with Syn (see section 7.3.3) was now ranked $7th$ and that the document 4354 (see Figure 7.9) ended up much lower in the ranking.

The main conclusion of the experiment Syn3 is that precision can be ameliorated with a more accurate weighting mechanism. In some further work, additional experiments will be carried out using various weighting mechanisms that were showed to capture well the informativeness of terms in documents (for example, see [SM80] for various formulations of the terms weights in the vector space model), although the improvement may be limited since the basic situations, as implemented here, did not constitute proper semantic structures as defined in Chapter 5.

7.4.4 New measure of exhaustivity (Syn4)

The measure of exhaustivity, proposed in Chapter 5, showed to be too strict. To compute the relevance of a document to a query that includes both exhaustivity and specificity, a new measure of exhaustivity was defined in this experiment, called *Syn4*. A document and a query were represented by a set of terms d and q, respectively. The number of terms common to the document and the query was computed and assigned to a variable E. For all the other terms in the document that were not contained in the query, and that had a WordNet entry, the following procedure was applied. Let n be the number of senses of that term in the entry. For each sense, the following was done:

(i) if m terms in that sense appeared in the query, that were not matched before, the value m/n was added to the value of E.

(ii) otherwise, the same process was applied for the terms in that entry[92]. The result was assigned to a variable E' that was initially set to 0 (since no terms in that sense appeared in the query). The value E'/n was then added to the value of E

The result of this procedure was a numerical value (E) that represented the number of common terms between the query and the document, either explicit, or implicit (via the use of WordNet relationships). The uncertainty of the relationships (see Chapter 6) was taken into account. For example, suppose that the term "horse" appeared in the document, and that the term "cavalry" appeared in the query. In WordNet, "cavalry" is a synonym of "horse" for one of its sense. Since horse has 5 senses, then the uncertainty associated to the relationships was $1/5$, which was added to the variable E.

To capture the extent to which the query is covered by the document, the above value E is divided by the number of terms in the query. With this formulation, the more terms in the query that are explicitly or implicitly contained in the document, the higher will be the result of the division. Hence, this formulation expressed a measure of exhaustivity. The precision and recall values obtained with the experiment Syn4 are shown in Figure 7.17.

The exhaustive model gave better precision values than the structure model, and lower precision values than the vector space model. However, it should be reminded that the three models base their relevance on different criteria.

[92] The terms appearing in the same sense of a term constitute a synset. In WordNet, all the terms in a synset have the same synonyms. So in practice, it was sufficient to deal with only one term in the synset.

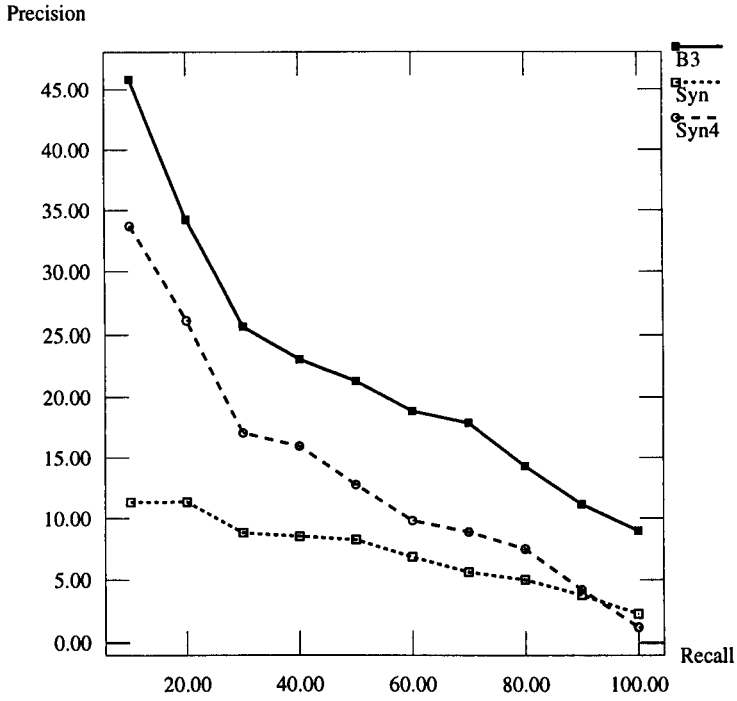


Figure 7.17: Precision and recall values obtained with the experiment Syn4

As for the evaluation of the specificity measure, no relevance assessment on the exhaustivity of documents to queries are known. Therefore, to have an idea on how well the exhaustive model performed, the top most ranked documents retrieved for query 13 by the exhaustive model are examined:

Documents	Relevance
8136	0.714286
2458	0.6
5873	0.6
7908	0.6

Table 7.8: The four top most ranked documents as established by the new exhaustive model for query 13

These documents are showed in Figure 7.18. These documents seem highly exhaustive to the query 13 (shown in Figure 7.9), thus showing that the formulation in Syn4 indeed expressed exhaustivity.

Better formulations could be searched for. For example, with Syn4, longer documents seemed to have higher chance to be established exhaustive to a query than shorter documents. This could be remedy by having a document representation similar to that used in the structured model. That is, the information in the document is organized into semantic structures (e.g., concepts). The computation of the relevance will however be different. No other formulation was experimented with because as shown in the structured model, to obtain positive results, better representations of documents that what were obtained are mandatory. As it was already observed through this chapter, this would be difficult, if at all possible, because the WordNet relationships are not specific to the NPL collection.

Document 8136
design of hf and if amplifiers for multi channel fm links bandwidth required is determined from the side band amplitudes for given modulation index and from the permissible distortion expressions for involve the derivatives of the amplifier response curves and are tabulated for single stage circuits and two stage band pass filters the computation for multistage circuits is shown to involve the same derivatives under ideal conditions second harmonic distortion would be eliminated by exact tuning to the central frequency and third harmonic distortion sufficiently reduced by using symmetrical filters with a coupling coefficient in practice tuning is not exact but second harmonic distortion can be kept at a tolerable level by adjustment of the pass band response at the alignment stage valve capacitance variations are allowed for in the shunt circuit capacitance neutralization can practically eliminate the effects of feedback via grid anode capacitance criteria for the choice of if are explained

Document 5873
design of wide band tuned amplifiers amplifiers with schienemann butterworth and tchebycheff band pass characteristics are considered in detail and formulae design curves and numerical examples are given

Document 2458
band pass amplifiers their synthesis and gain bandwidth factor seven types of band pass amplifier are investigated and compared and their design formulae are given

Document 7908
amplifier stages with transitionally coupled two stage band pass filters particularly for large bandwidths the amplitude and group delay characteristics of an amplifier stage consisting of two coupled circuits are analysed for the case when the amplification is constant over a wide frequency band transitional coupling the case when the damping factors d and d of the two circuits are equal is considered first and formulae are also given for the cases of either d or d tending to zero formulae are also given for transforming a filter with indirect inductive coupling into one with direct inductive coupling design curves are shown

Figure 7.18: The NPL documents 8136, 2458, 5873 and 7908

7.4.5 The Combined Model (Syn5 and Syn6)

Two experiments were performed to evaluate the combined model. They both used the measure of exhaustivity defined in the previous section, but used two different measures of specificity: the one initially defined (Syn) and the improved one (Syn3) experimented with in section 7.4.3. The two experiments are referred to as *Syn5* and *Syn6*, respectively. The precision and recall values obtained with these two combinations are shown in the graph in Figure 7.19.

With both combinations, an increase in the precision values was observed with respect to the specificity and the exhaustive models. Therefore, the combination of a measure of specificity and a measure of exhaustivity, results into a measure of relevance taking into account both specificity and exhaustivity.

A higher increase was obtained with the second combination (syn6). This is because the formulation expressing the specificity of a document to a query used in Syn3 was better than that used in Syn.

However, the results are still lower than those obtained with the benchmark B3 (the vector space model). This is due to that, as already discovered throughout this chapter, the WordNet thesaurus was not the best knowledge base to use with the NPL collection. As a result, the structured model behaved poorly, and hence the results obtained by the combined model were lower than those obtained with the vector space model. To show the problem encountered with the WordNet relationships, a final experiment was performed. It is described next.

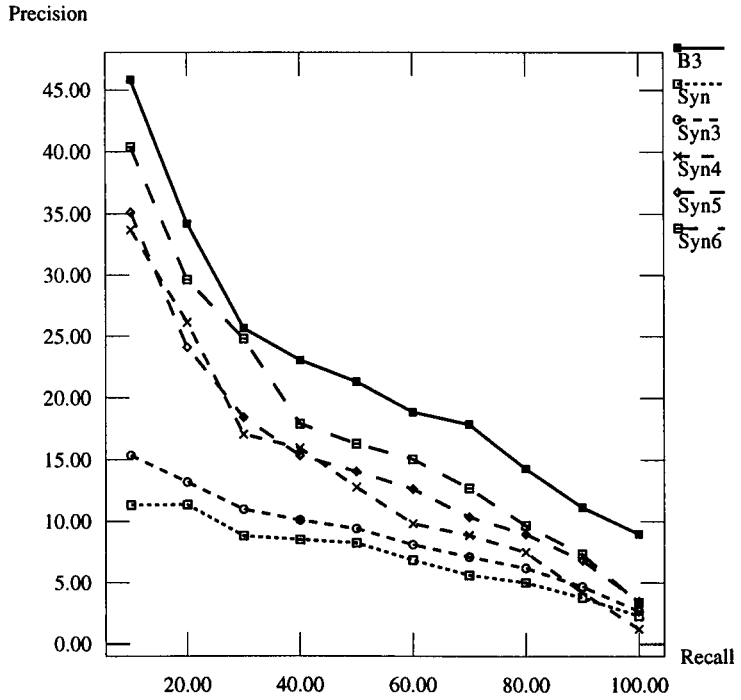


Figure 7.19: Precision and recall values obtained with the experiments Syn5 and Syn6

7.4.6 Query terms weights (Syn7)

This experiment, called *Syn7*, was carried out to establish precisely how appropriate were the WordNet relationships to compute a document's implicit information content. The adopted approach used the vector space model but incorporated the WordNet relationships in the computation of the relevance of a document to a query. A document and a query were represented as two vectors $\langle w_{1,d}, \ldots, w_{N,d} \rangle$ and $\langle w_{1,q}, \ldots, w_{N,q} \rangle$ which were defined as described in section 7.2.5. The relevance was computed as follows:

$$R(d,q) = \frac{\sum_{i=1}^{N} w_{i,q} * w_{i,d} * \Delta(i)}{\sqrt{\sum_{i=1}^{N} (w_{i,d})^2} * \sqrt{\sum_{i=1}^{N} (w_{i,q})^2}}$$

$\Delta(i)$ is 1 if the term t_i is in the query. Otherwise, the WordNet entry of that term was looked at. Suppose that the term t_i had n senses. A variable A was set to 0. For each sense, the following was done:

(i) if one term or several terms in that sense appears in the query, the value $1/n$ was added to A.

(ii) otherwise, the same process was applied for that entry. The resulting value was assigned to A'. Then the value A'/n was added to A

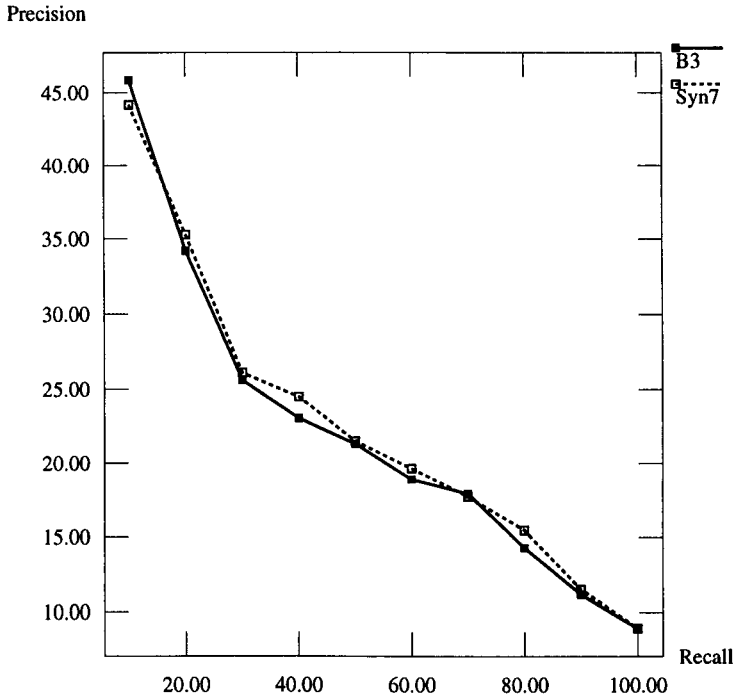


Figure 7.20: Precision and recall values obtained with the experiment Syn7

The result of this procedure was $\triangle(i) = A$, the idea being that a term t_i of a document may be related via WordNet to some query terms. Since no disambiguation was done, the relationships could be uncertain. The more uncertainty was introduced, the lower was the value of $\triangle(i)$.

The precision and recall values obtained with such a formulation of relevance are shown in the Figure 7.20. The results show an overall increase in the precision values. This increase is, however, very small, and may indicate that it would have been difficult to obtain good results with the combined model, in particular, because of the problems arising with the structured model. This is due to the fact that the WordNet relationships could not adequately determine a document's implicit information content. Moreover, the WordNet relationships are much too general. This was already made obvious when the results obtained with the structured model, using different types of relationships, happened to be very similar (see Figure 7.8).

If semantic relationships can only be extracted from WordNet (or any unspecific knowledge base), the transformation process must be more carefully applied to documents. To achieve this, techniques similar to those used in expert systems may be used, for example, using heuristics specific to the document collection. This task requires appropriate expertise, and hence, will be the purpose of future research.

7.5 Conclusion and Discussion

Different experiments were carried out to evaluate the unstructured, structured, exhaustive and combined models. These were done in two phases. First, the experiments were based on the implementations of these models as described in Chapter 6. The following summarizes the results of the various experiments in the first phase:

(i) **The Unstructured Model** The aim of this experiment was to determine whether more relevant documents were retrieved with the use of the WordNet relationships than without their use. The results showed that this was the case, although, there were not all that many more relevant documents to be retrieved. The results also showed a high increase in the number of irrelevant documents retrieved, and that the best results in terms of precision and recall values were obtained with synonyms.

(ii) **The Structured Model** The results obtained were poor, but this was due to the fact that the relevance degree, as computed by the structured model, expressed a measure of specificity, which depended strongly on the fact that the information into the document was organized into semantic structures (the basic situations). WordNet did not allow an adequate semantic-based structured representations of the NPL documents.

(iii) **The Exhaustive Model** The measure of exhaustivity was too strict.

(iv) **The Combined Model** Due to the above, the combined model collapsed into the structured model, thus giving poor result when compared to the vector space model.

The conclusion from the first phase is that taking into account the implicit information as well as the explicit information contained in documents did allow retrieval of more relevant documents than when only the explicit information was considered. However, it was essential to correctly determine this implicit information. In particular, this was mandatory when the basic situations constituting a document were constructed. The WordNet relationships with respect to the NPL collections proved inappropriate for the task, hence poor results were obtained.

Some issues were raised during the first phase. From them, further experiments were performed on a smaller number of queries, and using only synonyms. These additional experiments are summarized below:

(i) **Syn1:** Several types of relationships were used in the structured model.

(ii) **Syn2:** Only the first two senses of terms were taken into account in the structured model.

(iii) **Syn3:** A different mechanism was used in the structured model.

(iv) **Syn4:** A new formulation of the exhaustivity measure was proposed.

(v) **Syn5:** The combined model was experiment with the measure of specificity obtained in Syn and the new measure of exhaustivity.

(vi) **Syn6:** The same as above, but the measure of specificity was that obtained with Syn3.

(vii) **Syn7:** This experiment was essentially the vector space model with incorporation of WordNet relationships.

The precision and recall values of the experiments carried in the second phase (except for Syn1) are all shown on the following graph:

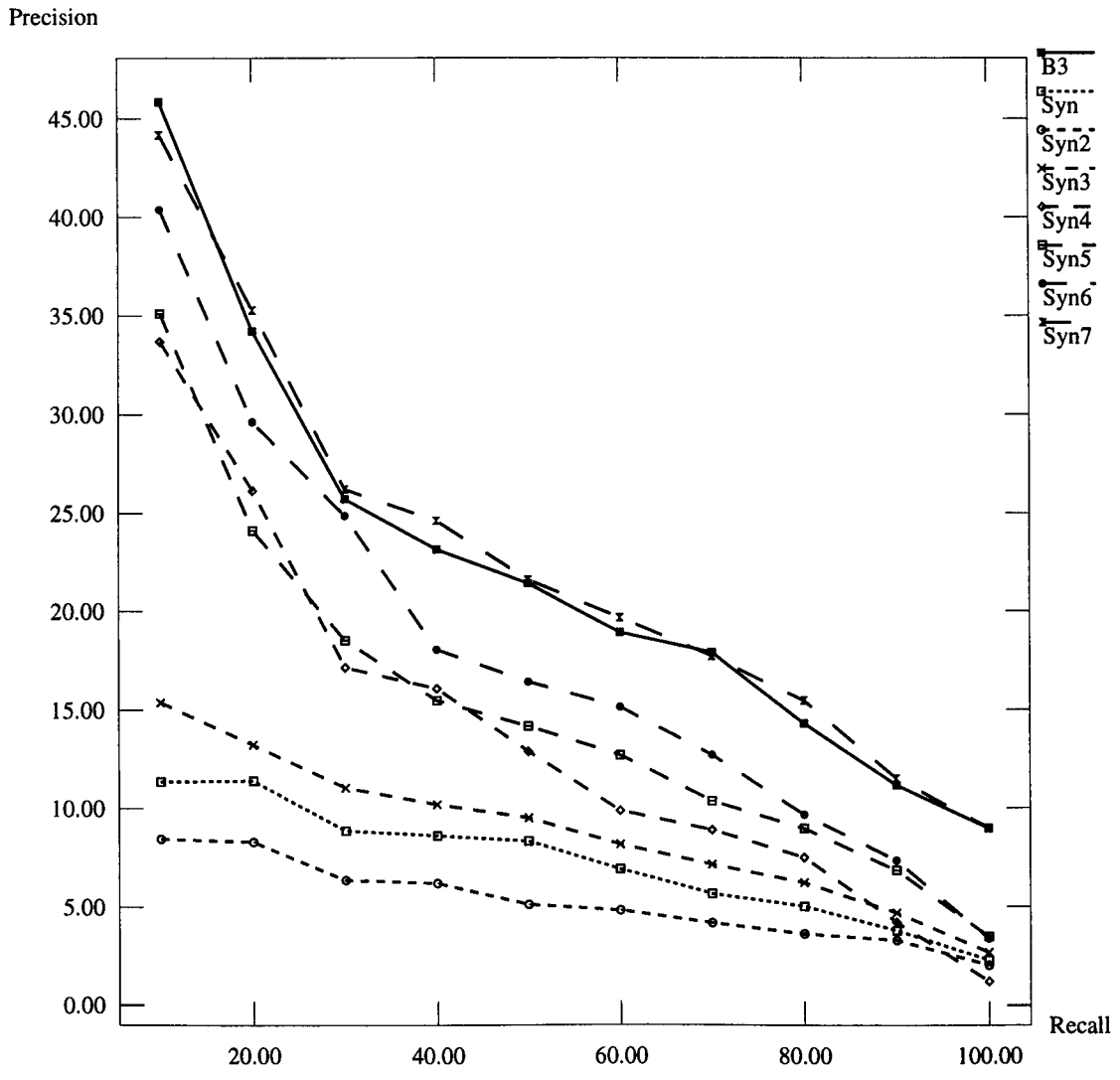


Figure 7.21: The precision and recall values obtained with the experiments Syn, Syn2, Syn3, Syn4, Syn5, Syn6 and Syn7

The results obtained with the different experiments were disappointing. However, after looking more closely at the documents, the queries, and more importantly, the WordNet thesaurus, obtaining positive results would have been difficult. For example, the experiment Syn7 showed that very little improvement of the precision values was obtained with the use of the WordNet relationships in the formulation of the vector space model.

The implementations of the models proposed in this thesis depended strongly on the availability of a knowledge base or a thesaurus implementing the constraints and appropriate to the NPL documents. Since such a knowledge base or thesaurus did not exist at the time of the experiments carried out in this thesis, obviously low performances were somewhat expected. Nevertheless, the experiment Syn6 showed that, even with a bad capturing of the flow of information (the constraints defined the nature of the flow of information so important in an IR system), the results although not as good as those obtained with the vector space model were still acceptable, in the sense that they could indicate that if a better knowledge base or thesaurus was available, better performances

will certainly obtain.

7.6 Appendix

This appendix contains the tables showing the precision values for standard recall values obtained with the different experiments. The tables also show increase or decrease with respect to benchmarks or other experiments. All values are shown in percentage.

		U1	U2	U3	U4	U5
	10	2.36	2.37	2.21	2.20	2.17
	20	1.78	1.59	1.45	1.69	1.48
R	30	1.54	1.25	1.33	1.35	1.33
E	40	1.51	1.24	1.29	1.24	1.30
C	50	1.35	1.15	1.19	1.19	1.23
A	60	1.24	1.13	1.20	1.15	1.18
L	70	1.19	1.10	1.16	1.13	1.16
L	80	1.19	1.13	1.17	1.14	1.17
	80	1.15	1.04	1.13	1.08	1.14
	100	0.57	0.52	0.58	0.52	0.53
	Average	1.39	1.26	1.27	1.27	1.27
	Increase B1	+0.08	-0.06	-0.04	-0.04	-0.04

Table 7.9: Precision and recall values for the unstructured model

		S1	S2	S4	S5
	10	14.13	12.60	13.04	13.72
	20	11.10	9.73	10.33	10.85
R	30	9.02	8.36	8.59	8.97
E	40	7.29	6.65	6.97	7.30
C	50	6.62	5.66	5.86	6.15
A	60	5.48	4.52	5.12	5.34
L	70	4.62	3.55	4.14	4.39
L	80	3.71	2.83	3.34	3.47
	80	2.64	2.16	2.51	2.61
	100	1.37	1.03	1.31	1.34
	Average	6.60	5.71	6.13	6.42
	Increase B2	-13.29	-14.17	-13.76	-13.47

Table 7.10: Precision and recall values for the structured model

		Syn	B3
R E C A L L	10	11.33	45.84
	20	11.36	34.20
	30	8.83	25.68
	40	8.55	23.11
	50	8.27	21.38
	60	6.87	18.90
	70	5.63	17.90
	80	5.00	14.29
	90	3.77	11.14
	100	2.28	8.96
	Average	7.19	22.14
	Increase B2		+2.26
	Increase S1	+0.59	

Table 7.11: Comparison of the benchmarks with 12 vs. 40 queries

		Syn1	Syn2	Syn3	Syn4	Syn5	Syn6	Syn7
R E C A L L	10	11.30	8.43	15.37	33.68	35.11	40.38	44.18
	20	11.35	8.27	13.20	26.12	24.10	29.61	35.27
	30	8.78	6.32	11.00	17.11	18.47	24.82	26.17
	40	8.55	6.14	10.14	16.03	15.42	17.99	24.57
	50	8.30	5.06	9.45	12.83	14.11	16.37	21.56
	60	6.92	4.77	8.12	9.83	12.65	15.11	19.66
	70	5.64	4.15	7.12	8.88	10.35	12.69	17.71
	80	5.02	3.60	6.20	7.48	8.95	9.65	15.46
	90	3.77	3.27	4.67	4.19	6.82	7.32	11.48
	100	2.26	2.02	2.66	1.21	3.46	3.36	8.98
	Average	7.19	5.21	8.80	13.74	14.95	17.74	22.51
	Increase B3	-14.95	-16.93	-13.35	-8.40	-7.20	-4.41	+0.37
	Increase Syn	0.00	-1.99	+1.60	+6.55	+7.75	+10.54	+15.31

Table 7.12: Precision and recall values for the experiments Syn1, Syn2, Syn3, Syn4, Syn5, Syn6 and Syn7

Chapter 8
Conclusions and Future Work

8.1 Introduction

This chapter summarizes the research performed in this thesis and the results achieved. It shows the main contributions of this work, and it discusses some of its limitations, and how they might be overcome. It also suggests future directions of research.

8.2 Summary of research carried out

Current IR models only offer simplistic and specific representations of information. There is therefore a need for the development of a new formalism able to model IR systems in a more generic manner. Van Rijsbergen [vR86a, vR86b, vR89] suggested that such formalisms can both be appropriately and powerfully defined within a logic. The resulting formalism should be able to capture information as it appears in an IR system, and also in any of its inherent forms.

8.2.1 Logic-based Information Retrieval models

In a logic-based model of an IR system, the information content of a document is represented by a sentence d, and the information need, as phrased in the query, is represented by a sentence q. The truth of $d \rightarrow q$ in terms of a logic means that the information captured by d is sufficient to infer the information represented by q; that is, the document is relevant to the query.

8.2.2 Features of information in Information Retrieval

Several essential features of information in an IR system were identified:

(i) Flow of information: What information an object (e.g., a text, an image, a video) contains about another object (e.g., a query) is the main purpose of an IR system.

(ii) Intensionality: The meaning of an item of information is context-dependent.

(iii) Partiality: The representation of a document is only partial, but can grow when the implicit information in the document becomes available.

(iv) Structure: An underlying structure often accompanies documents and must be represented

appropriately to capture the semantics and pragmatics of information.

(v) Uncertainty: The flow of information that arises from a document's representation can be uncertain, which affects the relevance of the document to a query.

(vi) Significance: An item of information that occurs frequently in a document can imply that this item is a significant part of the document.

The first four features were qualitative whereas the last two were quantitative. Their modelling required different frameworks: a theory of information, and a theory of uncertainty, respectively.

The first objective of this thesis was to determine the appropriate framework for each, and to develop a method to combine them in a consistent manner.

8.2.3 The Transformation Principle

The combination was based on the Transformation Principle, which stated that a document had an initial representation, and was transformed until information relevant to the query was found. This transformation was based on the flow of information, the nature of which was determined by semantic-based relationships which constituted a knowledge set. The uncertainty of the flow was used to assess the relevance of the document to the query, whereby the more uncertainty involved in the transformation, the less relevant the document was to the query.

8.2.4 Which Theory of Information?

An initial study was performed to ascertain the appropriate theory of information to capture the qualitative features of information. It was shown that Classical Logic, the most commonly used logic, was inappropriate in modelling many of these features, and in particular those derived from the capturing of the flow of information. Others frameworks were then studied:

(i) Truth-based frameworks:
  (a) Three-valued Logic [Kle67],
  (b) Modal Logic [HC68, Che80],
  (c) Default Theory [Rei80],
  (d) Belief Revision [Gar88],
  (e) Epistemic Logic [Moo80], and
  (f) Cumulative Logic [KLM90].

(ii) Semantics-based frameworks:
  (a) Intensional Logic [PtMW90],
  (b) Montague Semantics [DWP81], and
  (c) Data Semantics [Lan86].

(iii) Information-based frameworks:
  (a) Situation Theory [Bar89, Dev91],
  (b) Channel Theory [Bar91, Bar92], and
  (c) Scott Domains [Sco82].

Situation Theory was shown to represent all the qualitative features.

8.2.5 Situation Theory

This theory proposes an analysis of both the concept of information, its flow, and the manner in which intelligent organisms, referred to as cognitive agents, handle and respond to the information derived or ascertained from their environment. An information item, represented as a type, is obtained from a situation by means of constraints to which a cognitive agent is attuned. These constraints model natural laws, conventions, analytic rules, linguistic rules, etc. Constraints also allow the derivation of additional information by permitting one situation to provide information about another, which corresponds to the flow of information. The theory can also model the uncertain nature of the flow of information, albeit qualitatively. This was done by associating background conditions to constraints, which were then qualified as conditional as opposed to unconditional. A conditional constraint holds if its background conditions are satisfied.

8.2.6 Which Theory of Uncertainty?

A second study was performed to assess the appropriate theory of uncertainty to model the quantitative features of information. Several theories of uncertainty were examined:

(i) Probabilistic-based frameworks:
  (a) Probability Theory [Goo50],
  (b) Bayesian Methods [Pea88], and
  (c) Imaging [Lew73].

(ii) Fuzzy Logic [Zad87].

(iii) Dempster-Shafer's Theory of Evidence [Dem68, Sha76].

The framework that best modelled the quantitative features was Dempster-Shafer's Theory of Evidence, together with the notion of refinement later introduced by Shafer [Sha76].

8.2.7 Dempster-Shafer's Theory of Evidence

The initial Dempster-Shafer Theory of Evidence [Dem68] models uncertainty by assigning belief values, through a BPA, to sets of propositions, referred to as focal elements, with respect to some gathered evidence. A belief function is used, based on that BPA, to compute the belief of the propositions. The initial framework allowed the representation of the significance of information. The refinement later defined by Shafer [Sha76] allowed the representations of the uncertainty, its propagation and aggregation. The use of the overall framework gave the advantage that it could be easily mapped to the qualitative structured representation of a document, and its transformation. Additionally, it could be suitably mapped onto Situation Theory.

The second objective of this thesis was to develop a logic-based model based on Situation Theory and the Dempster-Shafer Theory of Evidence.

This was done in two steps. First, the unstructured model was defined in which the structure and the significance of information were not accounted for. Second, that model was extended

into the structured model, which incorporated structures and the significance of information. This strategy was adopted because it enabled the careful representation of the flow of information to be performed initially. The expression of the two models borrowed some of the terminology identified in Data Semantics [Lan86] and Scott Domains [Sco82] because it lead to simpler definitions.

8.2.8 The Unstructured Model

The document and the query were modelled by a situation d and a type φ, respectively. The unconditional and conditional constraints together with a function *cert* constitute the representation of the knowledge set. They constitute the semantic relationships, together with the uncertainty pertaining to them. Any unconditional constraint, or any conditional constraint with satisfied background conditions, was certain, and delivered implicit and certain information; the resulting flow was certain. Otherwise, the constraint was uncertain, delivering uncertain and implicit information, and the value of its uncertainty was given by the function *cert*; the resulting flow was uncertain.

The extent to which d supported, explicitly or implicitly, φ was determined by $\Re(d, \varphi)$. If $d \models \varphi$ then the document was relevant to the query, and $\Re(d, \varphi) = 1$. The same arose if a certain flow lead to the information item φ.

If an uncertain flow lead to the information item φ, then d, referred to as the original document, was transformed into d', referred to as the transformed document, such that $d' \models \varphi$. If no other constraints were used to construct d', $\Re(d, \varphi)$ equated the uncertainty attached to the use of the constraint. This was given by the function *cert*.

In this thesis, the transformation of a document (situation) captured one instance of the flow of information restricted to the phenomenon that leads to the implicit information of the document from its explicit information content. That is, the transformation process was an addition of information, and was referred to as an extension process.

Several constraints in sequence/parallel may arrive at φ. In these cases, the resulting flow of information was modelled by sequential and/or parallel extensions of the situation d, which constitute branches. Uncertainty was propagated along the sequential extensions and aggregated along the parallel extensions. Branches then became quantified with uncertainty values. A branch was minimal to a query if its leaf (its end situation) was the only situation in that branch that supports the information being sought in that query, thus capturing minimal transformation. The uncertainty values of the minimal branches whose leafs supported information relevant to the query were aggregated to compute the degree of relevance. In the unstructured model, the uncertainty was represented by a general uncertainty mechanism.

8.2.9 The Structured Model

The unstructured model was extended to include structures. In this thesis, a structure was defined as that containing semantically related information items. The structures were modeled as basic situations, which were situations obtained from a semantic-based analysis of the document's sentences. These basic situations constituted an initial weighted information domain, which modelled a structured representation of the explicit, and implicit and certain information content

of the document. The Dempster-Shafer Theory of Evidence was used to model the significance of information. The focal elements corresponded to the basic situations, and the BPA reflected their significance.

The transformation process was identical to that in the unstructured model, except that it began from a set of situations, as opposed to just one situation. On the basis of the flow of information, a transformed document, referred to as a refined weighted information domain, which was itself a set of basic situations, was constructed. This was formulated with the notion of refinement defined by Shafer. A refined information domain modelled some stage in the application of the information flow on the set of basic situations that originally constituted the initial document's information content. Each stage lead to a structured representation of the explicit, implicit and certain, and part of the implicit and uncertain information content of the document. The uncertainty generated by the flow, (its propagation and aggregation) was measured by the BPA of the situations that constituted the refined representations. A modified version of this theory was used to conform with the different concepts used to represent the qualitative features of the model.

The refinement process was repeatedly performed until no more refinement was possible. The final refined information domain contained a set of basic situations, and their associated BPA values. A belief function defined in that refined representation was used to assess the relevance of the document to the query.

8.2.10 Specificity and Exhaustivity

It was shown that the relevance degree as computed by the structured model constituted a measure of specificity of the document to a query, which is the extent to which the information in the document related to the query. A measure of exhaustivity, which is the extent to which all the information sought by the query was contained in the document, was also defined. A measure that combined both exhaustivity and specificity into one expression of relevance degree of the document to the query was advanced.

The final objective of the thesis was to implement the unstructured and the structured models, as well as the exhaustivity and the combined measures, and to experiment with them to determine their validity.

8.2.11 Implementation

Types, which model information items, corresponded to terms. Constraints, which were semantic relationships between terms, were extracted from an existing on-line thesaurus, known as WordNet [Mil90]. The polysemic nature of WordNet terms was used to define whether a constraint was unconditional or conditional. The background conditions and the uncertainty attached to conditional constraints was based on the possible senses of terms in a situation. The basic situations were constructed based on WordNet, and their significance was computed using conventional IR weighting mechanisms [SM80].

8.2.12 Experiments and Evaluation

The experiments used the NPL test collection [vRRP80]. The results obtained with the unstructured model showed that more relevant documents were retrieved with the use of the WordNet relationships than without their use. The results also showed a high increase in the number of irrelevant documents retrieved with the use of WordNet relationships, and that the best performance in terms of precision and recall values [vR79] was obtained with synonyms. The results obtained with the structured model obtained were poor. This was because the relevance degree as evaluated by the structured model expressed a measure of specificity, which depended strongly on the fact that the information in the document was correctly organized into basic situations. WordNet did not allow an adequate construction of the basic situations for the NPL documents. Finally, the measure of exhaustivity was too strict and, as a result, the combined model collapsed into the structured model, thus giving poor results when compared to standard IR models.

Various issues were raised when analyzing these experiments. From them, further experiments were performed on a smaller number of queries, using only synonyms. The results obtained with these experiments were disappointing, in the sense that they were not as good as those obtained with standard IR models. However, after looking more closely at the documents, the queries, and more importantly, the WordNet thesaurus, obtaining positive results would have been difficult, because the constraints derived from the thesaurus provided semantic-based relationships were too general for the NPL documents. Nevertheless, the experiments showed that even with a bad capturing of the flow of information, the results, although not satisfactory, were still acceptable in the sense that they indicated that a better implementation of the constraints would derive better performances.

8.3 Limitations of this research

The model developed in this thesis was the first of its kind to capture within an uniform framework the different features of information as its appears in an IR system. However, the model had some limitations. These are discussed next.

8.3.1 The model is difficult to implement

The implementation of the model was difficult, mainly because it required appropriate data. For example, the background conditions which were primordial in representing the uncertain nature of flow of information, or in fact any reasoning process, are difficult to identify. In this thesis, the background conditions were implemented as terms senses. With other applications, implementing the background conditions may be impractical. Another example is that the relationships modelling the flow of information must be appropriately specified. In this thesis, the relationships captured thesaural information, which was derived from an on-line thesaurus. This implementation was inappropriate because the semantics provided by the thesaurus was too general for the NPL collection.

The transformation process relies critically on the fact that an appropriate indexing of the documents and queries was performed, and that accurate semantics were available. The indexing, as implemented in this thesis, was not refined. For example, single terms were independently

extracted from documents, and no disambiguation was carried out. Also, as above explained, the semantics used in the implementation were inappropriate. The transformation of a document could be refined to encompass the above two deficiencies (for example, with the use of heuristic rules). However, such a refinement may be constrained by resource factors (e.g., speed, memory space), and, hence, may not be practical, in particular for interactive systems.

This first limitation arises with most logic-based IR models (for a survey, see [Lal96b]). In addition to their complex implementation, often only small document collections can be handled. However, implementations on larger document collections have been recently attempted. Positive results were obtained in [CvR95a, CvR95b][93]. Furthermore, as in many areas of artificial intelligence [RN95], such as natural language processing, better technologies that can deal with complex formalisms, such as those involved with logics, are becoming increasingly more available. Therefore, the implementation of logic-based models, and hence, the one developed in this thesis, will become less complex, resulting in the elimination of this first limitation.

8.3.2 The model does not capture dependence in information

The formulations of the propagation and the aggregation of uncertainty assumed the independence of information and the background conditions. For example, the formulation of the aggregation did not consider that two situations extended into one situation may share common information. The information supported by the two situations was treated independently, as was its uncertainty. Another example is that any dependency relationship between the background conditions of conditional constraints was ignored in the extension of a situation. The only relationship considered was that of incompatible background conditions.

More adequate formulations of both the propagation and the aggregation of uncertainty should be investigated to capture dependent knowledge (e.g., information, constraints, background conditions). The capturing of dependency are known problems in the world of uncertainty theory [KC93].

However, formulations of the propagation and the aggregation of uncertainty that capture this dependence can be easily incorporated in the structured model. Indeed, the propagation and the aggregation of uncertainty are expressed by the relationships between the BPA associated to a weighted information domain and the BPA associated to its refined weighted information domain. To capture the dependence of information, it is then only necessary to reformulate the relationships between the two BPAs. The rest of the model can remain the same.

8.3.3 The transformation is implemented as an addition of information

The model captured only one information process; the addition of information. The model must be extended to incorporate modification and deletion of information. This will allow the capturing of different type of reasoning; in particular, non-monotonic [RN95]. For example, a deletion may indicate that what has been determined so far by the flow of information was incorrect; for example, the system had used the wrong sense of a polysemic term. The system then has to backtrack to an earlier state that is recognized as correct, by a user for example.

[93] For a description of the implementation of a logical model on large document collections, such as TREC, see [CRSvR96].

8.3.4 The model applies to textual information

The model dealt with textual documents only, and must be generalized to include any type of information. This is important due to the increasingly availability of multimedia documents in IR systems [Dun91, GS92].

8.4 Future Work

Future research areas are suggested in this section. Three directions are discussed: one which remedies the limitations listed in the previous section, one which shows the potential of the model to other applications, and one which allows the theoretical study of IR systems.

8.4.1 Improvements of the model

The model developed in this thesis needs enhancement to be sufficiently expressive for use as the basis for future generations of IR systems. For this purpose, the limitations listed in section 8.3 must be overcome. Three possible enhancements are discussed in this section.

8.4.1.1 Improving the model performance when implemented

Obtaining good performance is mandatory for the model to be used in real applications. One of the reasons for poor results was that the expression of the relevance of a document in the structured model to a query was a measure of specificity. The exhaustivity of the document to the query was not captured, unless computed separately, and then combined into the structured model. The fact that exhaustivity was not captured in the model may be because only the document's information content had a structured representation. A more appropriate measure of the relevance may be obtained if the information need of a query was also structured. This has the additional advantage that a richer semantic expression of the information need will be available. Such a representation is possible with the use of the Dempster-Shafer Theory of Evidence because the latter provides the notion of common refinement which can be used to formalize the comparison of a structured representation of the document's information content and to a structured representation of the query's information need.

8.4.1.2 Using better indexing and semantics

Poor experimental results were obtained for two other reasons. First, as explained in section 8.3, the implementation of the constraints was inappropriate. Better data should be used to implement these constraints. For example, some document collections come with their own thesaurus (e.g., the INSPEC collection). Better experimental results should be obtained with these collections.

Second, the transformation process, as implemented in this thesis, was inefficient due to a simplistic indexing of documents, and inadequate semantics. This deficiency can be overcome in two ways. First, the indexing process could be improved (e.g., disambiguating document terms, using noun-phrases, etc.), and/or proper semantics provided (see above). If this is not possible, then the transformation process must be controlled. Techniques applied in artificial intelligence may be used for this purpose. For example, in expert systems, heuristic rules are often used to constrain,

for instance, an inference process. Similar techniques could be used to control the transformation process, for example, by providing meta-information of the domain covered by the document collection. Alternatively, user feedback might help to improve direct transformation (see section 8.4.1.4).

8.4.1.3 Applying the model to various media of information

The model must be generalized to incorporate any information media. This is possible because the model is based on Situation Theory which is a framework concerned with the information carried by a situation, not by the way the information is carried. That is, Situation Theory is not concerned by how the information is delivered, but with what the information represents, since a situation can be a text, an image or speech. For example, a system that contains texts and images provides information; some of it comes from natural language, and some from the images. A model based on Situation Theory can cater to any kind of information, and does so according to the way in which information is handled in the real world. A flow can be associated with each medium. Obviously, there is still a gap with respect to the implementation of these flows. For example, how to represent, or index, the information contained in a picture? Future research is necessary to implement efficiently such a model for any kind of information, but some of the background theory is already presented here.

8.4.1.4 Generalization of the transformation process

The transformation of a document, to be generalized to capture any information process, must be modelled on a basis other than the extension of that document. As discussed in Chapter 2, channels [Bar92] can be used for this purpose. A channel is defined as the device that carries information (its flow) between one situation to another. It is a link between situations. Therefore, with channels, it becomes possible to represent a transformation as a modification or a deletion of information.

The modelling of a modification of information using channels was briefly discussed in Chapter 2. A channel carries the flow of information between two situations, and one of the situations can be viewed as the transformation of the other situation. The information supported by the original situation is not necessarily supported by the transformed situation. Some of it may be modified, for example, on the basis of semantics.

A general model of an IR system based on channels was proposed by Van Rijsbergen and Lalmas [vRL96]. They developed an information calculus to model an IR system, where channels were shown to possess properties that reflected the way the flow of information appears in an IR system (see [vRL96] for a description of the information calculus).

In the remainder of this section, the transformation defined as a deletion of information is discussed. This use allows incorporation of user interaction into the retrieval process. Such IR systems are referred to as interactive.

The model developed in this thesis can be generalized to capture user interaction. A situation can represent a state of the IR system (obtained by the flow of information) that is in accord with a user's belief. It can also represent a user's belief (or knowledge state). Often, users may change their beliefs (for example, when acquiring a new piece of information). This change of beliefs can be represented by a transformation process that corresponds to a deletion of information.

In the first case, the IR system must go back to a state (a situation) that is compatible with the user's beliefs. The situation is transformed to one previously obtained, and all the information that was acquired after this situation is deleted. In the second case, the user may have acquired a belief that contradicts those he or she already holds. For example, he or she realized that his or her interpretation of a term was erroneous. In that case, the transformation process represents a passage of beliefs; the transformed situation is built from the previous one, and the information supported by that situation is such that any introduced inconsistency is removed; some information is deleted (not necessarily that acquired last). Note that the transformation, here, applies to knowledge, and not document.

The model developed in this thesis can incorporate both of the above transformations; by modelling transformations by channels, instead of extensions. Consequently, flows other than semantics-based can be represented, for example, those reflecting intentionality (e.g., beliefs, knowledge), or pragmatics.

The way channels are determined depends on how a consistent state is obtained. For instance, the phenomenon discussed in the second example is often referred to as a belief revision [Gar88] (see Chapter 2). Techniques have been developed to capture this phenomena, and should be investigated to determine whether they can be used to define (and implement) channels. Similar work, but not related to IR, already exists with respect to Default Theory [Rei80] (both Default Theory and Belief Revisions are examples of belief systems — See Chapter 2). The aim of this work was to formalize a default logic within Situation Theory [Cav93].

To conclude, the model proposed in this thesis can be easily generalized to capture any type of transformation (or flow): addition, modification or deletion of information. As discussed below, by replacing the extension process by a channel, we have at our disposal a general concept which can model any transformation process, hence any flow of information. The way a channel is defined to fit a particular transformation depends on the application.

8.4.2 Applications of the model

The structured model can be applied to embody types of structures other than those based on semantically related information. Two possible applications are discussed in this section.

8.4.2.1 Application to pragmatic-based structures

An example of a pragmatic-based structure is one based on discourse [Kam91]. A document collection that consists of abstracts structured into discourses such as "purpose" "methodology", "result", etc., was constructed by Liddy [Lid91]. The collection consisted of 276 empirical abstracts from the ERIC and PsycINFO databases. The structured model can be used with this collection. Each discourse will be represented by a basic situation. The definition of a basic situation will have to be modified, since it will not be semantically based. The BPA attached to the basic situations will take into account the significance attached to each type of discourse, as well as the significance of the information in the discourses. A study will be necessary to rank discourse types.

Applying the model to these types of structures is of particular interested because it allows a more focussed retrieval process; specific parts of documents can be considered and retrieved. Indeed, such systems allow the formulation of precise queries of the form: "I am interested in the

methodologies used to study the effect of bat bites on humans". Such query is only concerned with the methodology discourse of documents. Such types of retrieval is referred to as passage retrieval (see [SAB93, Cal94]).

8.4.2.2 Application to linked documents

Two types of linked documents, HTML documents, and documents that cite other documents (e.g., articles) are discussed.

HTML [Int96] documents have been designed to be read by Web browsers. A HTML document is organized into structures delimited by tags. Examples of structures include title, paragraph, different heading levels, and links to other HTML documents. The structured model can be used for HTML documents retrieval. This is of particular importance because of the increasingly emphasize of network-based IR.

The structures defined by a HTML document can constitute the basic situations. Such an application is attractive for two reasons. First, it allows the incorporation of multimedia documents as well as textual (thus referring to one of the possible enhancements of the model discussed in section 8.4.1). Second, it may offer a solution to the difficult task of implementing the flow of information. Indeed, one limitation of the model, as explained in section 8.3, was that it required an adequate knowledge base, from which semantics and pragmatics of information can be extracted.

An implementation of the flow of information is partly accomplished with HTML documents because these types of documents can refer explicitly via anchors (or links) to other HTML documents. The fact that a document refers to another document could be viewed as the first document containing information about the second document; that is, there is a flow of information between the first document to the latter document. Obviously this is not always the case, for some anchors are randomly defined. However, these anchors still contain information, maybe pragmatic, about a user's interest (the owner of the document). With this application, a document is relevant to query if it contains information concerning the query, or if it refers to documents that contain information relevant to the query.

Documents can cite other documents. The CACM test collection consists of such documents. A transformed document is one that is referred to by another document. A flow of information arises between the two documents because the transformed document contains information on aspects discussed in the document citing it, or vice versa. The uncertainty of a transformation can be defined on the extent to which the original and the transformed documents are similar (the link between documents varies in strength). Similarity or statistical measures can be used for that purpose [vR79]. Also, the fact that a document refers several times to the same document can be viewed as a "stronger" flow of information than if only one reference was made.

The benefit of applying the model to these types of documents is that the evaluation of the relevance of a document to a query can take into account related documents, since citations are usually intentional. That is, the computation of the relevance of a document is not only with respect to itself, but with respect to related documents; thus, documents are not considered independently of each other any longer.

8.4.3 Theoretical study of Information Retrieval systems

A transformation is the result of a flow of information and relates two documents represented by situations. In this thesis, the flow of information captured thesaural relationships. For example, a flow delivers all the synonyms of the terms that appear in the document. There is nothing new here, for this process corresponds exactly to the way some IR systems function. The improvement is that the process can be formally described and, as a result, formally studied. This leads back to the remark that the use logic for IR modelling makes it possible to reason about IR systems and their properties, and that it allows the inductive comparison of IR systems. These issues are becoming increasingly important because an IR system's performance or behavior cannot always be explained by empirical evaluations.

For example, this thesis considered only the flow related with semantics of information. Another view could be that a flow models a retrieval method. Indeed, one can define several types of flows, one for each type of IR methods (Boolean, probabilistic, vector space or logical). A method can be used separately (i.e., one type of flow is involved) or can be combined with one or more other methods (i.e., parallel flows are involved). The document that is retrieved by many methods can be considered to be highly relevant to the information need. Obviously, it is necessary to define what a Boolean or a vector space flow is. The advantage of this approach is that, as well as being able to model different IR methods, the model can be used to compare them formally. The properties of the corresponding flows might lead to interesting results. Huibers and Bruza [HB94, BH94] are already researching this area.

8.5 Conclusions and contributions of this thesis

The work performed in this thesis was a first step towards the development of a general formalism to model an IR system. This was achieved by using a theory of information and a theory of uncertainty so that information as it appears in an IR system could be captured. Several essential features of information in an IR system were identified:

- flow
- partiality
- intensionality
- structure
- significance, and
- uncertainty.

The theory of information was Situation Theory and the theory of uncertainty was the Dempster-Shafer Theory of Evidence. The two theories were combined via the Transformation Principle. These particular theories were chosen because they allowed the appropriate representation of the above features.

The model developed in this thesis is the first of its kind to capture, in a general manner, the above features of information within a uniform framework. With a better understanding of the nature of information in IR, it became possible to first identify these features, and then to model

them appropriately. As a result, the model developed in this thesis can be easily generalized to be applicable to many types of IR systems (e.g., interactive and multimedia systems) or to capture many aspects of the IR process (e.g., user's knowledge).

References and Bibliography

[AK92] G. Amati and S. Kerpedjiev. An information retrieval logical model. Technical Report Rel 5B04892, Fondazione Ugo Bordoni, Roma, 1992.

[Arb92] Arbotext, Inc., Ann Arbor. *The Publisher^{TM}*, 1987–1992.

[Bar84] H. Barendregt. *The Lambda Calculus. Its syntax and semantics.* North-Holland, Amsterdam, New York, Oxford, 1984.

[Bar89] J. Barwise. *The Situation in Logic.* CLSI Lectures Notes 17, Stanford, California, 1989.

[Bar91] J. Barwise. Information links in domain theory. In S. Brooke & al., editor, *Proceedings of the Mathematical Foundation of Programming Semantics Conference*, pages 168–192. LNCS 598, Springer, 1991.

[Bar92] J. Barwise. Constraints, channels and the flow of information. In *Situation Theory and its Application*, volume III, 1992. (To appear).

[BB87] B. Boguraev and T. Briscoe. Large lexicons for natural language processing: Utilizing the grammar coding system of LDOCE. *Computational Linguistics*, 13, 1987.

[BE87] J. Barwise and J. Etchemendy. *The Liar: An Essay on Truth and Circularity.* Oxford University Press, New York, 1987.

[BE90] J. Barwise and J. Etchemendy. Information, infons, and inference. In *Situation Theory and its Applications*, volume I, pages 33–78. CSLI Lecture Notes 22, 1990.

[Ber88] C. Berrut. *Une méthode d'indéxation fondée sur l'analyse sémantique de documents spécialisés. Le prototype RIME et son application à un corpus médical.* PhD thesis, Université Joseph Fourier, Grenoble I, 1988.

[BGLY86] J. C. Bexdek, B. Gautam, and H. Li-Ya. Transitive closures of fuzzy thesauri for information-retrieval systems. *Int. Journal of Man-machine Studies*, 25:343–356, 1986.

[BH94] P. D. Bruza and T. W. C. Huibers. Investigating aboutness axioms using information fields. In W. C. Croft and C. J. van Rijsbergen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 112–121, Dublin, Ireland, 1994.

[BH95a] F. C. Berger and T. W. C Huibers. A framework based on situation theory for searching on a thesaurus. In J. Rowley, editor, *The new Review of Document and text Management, Proceedings of the 17th British Computer Society Information Retrieval Colloquium*, volume 1, pages 253–276, Crewe, England, 1995.

[BH95b] P. D. Bruza and T. W. C Huibers. How monotonic is aboutness? Technical report, Department of Computer Science, Utrecht University, The Netherlands, 1995. Technical Report UU-CS-1995–09.

[BH95c] P. D. Bruza and T. W. C Huibers. A study of aboutness in information retrieval. *Artificial Intelligence Review*, 1995. (To appear).

[Bla90] D. C. Blair. *Language and Representation*. Elsevier, 1990.

[Bla92] A. W. Black. *A Situation Theoretic Approach to Computational Semantics*. PhD thesis, University of Edinburgh, Scotland, 1992.

[Boo85] A. Bookstein. Probability and fuzzy-set applications to information retrieval. In M. E. Williams, editor, *Annual Review of Information Science and Technology*, pages 117–151. Knowledge Industries Publications, Inc., 1985.

[BP83] J. Barwise and J. Perry. *Situations and Attitudes*. Bradford Books, MIT Press, Cambridge, Massachusetts, 1983.

[BP93] G. Bordogna and G. Pasi. A fuzzy linguistic approach generalizing Boolean information retrieval: A model and its evaluation. *Journal of the American Society for Information Science*, 44(2):70–82, 1993.

[Bri62] L. Brilloun. *Science and Information Theory*. Academic Press Inc, 1962.

[Bru89] M. F. Bruandet. Outline of a knowledge-base model for an intelligent information retrieval system. *Information Processing & Management*, 25(1):89–115, 1989.

[Bru93] P. D. Bruza. *Stratified Information Disclosure: A synthesis between hypermedia and information retrieval*. PhD thesis, University of Nijmegen, 1993.

[BS75] A. Bookstein and D. Swanson. Probabilistic models for automatic indexing. *Journal of the American Society for Information Science*, 25(5):312–318, 1975.

[BS84] B. C. Buchanan and E. H. Shortliffe, editors. *Rule based expert systems: the MYCIN experiments of the Stanford heuristic programming project*. Addison-Wesley, 1984.

[BvdW91] P. D. Bruza and T. P. van der Weide. The modelling and retrieval of documents using index expressions. *SIGIR Forum*, 25(2):91–103, 1991.

[BvdW92] P. D. Bruza and T. P. van der Weide. Stratified hypermedia structures for information disclosure. *The Computer Journal, Special Issue*, 35(3):208–220, 1992.

[Cal91] P. G. Calabrese. Deduction and inference using conditional logic and probability. In I. R. Goomam, M. M. Gupta, H. T. Nguyen, and G. S. Rogers, editors, *Conditional*

Logic in Expert Systems. Elsevier Science Publishers B.V., North Holland, 1991.

[Cal94] J. Callan. Passage-level evidence in document retrieval. In W. B. Croft and C. J. van Rijsbergen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 302–310, Dublin, Ireland, 1994.

[Cav93] L. Cavedon. Conditional constraints and default reasoning. Centre for Cognitive Science, University of Edinburgh, 1993.

[CC92] Y. Chiaramella and J. P. Chevallet. About retrieval models and logic. *The Computer Journal, Special Issue*, 35(3):233–242, 1992.

[CC95] J. P. Chevallet and Y. Chiaramella. Extending a Logic-based Retrieval Model with Algebraic Knowledge. In I. Ruthven, editor, *MIRO '95, Proceedings of the Final Workshop on Multimedia Information Retrieval, Glasgow, Scotland*. Electronic Workshops in Computing. Springer-Verlag, 1995.

[CCLvR96] F. Crestani, I. Campbell, M. Lalmas, and C. J. van Rijsbergen. Is this document relevant?. . . probably. A survey of probabilistic models in Information Retrieval. Department of Computing Science, University of Glasgow, Scotland (Unpublished), 1996.

[CGD92] W. S. Cooper, F. C. Gey, and D. P. Dabney. Probabilistic retrieval based on staged logistic regression. In N. Belkin, P. Ingwersen, and A. M. Pejtersen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 198–210, Copenhagen, Denmark, 1992.

[Che80] B. F. Chellas. *Modal Logic: An introduction*. Cambridge University Press, 1980.

[Che92] J. P. Chevallet. *Un modèle logique de recherche d'information appliqué au formalisme des graphes conceptuels. Le prototype ELEN et son expérimentation sur un corpus de composants logiciels*. PhD thesis, Université Joseph Fourier, Grenoble I, 1992.

[Che94] P. S. Chen. On inference rules of logic-based information retrieval systems. *Information Processing and Management*, 30(1):43–60, 1994.

[Chu41] A. Church. *The calculi of lambda conversion*. Princeton University Press, Princeton, 1941.

[CMP90] R. Cooper, K. Mukai, and J. Perry, editors. *Situation Theory and its Applications*, volume I. CSLI Lecture Notes 22, Stanford, California, 1990.

[CN90] Y. Chiaramella and J. Nie. A retrieval model based on an extended Modal Logic and its application to the RIME experiment approach. In J. L. Vidick, editor, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 25–43, Brussels, Belgium, 1990.

[Con87] J. Conklin. Hypertext: an introduction and a survey. *IEEE Computer*, 20(9):17–41, 1987.

[Coo] R. Cooper. Introduction to situation semantics. (In progress).

[Coo91a] R. Cooper. Situation theoretic grammar. The Third European Summer School in Language, Logic and Information, Universität des Saarlandes, 1991.

[Coo91b] W. S. Cooper. Some inconsistencies and misnomers in probabilistic information retrieval. In A. Bookstein, Y. Chiaramella, G. Salton, and V. V. Raghavan, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 57–61, Chicago, Illinois USA, 1991.

[CRJ92] A. Cawsey, S. Reece, and K. Sparck Jones. Automating the librarian: Belief revision as a basis for system action and communication with the user. *The Computer Journal, Special Issue*, 35(3):221–232, 1992.

[Cro90] C. J. Crouch. An approach to automatic construction of global thesauri. *Information Processing & Management*, 26(5):629–640, 1990.

[Cro92] W. B. Croft. Text retrieval and inference. In P. S. Jacobs, editor, *Text-Based Intelligent Systems. Current Research and Practice in Information Extraction and Retrieval*, pages 127–155. Lawrence Erlbaum Associates, 1992.

[CRSvR96] F. Crestani, I. Ruthven, M. Sanderson, and C. J. van Rijsbergen. The troubles with using a logical model of IR on a large collection of documents. Experimenting retrieval by logical imaging on TREC. In *Proceedings of TREC–4*, 1996. (To appear).

[CS57] H. T. Clifford and W. Stephensen. *An introduction to numerical classification.* Academic Press, 1957.

[Cum89] R. Cummins. *Meaning and Mental Representation.* Bradford Book, MIT, Cambridge, 1989.

[CvR94] F. Crestani and C. J. van Rijsbergen. Information retrieval by imaging. In *Proceedings of the 16th British Computer Society Information Retrieval Colloquium*, Drymen, Scotland, 1994. (To appear).

[CvR95a] F. Crestani and C. J. van Rijsbergen. Information retrieval by logical imaging. *Journal of Documentation*, 51(1):3–17, 1995.

[CvR95b] F. Crestani and C. J. van Rijsbergen. Probability kinematics in information retrieval. In E. A. Fox, P. Ingwersen, and R. Fidel, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 291–299, Seattle, Washington USA, 1995.

[CY92] C. J. Crouch and B. Yang. Experiments in automatic thesaurus construction. In N. Belkin, P. Ingwersen, and A. M. Pejtersen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 77–88, Copenhagen, Denmark, 1992.

[Dem68] A. P. Dempster. A generalization of the Bayesian inference. *Journal of Royal Statistical Society*, 30:205–447, 1968.

[Den64] S. F. Dennis. The construction of a thesaurus automatically from a simple text. In *Sta-*

tistical Association Methods for Mechanized Documentation, Symposium Proceedings, Washington, pages 61–148, 1964.

[Dev91] K. J. Devlin. *Logic and Information*. Cambridge University Press, Cambridge, England, 1991.

[DGH76] R. Duda, J. Gaschnig, and P. Hart. Model Design in the Prospector Consultant System for Mineral Exploitation. *Expert Systems and AI Applications*, pages 334–348, 1976.

[DHN76] R. O. Duda, P. E. Hart, and N. J. Nilson. Subjective Bayesian methods for rules-based inference systems. In *AFIPS Proceedings*, volume 45, pages 1075–1082, New York, 1976.

[DKZ85] D. R. Dowty, L. Karttunen, and A. M. Zwicky, editors. *Natural Language Processing: Theoretical, Computational and Psychological Perspectives*. Cambridge University Press, New York, 1985.

[Doy75] L. B. Doyle. *Information Retrieval and Processing*. Melville, 1975.

[DP85] D. Dubois and H. Prade. Combination and propagation of certainty with belief functions - a reexamination. In *Proceedings of 9th IJCAI*, Los Angeles, 1985.

[DP86] D. Dubois and H. Prade. A tentative comparison of numerical approximate reasoning methodologies. In *Workshop on IDSS for Plant Operation, CCE*, Ispra, 1986.

[DP87] D. Dubois and H. Prade. Combinations d'information incertaines. Technical Report L. S. I 263, Equipe Intelligence Artificielle et Robotique, Communication, Decision, Raisonement, 1987.

[DP90] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 1990.

[Dre81] F. Dretske. *Knowledge and The Flow of Information*. Bradford Books, MIT Press, Cambridge, Massachusetts, 1981.

[dSM93] W. Teixera de Silva and R. L. Milidiu. Belief function model for information retrieval. *Journal of the American Society of Information Science*, 4(1):10–18, 1993.

[Dun91] M. D. Dunlop. *Multimedia Information Retrieval*. PhD thesis, University of Glasgow, Scotland, 1991.

[DWP81] D. R. Dowty, R. E. Wall, and S. Peters. *Introduction to Montague Semantics*. Studies in Linguistics and Philosophy. D. Reidel Publishing Company, 1981.

[ET96] J. Ellman and J. Tait. INTERNET Challenges for Information Retrieval. In *Proceedings of the 18th British Computer Society Information Retrieval Colloquium*, pages 1–12, Manchester Metropolitan University, England, UK, 1996.

[Eva82] G. Evans. *The Variety of Reference*. Oxford University Press, 1982.

[Fag87] J. L. Fagan. *Experiments in Automatic Phrase Indexing for Document Retrieval: A*

Comparison of Syntactic and Non-Syntactic Methods. PhD thesis, Cornell University, Ithaca, New York, 1987.

[Far80a] J. Farradane. Relational indexing, part I. *Journal of Information Science*, 1(5):267–276, 1980.

[Far80b] J. Farradane. Relational indexing, part II. *Journal of Information Science*, 1(6):313–324, 1980.

[Fau78] R. Faure. *Précis de Recherche Opérationnelle.* Dunod Decision, 1978.

[FBY92] W. B. Frakes and R. Baeza-Yates, editors. *Information Retrieval: Data Structures & Algorithms.* Prentice Hall, 1992.

[Fer90] T. Fernando. *Mathematical foundations of situation theory.* PhD thesis, Stanford University, 1990.

[FLV87] J. E. Fenstad, T. Langholm, and E. Vestre. *Situations, Language and Logic.* Reidel, Dordrecht, 1987.

[Fox89] E. A. Fox. Research and development of information retrieval models and their application. *Information Processing & Management*, 24(1):1–6, 1989.

[Fre60] G. Frege. *Translations from the Philosophical Writings of Gottlob Frege.* Oxford, Blackwell, 2nd edition, 1960.

[Fro86] R. A. Frost. *Introduction to Knowledge Base Systems.* William Collins Sons and Co. Ltd, 1986.

[Fuh92] N. Fuhr. Probabilistic models in information retrieval. *The Computer Journal, Special Issue*, 35(3):243–255, 1992.

[Gal87] J. H. Gallier. *Logic for Computer Science. Foundations of Automatic Theorem Proving.* John Wiley & Sons, 1987.

[Gar88] P. Gardenfors. *Knowledge in Flux: Modelling the Dynamics of Epistemic States.* MIT Press, 1988.

[Goo50] I. J. Good. *Probability and the Weighing of Evidence.* Charles Griffin & Company Limited, 1950.

[Gra71] G. Gratzer. *Lattice-Theory. First concepts and distributive lattice.* San Francisco, Freeman and Co., 1971.

[GS92] U. Glavitsch and P. Schauble. A system for retrieving speech documents. In N. Belkin, P. Ingwersen, and A. M. Pejtersen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 168–176, Copenhagen, Denmark, 1992.

[HB94] T. W. C. Huibers and P. D. Bruza. Situations, a general framework for studying information retrieval. In *Proceedings of the 16th British Computer Society Information*

Retrieval Colloquium, Drymen, Scotland, 1994. (To appear).

[HC68] G. E. Hughes and M. J. Cresswell. *An Introduction to Modal Logic*. London, Methuen, 1968.

[HC84] G. E. Hughes and M. J. Cresswell. *A Companion to Modal Logic*. Methuen, London, 1984.

[HD95] T. W. C Huibers and N. Denos. A qualitative ranking method for logical information retrieval models. Technical Report RAP95-005, Groupe MRIM of the Laboratoire de Génie Informatique, Grenoble, France, 1995.

[HJ91] S. P. Harbison and G. L. Steele Jr. *C A reference Manual*. Prentice Hall, Engliwood Cliffs, NJ 07632, 1991.

[HLvR96] T. W. C. Huibers, M. Lalmas, and C. J. van Rijsbergen. Information Retrieval and Situation Theory. Technical Report UU-CS-1996–04, Department of Computer Science, Universiteit Utrecht, 1996.

[HOC95] T. Huibers, I. Ounis, and J. P. Chevallet. Axiomatization of a conceptual graph formalism for information retrieval in a situated framework. Technical Report RAP95–004, Group MRIM of the Laboratoire de Génie Informatique, Grenoble, France, 1995.

[Hoe66] P. G. Hoel. *Elementary Statistics*. Wiley Series in Probability and Mathematical Statistics, second edition, 1966.

[HSP81] W. L. Harper, R. Stalnaker, and G. Pearce. *Ifs*. D. Reidel Publishing Company, 1981.

[Hun95] A. Hunter. Using default logic in information retrieval. In *Symbolic and Quantitative Approaches to Uncertainty*, volume 946, pages 235–242. Lectures Notes in Computing Science, 1995.

[HvL96] T. W. C. Huibers and B. van Linder. Formalising Intelligent Information Retrieval Agents. In *Proceedings of the 18th British Computer Society Information Retrieval Colloquium*, pages 125–143, Manchester Metropolitan University, England, UK, 1996.

[HvLB94] T. W. C. Huibers, B. van Linder, and P. D. Bruza. Een theorie voor het bestuderen van information retrieval modellen. In L. G. M. Noordman and W. A. M. de Vroom, editors, *Een theorie voor het bestuderen van information retrieval modellen*, pages 85–102, Stichting StinfoN, 1994. (In Dutch).

[Ing94] P. Ingwersen. Polyrepresentation of information needs and semantic entities: Elements of a cognitive theory for information retrieval interaction. In W. B. Croft and C. J. van Rijsbergen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 101–110, Dublin, Ireland, 1994.

[Int96] *The INTERNET Unleashed*. Sams Publishing, 1996.

[Jan80a] T. M. V. Janssen. *Foundations and Applications of Montague Grammar. Part 1: Philosophy, Framework, Computer science*. Number 19. Centre for Mathematics and

Computer Science, 1980.

[Jan80b] T. M. V. Janssen. *Foundations and Applications of Montague Grammar. Part 2: Applications to Natural Language.* Number 28. Centre for Mathematics and Computer Science, 1980.

[Jef83] R. C. Jeffrey. *The Logic of Decision.* University of Chicago Press, second edition, 1983.

[Jon71] K. Sparck Jones. *Automatic keyword Classification for Information Retrieval.* Butterworths, 1971.

[Jon74] K. Sparck Jones. Automatic indexing. *Journal of Documentation*, 30:393–432, 1974.

[Jow90] H. E. Jowsey. *Constraining Montague Grammar for Computational Applications.* PhD thesis, University of Edinburgh, 1990.

[JS71] N. Jardine and R. Simson. *Mathematical Taxonomy.* John Wiley & Sons Ltd, 1971.

[Kam91] H. Kamp. Procedural and cognitive aspects of propositional attitudes contexts. The 3rd European Summer School in Language, Logic and Information, Universität des Saarlandes, 1991.

[KC92] R. Krovetz and W. B. Croft. Lexical ambiguity and information retrieval. In *ACM Transactions on Information System*, number 10, 1992.

[KC93] P. Krause and D. Clark. *Representing Uncertain Knowledge. An Artificial Intelligence Approach.* Intellect, Oxford, England, 1993.

[Kle67] S. C. Kleene. *Mathematical Logic.* New York, Wiley, 1967.

[KLM90] S. Krauss, D. Lehmann, and M. Magidor. Non-monotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence*, 44:167–207, 1990.

[Klu74] H. E. Klugh. *Statistics: The Essentials for Research.* John Wiley & Sons, 1974.

[Kra91] M. Kracker. A fuzzy concept network model and its application. Technical report, Gesellschaft fur Mathematik und Datenverbeitung MBH, 1991.

[Kri63] S. A. Kripke. Semantic analysis of modal logic. *Zeitschrift fur Mathematische Logik und Grundlagen der Mathematik*, 9:67–96, 1963.

[Kuh64] J. L. Kuhns. The continuum of coefficients of association. In *Statistical Association Methods for Mechanized Documentation, Symposium Proceedings, Washington*, pages 33–39, 1964.

[Lal95a] M. Lalmas, editor. *First Workshop on the treatment of uncertainty in logic-based models of information retrieval systems.* Technical Report, TR-1995-18, Department of Computing Science, University of Glasgow, Scotland, 1995.

[Lal95b] M. Lalmas. From a qualitative towards a quantitative representation of uncertainty on a situation theory based model of an information retrieval system. In *First Workshop*

on the treatment on uncertainty in logic-based models of information retrieval systems. Technical Report TR-1995-18, Department of Computing Science, University of Glasgow, Scotland, 1995.

[Lal96a] M. Lalmas. Modelling Information Retrieval with the Dempster-Shafer Theory of Evidence: A Study. In *ECAI Workshop on Uncertainty in Information Systems: Questions of Viability.*, Budapest, 1996. (To appear).

[Lal96b] M. Lalmas. The use of logic in information retrieval modelling. Technical Report TR-1996–1, Department of Computing Science, University of Glasgow, Scotland, 1996.

[Lan71] S. Mac Lane. *Categories for the Working Mathematician.* Springer-Verlag, New York Heidelberg Berlin, 1971.

[Lan86] F. Landman. *Towards a Theory of Information. The status of partial objects in semantics.* Dordrecht, Foris, 1986.

[Lew73] D. Lewis. *Counterfactuals.* Cambridge, Harvard University Press, 1973.

[Lew75] D. Lewis. Probabilities of conditionals and conditional probabilities. *In Philosophical Review, Kluwer Academic Publishers*, 85:297–315, 1975.

[Lew76] D. Lewis. Probability of conditionals and conditional probabilities. *Philosophical Review*, LXXXV(3):297–315, 1976.

[Lid91] E. D. Liddy. The discourse level structure of empirical abstracts: An exploratory study. *Information Processing and Management*, 27(1):55–81, 1991.

[LL93] D. D. Lewis and E. D. Liddy. Natural language processing for information retrieval. In *Association for Computational Linguistics. 31st Annual Meeting. ACL-93*, 1993. Tutorial Note.

[LRJ94] B. Logan, S. Reece, and K. Sparck Jones. Modelling information retrieval agents with belief revision. In W. C. Croft and C. J. van Rijsbergen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 91–100, Dublin, Ireland, 1994.

[Luk90] W. Lukaszewicz. *Non-Monotonic Reasoning.* Series in Artificial Intelligence. Ellis Horwood, 1990.

[LvR92] M. Lalmas and C. J. van Rijsbergen. A Logical Model of Information Retrieval based on Situation Theory. In *Proceedings of 14th British Computer Society Information Retrieval Colloquium*, pages 1–13, Lancaster, 1992.

[LvR93] M. Lalmas and C. J. van Rijsbergen. Situation Theory and Dempster-Shafer's Theory of Evidence for Information Retrieval. In *Proceedings of Workshop on Incompleteness and Uncertainty in Information Systems*, pages 62–67, Concordia University, Montreal, Canada, 1993.

[Mar92] E. L. Margulis. N-poisson document modelling. In N. Belkin, P. Ingwersen, and A. M.

Pejtersen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 177–189, Copenhagen, Denmark, 1992.

[Maz94] Z. Mazur. Models of a distributed information retrieval system based on thesauri with weights. *Information Processing and Management*, 30(1):61–78, 1994.

[Meg95] C. Meghini. An image retrieval model based on classical logic. In E. A Fox, P. Ingwersen, and R. Fidel, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 300–308, Seattle, Washington, USA, 1995.

[Mil90] G. A. Miller. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–312, 1990.

[MK95] A. Muller and S. Kutschekmanesch. Using Abductive Inference and Dynamic Indexing to Retrieve Multimedia SGML documents. In I. Ruthven, editor, *MIRO '95, Proceedings of the Final Workshop on Multimedia Information Retrieval, Glasgow, Scotland.* Electronic Workshops in Computing. Springer-Verlag, 1995.

[MMN83] S. Miyamoto, T. Miyake, and K. Nakayama. Generation of a pseudothesaurus based on cooccurences and fuzzy set operations. In *IEEE Transactions on Systems, Man and Cybernetics*, volume SMC-13, pages 62–70, 1983.

[Mon74] R. Montague. *Formal Philosophy*. New haven, Yale University, 1974.

[Moo80] R. C. Moore. Reasoning about knowledge and action. Technical Report 191, SRI, 1980.

[Mor90] J. M. Morrissey. Imprecise information and uncertainty in information systems. *ACM Transactions on Information Systems*, 8(2):159–180, 1990.

[Mor92] M. Morreau. Epistemic semantics for conterfactuals. *Journal of Philosophical Logic, Kluwer Academic Publishers*, 21:33–62, 1992.

[Mos91] L. Moss. Foundations of situation theory. The 3rd European Summer School in Language, Logic and Information, Universität des Saarlandes, 1991.

[MSST93] C. Meghini, F. Sebastiani, U. Straccia, and C. Thanos. A model of information retrieval based on terminological logic. In E. Rasmussen R. Korfhage and P. Willet, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 298–307, Pittsburgh, USA, 1993.

[Nea90] R. E. Neapolitan. *Probabilistic reasoning in expert systems*. John Willey & Son Inc., 1990.

[Nie88] J. Y. Nie. An outline of a general model for information retrieval. In Y. Chiaramella, editor, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 495–506, Grenoble, France, 1988.

[Nie89] J. Y. Nie. An information retrieval model based on modal logic. *Information Processing & Management*, 25(5):477–491, 1989.

[Nie90] J. Y. Nie. *Un Modèle de Logique Générale pour les Systemes de Recherche d'Informations. Application au Prototype RIME.* PhD thesis, Université Joseph Fourrier, Grenoble I, 1990.

[Nie92] J. Y. Nie. Towards a probabilistic modal logic for semantic-based information retrieval. In N. Belkin, P. Ingwersen, and A. M. Pejtersen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 140–141, Copenhagen, Denmark, 1992.

[NLB96] J. Y. Nie, F. Lepage, and M. Brisebois. Information retrieval as counterfactual. *The Computer Journal*, 1996. (To appear).

[Nut80] D. Nute. *Topics in Conditional Logic.* D. Reidel, Publishing Company, 1980.

[OLB+92] R. N. Oddy, E. D. Liddy, B. Balakrishnan, J. Elewononi, and E. Martin. Towards the use of situational information in information retrieval. *Journal of Documentation*, 48(2):123–171, 1992.

[Ous94] J. K. Ousterhout. *Tcl and the Tk Toolkit.* Addison-Wesley, 1994.

[Pac91] D. P. Pace. A thesaural model of information retrieval. *Information Processing and Management*, 27(5):433–447, 1991.

[Par94] J. B. Paris. *The Uncertain Reasoner's Companion. A Mathematical Approach.* Cambridge University Press, 1994.

[Pea88] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference.* Morgan Kaufman Publishers. Palo Alto, 1988.

[Por80] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[PS86] P. F. Patel-Schneider. A four-valued semantics for frame-based description languages. In *AAAI-86, 5th Conference of the American Association for Artificial Intelligence*, pages 344–348, Philadelphia, 1986.

[PtMW90] B. H. Partee, A. ter Meulen, and R. E. Wall. *Mathematical Methods in Linguistics*, volume 30 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, 1990.

[Ram88] A. Ramsey. *Formal Methods in Artificial Intelligence.* Cambridge University Press, 1988.

[Rei78] R. Reiter. *On Closed-World Databases*, pages 55–76. Plenum Press, New York, 1978.

[Rei80] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1):81–132, 1980.

[RN95] S. Russell and P. Norvig. *Artificial Intelligence. A modern Approach.* Prentice Hall International Editions, 1995.

[Rob77] S. E. Roberston. The probability ranking principle in IR. *Journal of Documentation*, 33:294–304, 1977.

[RSM94] R. Richardson, A. F. Smeaton, and J. Murphy. Using WordNet for conceptual distance measurement. In *Proceedings of the 16th British Computer Society Information Retrieval Colloquium*, Drymen, Scotland, 1994. (To appear).

[Rug92] G. Ruge. Experiments on linguistically-based term associations. *Information Processing and Management*, 28(3):317–332, 1992.

[SAB93] G. Salton, J. Allan, and C. Buckley. Approaches to passage retrieval in full text information systems. In R. Korfhage, E. Rasmussen, and P. Willet, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 49–58, Pittsburgh, USA, 1993.

[Saf87] A. Saffioti. An AI view of the treatment of uncertainty. *The Knowledge Engineering Review*, 2(2):75–97, 1987.

[Sal71] G. Salton. *The SMART Retrieval System.* Englewood Cliffs, N. J. , Prentice Hall, Inc, 1971.

[San94] M. Sanderson. Word sense disambiguation and information retrieval. In W. B. Croft and C. J. van Rijsbergen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 142–151, Dublin, Ireland, 1994.

[Sav92] J. Savoy. Bayesian inference networks and spreading activation in hypertext systems. *Information Processing and Management*, 28(3):389–406, 1992.

[SB64] J. Spiegel and E. Bennett. A modified statistical association procedure for automatic document content analysis and retrieval. In *Statistical Association Methods for Mechanized Documentation, Symposium Proceedings*, pages 47–60, 1964.

[SBS89] G. Salton, C. Buckley, and M. Smith. On the application of syntactic methodologies in automatic text analysis. *Information Processing and Management*, 26(1):73–92, 1989.

[Sco82] D. S. Scott. *Domains for denotational semantics.* Number 140. Lecture Notes in Computing Science, 1982.

[Seb94] F. Sebastiani. A probabilistic terminological logic for modelling information retrieval. In W. B. Croft and C. J. van Rijsbergen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 122–131, Dublin, Ireland, 1994.

[Seb95] F. Sebastiani. A note on logic and information retrieval. In I. Ruthven, editor, *MIRO '95, Proceedings of the Final Workshop on Multimedia Information Retrieval, Glasgow, Scotland.* Electronic Workshops in Computing. Springer-Verlag, 1995.

[Sel90] J. M. Seligman. *Perspectives: A relativistic approach to the theory of information.* PhD thesis, University of Edinburgh, 1990.

[Sem88] H. F. Sem. Discourse representation theory, situation schemata, and situation semantics, a comparison. Technical Report COSMOS-Report 2, Computational Semantics, Department of Mathematics, University of Oslo, 1988.

[Sem89] T. M. T. Sembok. *Logical-Linguistic Model and Experiments in Document Retrieval.* PhD thesis, University of Glasgow, Scotland, 1989.

[Sha76] G. Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, 1976.

[Sha81] G. Shafer. Jeffrey's rule of conditioning. *Philosophy of Science*, 48:337–362, 1981.

[SM80] G. Salton and M. J. McGill. *Introduction to modern information retrieval.* McGraw-Hill Book Company, 1980.

[Sme88] A. F. Smeaton. *Using Parsing of Natural Language as part of Document Retrieval.* PhD thesis, Dublin, 1988.

[Sme92] A. F. Smeaton. Progress in the application of natural language processing to information retrieval tasks. *The Computer Journal, Special Issue*, 36(3):268–278, 1992.

[Sri91] P. Srinivasan. Thesaurus construction. In W. Frakes and R. Baeza-Yates, editors, *Information Retrieval: Data Structures & Algorithms*, pages 161–218. Prentice Hall, 1991.

[SS73] P. H. A. Sneath and R. R. Sokal. *Numerical Taxonomy.* W. H. Freeman and Company, San Francisco, 1973.

[Sta84] R. Stalnaker. *Inquiry.* A Bradford Book, The MIT Press, 1984.

[Str95] U. Straccia. Document Retrieval by Relevance Terminological Logic. In I. Ruthven, editor, *MIRO '95, Proceedings of the Final Workshop on Multimedia Information Retrieval, Glasgow, Scotland.* Electronic Workshops in Computing. Springer-Verlag, 1995.

[SvR90] T. M. T. Sembok and C. J. van Rijsbergen. Silol: A simple logical-linguistic document retrieval system. *Information Processing & Management*, 26(1):111–134, 1990.

[SvR93] T. M. T. Sembok and C. J. van Rijsbergen. Imaging: a relevant feedback retrieval with nearest neighbor clusters. In R. Leon, editor, *Information Retrieval. New Systems and Current Research. Proceedings of the 15th British Computer Society Information Retrieval Colloquium*, pages 91–107, Glasgow, Scotland, 1993.

[SWY76] G. Salton, A. Wong, and C. T. Yu. Automatic indexing using term discrimination and term precision measurements. *Information Processing and Management*, 12:43–51, 1976.

[Tur84] R. Turner. *Logics for artificial intelligence.* Ellis Horwood Limited, 1984.

[Tur90] H. R. Turtle. *Inference Network for Document Retrieval.* PhD thesis, University of Massachusetts at Amherst, 1990.

[vB83] J. van Benthem. *The Logic of Time.* Dordrecht, Reidel, 1983.

[vB85] J. van Benthem. *A Manual of Intentional Logic.* University of Chicago Press, 1985.

[vB90] J. van Benthem. Modal logic as a theory of information. In *Prior Memorial Colloquium*, Christchurch, 1990.

[Voo93] E. M. Voorhees. Using WordNet to disambiguate word sense for text retrieval. In R. Korfhage, E. Rasmussen, and P. Willet, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 171–180, Pittsburgh, PA USA, 1993.

[vR79] C. J. van Rijsbergen. *Information Retrieval.* Butterworths, London, 2 edition, 1979.

[vR86a] C. J. van Rijsbergen. A new theoretical framework for information retrieval. In F. Rabitti, editor, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 194–200, Pisa, Italy, 1986.

[vR86b] C. J. van Rijsbergen. A non-classical logic for information retrieval. *The Computer Journal*, 29(6):481–485, 1986.

[vR89] C. J. van Rijsbergen. Towards an information logic. In N. J. Belkin and C. J. van Rijsbergen, editors, *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 77–86, Cambridge, Massachusetts USA, 1989.

[vR92] C. J. van Rijsbergen. Probabilistic retrieval revisited. *The Computer Journal, Special Issue*, 35(3):291–298, 1992.

[vR93] C. J. van Rijsbergen. Two essays in information retrieval. Technical Report IR-93–3, Department of Computing Science, Glasgow University, Scotland, 1993.

[vRL96] C. J. van Rijsbergen and M. Lalmas. An Information Calculus for Information Retrieval. *Journal of the American Society of Information Science*, 47(5):385–398, 1996. (To appear).

[vRRP80] C. J. van Rijsbergen, S. E. Robertson, and M. F. Porter. New models in probabilistic information retrieval. Technical Report 5587, British Library R&D Report, Computer Laboratory, University of Cambridge, 1980.

[Wat85] D. A. Waterman. *A guide to expert systems.* Addison-Wesley, 1985.

[WF82] M. Wolfenson and T. L. Fine. Bayes-like decision making with upper and lower probabilities. *Journal of the American Statistical Association, Theory and Methods Section*, 77(377):80–88, 1982.

[Win83] T. Winograd. *Language as a Cognitive Process. Volume I : Syntax.* Addison-Wesley Publishing Company, 1983.

[WY89] S. K. M. Wong and Y. Y. Yao. A probability distribution model for information retrieval. *Information Processing & Management*, 25:39–53, 1989.

[WY91] S. K. M. Wong and Y. Y. Yao. A probabilistic inference model for information retrieval. *Information Systems*, 16(3):301–321, 1991.

[Zad65] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[Zad78] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets Systems*, 1:3–28, 1978.

[Zad87] L. A. Zadeh. *Fuzzy sets and Applications: Selected Papers.* Wiley, New York, 1987.

[Zal88] E. N. Zalta. *Intensional Logic and the Metaphysics of Intentionality.* A Bradford Book, The MIT Press, Cambridge, Massachusetts, London, England, 1988.

[Zim91] H. J. Zimmermann. *Fuzzy set theory and its applications.* Kluwer Academic Publishers, second edition, 1991.