A model for representing and retrieving heterogeneous structured documents based on evidential reasoning

Mounia Lalmas*
Department of Computer Science
Queen Mary & Westfield College
University of London
mounia@dcs.qmw.ac.uk

Abstract

Documents often display an internal structure; they are composed of components. For example, a journal contains several articles, which themselves contain paragraphs, tables, etc. With structured documents, the retrievable units should be the document components as well as the whole document.

The components of a structured document can be of different types: various media, located in a number of sites, or written in several languages. An information retrieval model for *heterogeneous* structured documents must take into account this disparity among document components.

We present a model for representing and retrieving heterogeneous structured documents, that is multimedia, distributed and multilingual documents. The model is based on evidential reasoning, a formal theory that allows for the representation and the combination of knowledge. Here, knowledge is the content of document components. We show that the model provides for an appropriate representation and retrieval of heterogeneous structured documents.

1 Introduction and background

In traditional *Information Retrieval* (IR) systems, documents are retrieved as atomic units. However, often documents display an internal **structure**; they are composed of components. For example, an article may be divided into an introduction, several sections, each with subsections, and a conclusion; a conference proceedings contains several papers, each with its own structure. From the user's point of view, presenting only some components of documents can make it easier to distinguish potentially relevant documents from irrelevant ones. It can also make it easier for a user to target which components of the document may be most useful, especially for long documents, and documents that cover a variety of subjects.

With *structured documents*, the retrievable units should be the document components as well as the whole document. Also the retrieval process should return various levels of composite parts, for example, a section when only that section is relevant, a group of sections, when all the sections in the group are relevant, or the document itself, when the entire document is relevant. This is only possible if **the underlying IR model takes into account the inherent structure of the documents, in both representing and retrieving structured documents.**

*This work was carried out when the author was at Informatik VI, University of Dortmund, Germany.

In this work, the structure of a document corresponds to a tree whose nodes, referred to as *objects*, are the components of the document and whose edges represent the composition relationship (e.g., a chapter contains several sections). An object is considered to be an entity that has a coherent meaning in itself when displayed to the user. The *root* object of the tree embodies the whole document, and the *leaf* objects comprise the raw data (e.g., a piece of text, an image). Any non-leaf object is referred to as a *composite* object (the root object included).

In [1, 2, 3], a model for structured documents was advanced. The model, which is expressed within a framework based on formal logics, aims at providing two complimentary approaches for manipulating structured documents: browsing and querying. Browsing is done with respect to the structure of the documents. Querying can be of three kinds:

- structure query: selecting which part of the structure to retrieve (e.g., a title, a section, a title followed by a section);
- attribute query: specifying values for attributes associated with objects (e.g., author name, creation date);
- content query: seeking objects relevant to an information need.

In this paper, we concentrate on content queries, where the retrieval of documents is solely based on their content.

Chiaramella et al [1, 2] explain that, to allow for effective and efficient browsing and querying, content-based retrieval should exploit the structure of documents. The reasons are twofold. First, the *relationships between the retrieved document parts* should not be ignored. This aims at reducing "cognitive overload". For example, suppose that two document components, a chapter c and a section s, have been retrieved (by whatever techniques) for a given query. If the section s is part of chapter c, this information is not made explicit to a user until he or she browses down from chapter p or browses up from section s. Moreover, according to the ranking method, c and s will most probably be displayed at distant locations in the result. This redundancy has a negative impact on cognitive overload and wastes user time. It can also lead to user disorientation. Second, to be efficient, the retrieval should be *focussed*: if a composite object is not relevant to a query, then none of its component objects are relevant, and hence do not need to be evaluated for relevance.

To capture the relationships between document parts and enable focussed retrieval, Chiaramella et al define the representation of a composite object as the **aggregation** of the representations of its component objects. They also show that this approach makes it possible to return various **levels** of composite parts.

Their model, however, does not incorporate the uncertainty inherent to the representation of content. Due to the complex nature of information, representing the content of an objet or a document is an uncertain task because it often relies on incomplete evidence. For instance, it is not because a term has been extracted by the indexing algorithm that the term describes adequately the object.

To capture uncertainty, in [4], we extended the model developed by Chiaramella et al with the use of the *Dempster-Shafer theory of evidence* [5, 6]. We demonstrated in [4] and [7] the connections between their model and some functions offered by the Dempster-Shafer theory of evidence, in particular, the aggregation operation and the so-called *Dempster's combination rule*. We showed that our model appropriately provides for:

- representing individual and aggregated document components, *and* the uncertainty of their representation;

- calculating the relevance of a document or document component to a query;
- retrieving document components that are most relevant to a particular information need; and
- the properties of the aggregation rule are compatible with those proposed by Chiaramella et al.

In [8, 9] we carried out a range of experiments on structured text documents which showed that the use of the Dempster's combination rule to determine the representation of composite objects leads to:

- *effective retrieval*: the components of a document that are most relevant to a query are retrieved before those less relevant.
- *efficient retrieval*: if a composite object is not relevant to a query, then none of its component objects are relevant (focussed retrieval).

Although we did not perform experiments on non-text structured documents, our model can be applied to any collection of structured documents for which the *indexing vocabulary* is common to all objects. With our model, it is necessary to determine an indexing vocabulary that is common to all document components, and all documents of the collection. This poses a problem for **multimedia, distributed, and multilingual documents**, for which the objects composing a document can be of different media, distributed over several sites, or written in various languages.

For instance, a structured document can be composed of image objects, text objects and video objects (e.g., web documents, illustrated on-line books, newspapers). Different indexing vocabularies are used to represent the content of objects of different media. A structured document can have some components in one database, and other components in another database (e.g., web documents, documents in a digital library). The indexing vocabulary (e.g., terms) in one site may not be the same as that in another site. Finally, a structured document can be composed of objects whose textual content is written in various languages (e.g., the web pages of a university in a non-English speaking country are often written in both English and the language of the country of the university). Different indexing vocabularies apply to different languages.

The model developed in [4] cannot deal with **heterogeneous** structured documents, that is multimedia, distributed and multilingual structured documents. This is because the aggregation as defined by the Dempster's combination rule requires that the indexing vocabulary is the same for all objects. **Therefore, we need a more general model that allows for the disparity of indexing vocabularies**. For this purpose, we require:

Disparity of indexing vocabularies a theory that does not necessitate the construction of a uniform indexing vocabulary, so the model can be applied to multimedia, distributed, and multilingual structured documents.

Aggregation of indexing a theory that supports an appropriate *aggregation of indexing*, which takes into account the fact that indexing elements can have the same or related meaning (e.g., "mammal" vs 'dolphin"). In this paper, we say that the indexing elements are *informationally related*. For instance:

- consider an object composed of two component objects: an image object indexed by the colour "blue", and a text object indexed by the term "sky". The two elements may be based on different indexing vocabularies, but they are also informationally related because often the colour of the sky is blue.

- the indexing vocabulary in a site A may be more refined than that of a site B. For instance, in site A every document dealing with animal is indexed by the element "animal", whereas in site B, the elements "cat", "dog", and "horse" can also be used as indexing elements.
- the term "mer" in a French document is related to the term "Meer" in a German document ("mer" and "Meer" are, respectively, the French and German words for "sea").

Aggregation of uncertainty a theory that allows for the appropriate *aggregation of uncertainty*. The uncertainty of the representation of a composite object must take into consideration the uncertainty of the representations of its component objects. The use of an element in indexing the composite object should be less uncertain if that element appears, maybe in different forms, in the representations of several of its component objects, than that of an element that appears in the representation of only one of its component objects.

The theory of **evidential reasoning** developed by Ruspini [10, 11, 12, 13] fulfils these requirements and, as it is a generalisation of the Dempster-Shafer theory of evidence, it possesses similar characteristics that from our experience are effective in modelling the representation and the retrieval of structured documents, and in efficiently implementing such a modelling.

In evidential reasoning, propositional logic is extended with epistemic operators to represent the knowledge held by an agent, and uncertainty is expressed on a probabilistic basis. Here an agent corresponds to the indexing method associated with the representation of components of a given **type**. Based on the representation, the theory allows to combine the knowledge held by a number of agents. The combination is used to define the aggregation of representations.

In this paper, we use evidential reasoning to build a general model for heterogeneous structured documents. The proposed model encompasses the following cases:

- a structured document composed of objects of the same media, located on the same site, and written in the same language, thus including our previous model based on the Dempster-Shafer theory of evidence;
- a structured document composed of objects of different media, located on several sites, or written in various languages, thus involving a number of indexing vocabularies.

The paper is organised as follows. In Section 2, we model the indexing vocabularies associated with document components. In Section 3, we model the representation of leaf document components. In Section 4, we model the representation of composite document components. The modellings of the representations of leaf and composite components use the same ontological concepts. What differs is how these concepts are obtained. For a composite object, they are the outcome of an *aggregation* operation performed on the representations of its component objects. In Section 5, we discuss an important property of the aggregation operation which ensures focussed retrieval. The retrieval process is presented in Section 6. Related work is described in Section 7. We conclude in Section 8. An overview of the notations used in the paper is given in an appendix.

We do not describe evidential reasoning itself, but we use it to express our model. For details of the theory, the reader should refer to [10, 11]. In our paper, some of the ontology original to evidential reasoning (e.g., epistemic states, epistemic algebra, marginal epistemic algebra, and so forth) is replaced by one closer to IR ontology. In addition, some of the concepts introduced by Ruspini, aimed at explaining the

foundations of evidential reasoning, are not included as they are not necessary to express our model.

2 Modelling indexing vocabularies

The objects of which heterogeneous structured documents are composed have each a *type*. We introduce two sets: $\mathcal{O}$ is the set of objects of which heterogeneous structured documents are composed; $\mathcal{T}$ is the set of types (media, sites, languages and any combination of these), e.g., text in the Dortmund site, image, speech, text in French in the Montreal site.

The type of an object is modelled with a function relating each object to its type.

Definition 2.1 (Type of an object) *The function* $type : \mathcal{O} \mapsto \wp(\mathcal{T})$ *maps each object* $o \in O$ *to a set of types* $t \subseteq \mathcal{T}$.

An object $o \in \mathcal{O}$ is said to be *monotype* if $type(o)$ is a singleton set; otherwise, it is said to be *multitype*. An object is monotype if it is a leaf object, or if it is composed of objects of the same type. An object is multitype if it is composed of objects of different types. Therefore, leaf objects are monotype, whereas composite objects can be either monotype or multitype.

The type of a composite object is defined as the *aggregation* of the types of its component objects.

Definition 2.2 (Aggregation of types) *Let* o, o_1 *and* o_2 *be objects of* $\mathcal{O}$ *such that* o *is composed of* o_1 *and* o_2. *The type of* o *is determined as:*

$$type(o) = type(o_1) \cup type(o_2)$$

Note that o is monotype if $type(o_1) = type(o_2)$, and o_1 and o_2 are monotype objects.

We chose to not use a type system, but a pure set union. The reason is that, in this work, it is sufficient to know that an object is composite, and what are the types of its components objects. For instance, an object composed of an image (type $\{Image\}$) and a piece of text (type $\{Text\}$) will have a type expressed as the set $\{Image, Text\}$. Since this is not a singleton set, the object is multitype.

To represent the content of an object, an indexing vocabulary associated with the type of the object is used. In this section, we describe how indexing vocabularies are modelled. This is done in two steps. First, we symbolise the **indexing vocabulary** associated with a type. Then, we define the **aggregation** of indexing vocabularies. The latter determines the indexing vocabulary used to represent composite objects.

2.1 Symbolising of an indexing vocabulary

An *indexing vocabulary* is associated with a type. Objects of a given type are represented by elements of the corresponding indexing vocabulary. The elements can be keywords, phrases, sentences, concepts derived from histograms, phonemes extracted from speech documents, etc., depending on the type.

The indexing vocabulary associated with a type is symbolised by a **syntax** and a **semantics**.

2.1.1 Syntax

The syntax is defined upon a *proposition space* and a *sentence space*.

Definition 2.3 (Proposition space) *For a type $t \subseteq \mathcal{T}$, $P_t = \{p_1, \ldots, p_n\}$ is the set of propositions symbolising the indexing vocabulary associated with t.*

For a type t, each element of the indexing vocabulary is symbolised by a proposition of P_t. For example, the term "wine" is symbolised by the proposition $wine \in P_{\{EnglishText\}}$, whereas the fact "the colour of the background is blue" is symbolised by a proposition $feature(colour, background, blue) \in P_{\{GlasgowImage\}}$, where $EnglishText, GlasgowImage \in \mathcal{T}$ ($EnglishText$ is the type text written in English, and $GlasgowImage$ is the type image in the Glasgow site).

The propositions symbolise elements of the indexing vocabulary. An object content can be described by individual elements (the object is about "wine"), the combination of individual elements (the object is about "wine and/or salmon"), or by stating that it is not about an individual element or a combination of them (the object is not about "wine"). All the possible (allowed) descriptions constitute a *sentence space* defined upon conjunction, disjunction, and negation of propositions.

Definition 2.4 (Sentence space) *Given a proposition space P_t associated with a type $t \subseteq \mathcal{T}$, the set of sentences that can be used to describe the content of an object of type t constitutes a* sentence space *S_t defined as follows:*

(1) the true *and* false *sentences denoted, respectively, $\top$ and $\bot$ are sentences of S_t;*

(2) any proposition p_i in P_t is a sentence of S_t;

(3) if ϕ and ψ are sentences of S_t, then so are $\phi \vee \psi$, $\phi \wedge \psi$, and $\neg\phi$.

For example, let "wine" and "salmon" be two elements of the indexing vocabulary symbolised, respectively, by the propositions $wine \in P_{\{EnglishText\}}$ and $salmon \in P_{\{EnglishText\}}$. The sentence $wine \wedge salmon \in S_{\{EnglishText\}}$ can be used to express that an object is about both "wine and salmon".

2.1.2 Semantics

The semantics of the indexing vocabulary is expressed with a possible worlds approach [14, 15].

Definition 2.5 (Type structure) *For each type $t \subseteq \mathcal{T}$, we define a* type structure *$F_t = \langle S_t, W_t, v_t, \pi_t \rangle$ where:*

(1) S_t is the sentence space associated with the type t.

(2) W_t is the set of possible worlds associated with the sentence space S_t.

(3) $v_t : W_t \times P_t \mapsto \{true, false\}$ where true and false are truth values.

(4) $\pi_t : W_t \times S_t \mapsto \{true, false\}$ is defined as follows. For all $w \in W_t$,

- *(a) For all $p \in P_t$, $\pi_t(w, p) = v_t(w, p)$;*
- *(b) $\pi_t(w, \top) = true$ and $\pi_t(w, \bot) = false$;*
- *(c) For all $\phi, \psi \in S_t$,*
  - *$\pi_t(w, \phi \wedge \psi) = true$ if and only if (iff) $\pi_t(w, \phi) = true$ and $\pi_t(w, \psi) = true$;*
  - *$\pi_t(w, \phi \vee \psi) = true$ iff $\pi_t(w, \phi) = true$ or $\pi_t(w, \psi) = true$;*
  - *$\pi_t(w, \neg\phi) = true$ iff $\pi_t(w, \phi) = false$.*

For simplicity, for any world $w \in W_t$ and sentence $\phi \in S_t$, if $\pi_t(w, \phi) = true$ (respectively, $\pi_t(w, \phi) = false$) we say that ϕ is true (respectively, false) in w.

The mapping v_t assigns truth values to propositions in a given world, whereas the mapping π_t assigns truth values to sentences (including propositions) in a given world. The truth values for v_t are constructed, whereas the truth values for π_t are dependent on those given by v_t.

The construction of v_t depends on whether the type t is an *aggregated type* (the type of a composite object) or not. For a non-aggregated type t, v_t is constructed from the proposition space P_t. Given a proposition space P_t, there is a maximum of $2^{|P_t|}$ possible worlds: one in which all the p_is are true, one in which $p_2, \ldots, p_n$ are true and $\neg p_1$ is true, etc. v_t reflects all these cases. In practice, the number of worlds in W_t can be smaller than $2^{|P_t|}$ because some propositions of P_t can be *informationally incompatible* with other propositions of P_t (e.g., no document about "wine" is about "computing", and vice versa).

A proposition (or a sentence) is informationally compatible with another one if either (1) they are not informationally related to each other (see Definition 2.9) or, (2) one is not informationally related to the negation of the other.

Two examples illustrating the construction of v_t are given in Sections 2.1.3 and 2.2.3.

For an aggregated type, v_t is constructed from an *aggregation* operation applied on type structures. This is described in Section 2.2.

With a possible worlds structure, the truth values of sentences can be related to each other. This is formalised with the notion of *logical implication* and *logical equivalence*.

Definition 2.6 (Logical implication) *For two sentences $\phi, \psi \in S_t$, the sentence ψ* logically implies *the sentence ϕ, denoted $\psi \Rightarrow \phi$, iff: for all possible worlds $w \in W_t$, if ψ is true in w, then ϕ is also true in w.*

For example, for $\phi, \psi \in S_t$, it can be proven that $\phi \wedge \psi \Rightarrow \psi$ (from standard propositional logic [16]).

Definition 2.7 (Logical equivalence) *For two sentences $\phi, \psi \in S_t$, the two sentences ψ and ϕ are* logically equivalent, *written $\psi \Leftrightarrow \phi$ iff $\psi \Rightarrow \phi$ and $\phi \Rightarrow \psi$.*

For example, for $\phi, \psi \in S_t$, it can be proven that $\phi \wedge \psi \Leftrightarrow \neg(\neg\psi \vee \neg\phi)$ (de Morgan's law [16]).

2.1.3 Example

Let $t_A \subseteq \mathcal{T}$ and $P_{t_A} = \{a, b, c\}$ be its associated proposition space. This means that the indexing vocabulary for the type t_A includes three elements symbolised by the propositions a, b and c. Suppose that a, b and c are informationally compatible with each other. Consequently, W_{t_A} is composed of 8 ($= 2^3$) worlds $w_1^A, \ldots, w_8^A$ listed in Table 1. The truth values of the propositions in P_{t_A} in the worlds (as given by the mapping v_{t_A}) are also displayed in the table.

From v_{t_A}, we derive π_{t_A}. For instance, $\pi_{t_A}(w_1^A, a \wedge b) = \pi_{t_A}(w_1^A, b \wedge c) = \pi_{t_A}(w_1^A, c) = true$, meaning that $a \wedge b$, $b \wedge c$ and c are true in world w_1^A; $\pi_{t_A}(w_2^A, c) = \pi_{t_A}(w_2^A, b \wedge c) = false$; etc.

Note that for all worlds $w^A \in W_{t_A}$, $\pi_{t_A}(w^A, \top) = true$ and $\pi_{t_A}(w^A, \bot) = false$; that is, the sentence $\top$ is true in all worlds and the sentence $\bot$ is false in all worlds.

Also, we can easily show for instance that $a \wedge \neg a \Leftrightarrow \bot$, $a \wedge b \Rightarrow a \vee b$, $c \Rightarrow \top$, etc.

To recap, an object has a type $t \subseteq \mathcal{T}$. The indexing vocabulary associated with the type is modelled by a sentence space S_t (the syntax) and a type structure F_t (the semantics).

worlds in W_{t_A}	$v_{t_A}(w_i^A, a)$	$v_{t_A}(w_i^A, b)$	$v_{t_A}(w_i^A, c)$
w_1^A	$true$	$true$	$true$
w_2^A	$true$	$true$	$false$
w_3^A	$true$	$false$	$true$
w_4^A	$true$	$false$	$false$
w_5^A	$false$	$true$	$true$
w_6^A	$false$	$true$	$false$
w_7^A	$false$	$false$	$true$
w_8^A	$false$	$false$	$false$

Table 1: Formalisation of the indexing vocabulary for a type t_A

For the indexing vocabulary of a leaf object, F_t is built directly from the propositions of P_t which symbolise the elements forming the indexing vocabulary. The example given in this section illustrated how this can be accomplished. For the indexing vocabulary of a composite object, F_t is built from the **aggregation** of type structures. This is defined in the next section.

2.2 Aggregation of indexing vocabularies

A composite object has a type built upon the types of its components objects. Let $t \subseteq \mathcal{T}$ be the type of an object that is composed of an object of type $t_A \subseteq \mathcal{T}$ and an object of type $t_B \subseteq \mathcal{T}$. From Definition 2.2, $t = t_A \cup t_B$. We refer to t as an *aggregated type*.

The indexing vocabulary of the aggregated type t is constructed upon the indexing vocabularies of the types t_A and t_B. The construction is an **aggregation of indexing vocabularies**. Formally, this is modelled as the *aggregation of type structures*, which yields the type structure modelling the indexing vocabulary associated with the aggregated type t. The aggregation of type structures is defined **syntactically** and **semantically**.

2.2.1 Syntax

The syntax of the indexing vocabulary of the aggregated type is defined upon a sentence space. Let P_{t_A} and P_{t_B} be the proposition spaces associated with t_A and t_B. Let S_{t_A} and S_{t_B} be the respective sentence spaces.

Definition 2.8 (Sentence space of an aggregated type) *Let $t, t_A, t_B \subseteq \mathcal{T}$. The sentence space S_t of an aggregated type $t = t_A \cup t_B$ is defined as follows:*

(1) if ϕ is a sentence of S_{t_A}, then it is a sentence of S_t.

(2) if ϕ is a sentence of S_{t_B}, then it is a sentence of S_t.

(3) clause (3) of Definition 2.4 defining well-formed sentences over S_t.

Clauses (1) and (2) mean that any sentence that can be used to index objects of type t_A or objects of type t_B can also be used to index composite objects of type $t = t_A \cup t_B$. Note that if $t_A = t_B$, then $S_{t_A} = S_{t_B} = S_t$.

2.2.2 Semantics

The semantics of the indexing vocabulary of the aggregated type $t = t_A \cup t_B$ is defined by a type structure constructed upon the type structures associated with

the types t_A and t_B. Let $F_t = \langle S_t, W_t, v_t, \pi_t \rangle$ be this type structure. The sentence space S_t was defined in the previous section. The mapping π_t is defined directly from the mapping v_t as given in Definition 2.5, clause (4). What remains to be determined are the set of possible worlds W_t and the mapping v_t.

The possible worlds forming W_t depend on the pairwise *compatible* combination of the possible worlds forming, respectively, W_{t_A} and W_{t_B} (the sets of possible worlds for the type structures associated with t_A and t_B, respectively). The general idea is as follows. For any pair of worlds $(w^A, w^B) \in W_{t_A} \times W_{t_B}$ that are compatible, a world w is created. Compatibility means that the two worlds w^A and w^B assign the same truth values to propositions of P_{t_A} and P_{t_B}, respectively, that are *informationally related*[1]: they are the same, they depict the same concept (e.g., synonym), or one depicts a more specific or general concept than the other (e.g., hyponym or hypernym). The relationship between the pair (w^A, w^B) and w is expressed by a function $\oplus$. That is, for any two worlds w^A and w^B, respectively, in W_{t_A} and W_{t_B}, if the propositions in P_{t_A} and P_{t_B} that are informationally related have the same truth value in, respectively, w^A and w^B, then a world w is created such that $\oplus(w^A, w^B) = w$. This is formally defined as follows.

Definition 2.9 (Construction of W_t) *For any world $w^A \in W_{t_A}$ and $w^B \in W_{t_B}$, if for all $p^A \in P_{t_A}$ and $p^B \in P_{t_B}$ either*

- *p^A and p^B are informationally related (e.g., they are equal, they are informationally equivalent, or one informationally implies the other) AND $v_{t_A}(w^A, p^A) = v_{t_B}(w^B, p^B)$, or*
- *p^A and p^B are not informationally related,*

then a world w is created.

The set of created worlds constitutes W_t.

The relation between w^A and w^B, and the created world w is expressed by the function $\oplus : W_{t_A} \times W_{t_B} \mapsto W_t$, where $\oplus(w^A, w^B) = w$.

Given two worlds $w^A \in W_{t_A}$ and $w^B \in W_{t_B}$, no world in W_t is created if there exist two propositions $p^A \in P_{t_A}$ and $p^B \in P_{t_B}$ that are informationally related, and for which $v_{t_A}(w^A, p^A) \neq v_{t_B}(w^B, p^B)$. Such a case means that the worlds w^A and W^B are "incompatible". In all other cases, a world in W_t is created.

To construct W_t, we need to know which propositions in P_{t_A} are informationally related to which propositions in P_{t_B}. This knowledge can be extracted from thesauri, dictionaries, more sophisticated knowledge bases, or can be determined manually (the worst case). Consider the case of the aggregation of an English and a French indexing vocabularies, both covering the topic "wine making". Let the two indexing vocabularies be modelled by the type structuresF_{t_A} and F_{t_B}, respectively. Let the propositions *wine* and *vin* be included in proposition spaces P_{t_A} and P_{t_B}, respectively. Consider two worlds w_1^A and w_2^A in W_{t_A} in which *wine* is, respectively, true and false, and world w^B in W_{t_B} in which *vin* is true. A language dictionary will provide the information that "wine" and "vin" refer to the same concept; the propositions *wine* and *vin* are informationally related (equivalent in this case). If we only consider the two propositions *wine* and *vin*, w_1^A and w^B are compatible, whereas w_2^A and w^B are not. As a result, a world w is created such that $w = \oplus(w_1^A, w^B)$. No world is created for the pair (w_2^A, w^B).

If the knowledge regarding informationally related propositions is not available, then all worlds of W_{t_A} are compatible with all worlds of W_{t_B}. The maximum number of worlds forming W_t that can be created is hence $|W_{t_A}| \times |W_{t_B}|$. However, this number can be smaller since some worlds in W_{t_A} may not be compatible with

[1] In the linguistic or ontological sense as given in a thesaurus, a language dictionary, etc.

some worlds in W_{t_B} (e.g., a world in W_{t_A} in which a is true and a world in W_{t_B} in which b is false, where $a \in P_{t_A}$ and $b \in P_{t_B}$ are informationally related). The availability of this knowledge therefore constrains the number of created worlds (the worlds forming W_t). In the remainder of this paper, we assume that the knowledge is available.

If $t_A = t_B$, then $W_{t_A} = W_{t_B}$. A world in W_{t_A} is related to itself. So for each world $w^A \in W_{t_A}$, a world $w \in W_t$ is created such that $\oplus(w^A, w^A) = w$. We have $|W_{t_A}| = |W_t|$[2].

We determine next how the mapping v_t is constructed. v_t assigns truth values to propositions from S_t in worlds of W_t. We define first the proposition space P_t (the domain of v_t). A proposition $p^A \in P_{t_A}$ can be informationally equivalent to a sentence of S_{t_B}, for instance, $p_1^B \wedge p_2^B$ where $p_1^B \in P_{t_B}$ and $p_2^B \in P_{t_B}$. In this case, p^A is a sentence, and not a proposition with respect to S_t, so p^A is not part of P_t. If p^A is informationally equivalent to a proposition of P_{t_B}, then p^A is part of P_t. Finally, if p^A is informationally equivalent to no sentences of S_{t_B}, then p^A is also part of P_t. To reflect the three cases, the proposition space P_t is defined as:

$$P_t = P'_{t_A} \cup P'_{t_B}$$

where
$P'_{t_A} = \{p^A \in P_{t_A} |$ there does not exist $\phi \in S_{t_B} - P_{t_B}$ such that p^A is informationally equivalent to $\phi\}$
and
$P'_{t_B} = \{p^B \in P_{t_B} |$ there does not exist $\phi \in S_{t_A} - P_{t_A}$ such that p^B is informationally equivalent to $\phi\}$.

The mapping v_t is now constructed as follow.

Definition 2.10 (Construction of v_t) *Let $v_t : P_t \mapsto \{true, false\}$. For any proposition $p \in P'_{t_A}$ or $p \in P'_{t_B}$ and world $w \in W_t$,*

$$v_t(w, p) = \begin{cases} v_{t_A}(w^A, p) & \text{if } p \in P'_{t_A}, \\ v_{t_B}(w^B, p) & \text{if } p \in P'_{t_B}. \end{cases}$$

where $w = \oplus(w^A, w^B)$.

Note that any proposition in P'_{t_A} or P'_{t_B} is a sentence of S_t (from Definition 2.8). v_t assigns a truth value to a proposition p in a world w that is the same as that of the proposition in w^A if $p \in P'_{t_A}$, and w^B if $p \in P'_{t_B}$. This means that, for $p \in P'_{t_A} \cap P'_{t_B}$, p cannot be true in w^A and false in w^B. This is ensured by the way the worlds in W_t are created (Definition 2.9). There cannot be a world in W_t in which two informationally related propositions have different truth values, where these truth values come from those assigned with respect to the worlds in W_{t_A} and W_{t_B}, respectively.

We can now define the aggregation of type structures leading to the type structure modelling the indexing vocabulary associated with the aggregated type t.

Definition 2.11 (Aggregation of type structures) *Let $t, t_A, t_B \subseteq \mathcal{T}$ and $t = t_A \cup t_B$, where the type structures for t_A and t_B are $F_{t_A} = \langle S_{t_A}, W_{t_A}, v_{t_A}, \pi_{t_A} \rangle$ and $F_{t_B} = \langle S_{t_B}, W_{t_B}, v_{t_B}, \pi_{t_B} \rangle$, respectively. The type structure $F_t = \langle S_t, W_t, v_t, \pi_t \rangle$ associated with t, is determined upon F_{t_A} and F_{t_B} as follows:*

[2] This is not completely correct since we have not made the assumption that the worlds must be distinguishable (two worlds cannot assign the same truth values to the propositions of the proposition space). A world $w^A \in W_{t_A}$ can be related to another world $w'^A \in W_{t_A}$. Therefore, there will be a second world created $w' \in W_t$ such that $\oplus(w^A, w'^A) = w'$. However, w and w' will not be distinguishable.

(1) S_t is defined as given in Definition 2.8;

(2) W_t is defined as given in Definition 2.9;

(3) v_t is defined as given in Definition 2.10;

(4) π_t is defined as given in Definition 2.5, clause (4).

Logical implication and logical equivalence, $\Rightarrow$ and $\Leftrightarrow$, also apply to the type structure F_t.

If $t_A = t_B$, we obtain $P'_{t_A} = P'_{t_B} = P_{t_A} = P_{t_B}$. Also, for $w \in W_t$, we must have $w^A \in W_{t_A}$ such that $\oplus(w^A, w^A) = w$. It can be easily shown that the truth value of any sentence of S_{t_A} in w^A is the same as that in w (we have $S_t = S_{t_A}$). Therefore, we obtain an identical type structure to F_{t_A}. This shows that the indexing vocabulary associated with an aggregated type based on two identical types is the same as that of the two types.

2.2.3 Example

Consider the proposition space $P_{t_B} = \{a, d\}$ for the type $t_B \subseteq \mathcal{T}$. This means that the indexing vocabulary associated to t_B includes two elements symbolised by the propositions a and d. The set of worlds W_{t_B} contains then $2^2 = 4$ worlds listed in Table 2. The truth values of the propositions of P_{t_B} in worlds of W_{t_B} are also displayed in the table (the mapping v_{t_B}).

worlds in W_{t_B}	$v_{t_B}(w_i^B, a)$	$v_{t_B}(w_i^B, d)$
w_1^B	$true$	$true$
w_2^B	$true$	$false$
w_3^B	$false$	$true$
w_4^B	$false$	$false$

Table 2: Formalisation of the indexing vocabulary for a type t_B

Compare this example to the proposition space P_{t_A} defined in Section 2.1.3. We have one common proposition, a. We assume that d is not informationally incompatible to a, b and c.

The mapping v_t constructed for the type structure F_t is given in Table 3. The propositions true in worlds of W_t are shown. The table also shows the worlds from W_{t_A} and W_{t_B} for which a world in W_t is created ($\oplus^{-1}(w_i)$[3]). In our case, $P_{t_A} = P'_{t_A}$ and $P_{t_B} = P'_{t_B}$.

In Table 3, for instance, w_1^A and w_1^B yield a world in W_t (w_1) because the truth value assigned to a is the same (a is the only proposition that belongs to both P'_{t_A} and P'_{t_B}) (this comes from Definition 2.9). From Definition 2.10, a, b, c and d are true in w_1 since a, b and c are true in w_1^A and a and d are true in w_1^B. w_1^A and w_3^B do not yield a world $w \in W_t$ such that $w = \oplus(w_1^A, w_3^B)$ because the truth values assigned to a in the two worlds w_1^A and w_3^B are different.

Note that the number of worlds in W_t is $16 < |W_{t_A}| \times |W_{t_B}| = 8 \times 4 = 32$.

2.3 Summary

In this section, we have presented the modelling of the indexing vocabulary associated with the type of leaf and composite objects. The syntax and the semantics were

[3] $\oplus^{-1}$ is the reverse mapping to $\oplus$: for all $w \in W_t$, $w^A \in W_{t_A}$ and $w^B \in W_{t_B}$, $\oplus^{-1}(w) = (w^A, w^B)$ iff $w = \oplus(w^A, w^B)$.

worlds w in W_t	$\oplus^{-1}(w_i)$	Propositions true in w_i	worlds w_i in W_t	$\oplus^{-1}(w_i)$	Propositions true in w_i
w_1	(w_1^A, w_1^B)	a, b, c, d	w_2	(w_2^A, w_1^B)	$a, b, \neg c, d$
w_3	(w_3^A, w_1^B)	$a, \neg b, c, d$	w_4	(w_4^A, w_1^B)	$a, \neg b, \neg c, d$
w_5	(w_1^A, w_2^B)	$a, b, c, \neg d$	w_6	(w_2^A, w_2^B)	$a, b, \neg c, \neg d$
w_7	(w_3^A, w_2^B)	$a, \neg b, c, \neg d$	w_8	(w_4^A, w_2^B)	$a, \neg b, \neg c, \neg d$
w_9	(w_5^A, w_3^B)	$\neg a, b, c, d$	w_{10}	(w_6^A, w_3^B)	$\neg a, b, \neg c, d$
w_{11}	(w_7^A, w_3^B)	$\neg a, \neg b, c, d$	w_{12}	(w_8^A, w_3^B)	$\neg a, \neg b, \neg c, d$
w_{13}	(w_5^A, w_4^B)	$\neg a, b, c, \neg d$	w_{14}	(w_6^A, w_4^B)	$\neg a, b, \neg c, \neg d$
w_{15}	(w_7^A, w_4^B)	$\neg a, \neg b, c, \neg d$	w_{16}	(w_8^A, w_4^B)	$\neg a, \neg b, \neg c, \neg d$

Table 3: Aggregation of type structures

given, and led to the definition of type structure. For a leaf object, the type structure is constructed from the elements of the indexing vocabulary. This was illustrated with two examples. For a composite object, the type structure is constructed as the aggregation of type structures (those modelling the indexing vocabularies of the components objects).

The modelling can be applied to multimedia, distributed, and multilingual documents, since the indexing vocabulary associated with an aggregated type t constructed upon two distinct types t_A and t_B is defined in terms of the indexing vocabularies associated with t_A and t_B. The modelling also applies to the restricted case of monomedia, non-distributed, and monolingual documents, since the indexing vocabulary associated with an aggregated type t constructed upon two identical types $t_A = t_B$ is the same as that associated with t_A (and t_B).

In the next section, we present the modelling of the representation of leaf objects and in the following one, the modelling of the representation of composite objects.

3 Modelling the representation of leaf objects

We describe the modelling of the representation of a leaf object $o \in \mathcal{O}$ where $type(o) = t \subseteq \mathcal{T}$ (t is a singleton set). There are two aspects to be modelled: the **indexing** and the **uncertainty of the indexing**.

3.1 Modelling the indexing

Modelling the *indexing* consists of modelling the elements of the indexing vocabulary representing the content of the object. This is defined by a **syntax** and a **semantics**, and is based on the type structure $F_t = \langle S_t, W_t, v_t, \pi_t \rangle$ modelling the indexing vocabulary associated with the type t.

3.1.1 Syntax

To model the content of an object, we need a way to express that some sentences of S_t play a role in the indexing of the object o: they *index* the object o. For this purpose, we introduce a **modal** operator I_o, and we extend the sentences space S_t to include *modal sentences*.

Modal operators allow us to distinguish which sentences of S_t index the objects. The truth values assigned to sentences in S_t only model the possible descriptions of the content of objects. Modal sentences state which of these possible descriptions are indeed descriptions of the content of objects.

Definition 3.1 (Modal space) *Let $o \in \mathcal{O}$ where $type(o) = t \subseteq \mathcal{T}$. The* modal space*, denoted S_o, associated with the object o is the set of sentences (well-defined formulae) defined as follows:*

(1) any sentence of S_t is also a sentence of S_o, where S_t is the sentence space associated with the type t;

(2) for $\phi \in S_t$, $\mathrm{I}_o\phi$ is a sentence of S_o.

Our definition does not consider sentences of the form $\mathrm{I}_o\mathrm{I}_o\phi$, and embedded modal sentences (e.g., $\phi \wedge \mathrm{I}_o\psi$). In our case, such sentences have no use. Therefore we use only a subset of evidential reasoning theory.

The sentences indexing a leaf object are derived from the output of the indexing process applied to the raw data of the object. We assume that such a set of sentences has been identified for each leaf object.

Definition 3.2 (Set of identified sentences) *We use the function $ID : Leaf(\mathcal{O}) \mapsto \wp(S_t)$ to represent the set of identified sentences, where $Leaf(\mathcal{O})$ is the set of leaf objects in $\mathcal{O}$.*

3.1.2 Semantics

The semantics of the indexing of a leaf object is expressed by a possible worlds approach. More precisely, it is defined upon the type structure associated with the type of the object, and the modal space defined in the previous section.

Intuitively, what we would like to obtain is the following. Every sentence ϕ in $ID(o)$ or that is logically implied by sentences of $ID(o)$ (the set $ID^{\wedge}(o)$ below) should index the object because it describes explicitly or implicitly the object content. The idea is that the modal sentence $\mathrm{I}_o\phi$ will be true (in some possible worlds). This is formally expressed in the following definition.

Definition 3.3 (Indexing structure) *Let $o \in \mathcal{O}$ where $type(o) = t \subseteq \mathcal{T}$. The indexing of the object o is modelled by an* indexing structure *$F_o = \langle F_t, S_o, \pi_o \rangle$ where:*

(1) $F_t = \langle W_t, S_t, v_t, \pi_t \rangle$ is the type structure modelling the indexing vocabulary associated with the type t;

(2) S_o is the modal space for object o;

(3) $\pi_o : W_t \times S_o \mapsto \{true, false\}$ where for $\phi \in S_t$ and $w \in W_t$:

- $\pi_o(w, \phi) = \pi_t(w, \phi)$;
- $\pi_o(w, \mathrm{I}_o\phi) = \pi_t(w, \phi)$ *if* $\phi \in ID(o)$;
- $\pi_o(w, \mathrm{I}_o\phi) = \pi_t(w, \psi)$ *if* $\psi \in ID^{\wedge}(o)$ *such that* $\psi \Rightarrow \phi$;
- $\pi_o(w, \mathrm{I}_o\phi) = false$ *in all other cases.*

where $ID^{\wedge} : Leaf(\mathcal{O}) \mapsto \wp(S_t)$ is the transitive And-closure *of the set of identified sentences given by the function ID (Definition 3.2). $ID^{\wedge}$ is defined as follows. For any leaf object $o \in Leaf(\mathcal{O})$:*

- $\top \in ID^{\wedge}(o)$.
- *if* $\phi \in ID(o)$, *then* $\phi \in ID^{\wedge}(o)$.
- *if* $\phi \in ID^{\wedge}(o)$ *and* $\phi' \in ID^{\wedge}(o)$, *and* ϕ *and* ϕ' *are not incompatible, then* $\phi \wedge \phi' \in ID^{\wedge}(o)$.

In the definition of π_o, we have four cases:

(1) the truth value of a non-modal sentence is that given by π_t;

(2) the second case applies to modal sentences. The sentence ψ is in $ID(o)$ (it is a sentence produced by the indexing process applied to the object). The truth value of $\phi = \mathrm{I}_o\psi$ in a world $w \in W_t$ is $true$ (respectively $false$) if the sentence ψ is $true$ (respectively $false$) in world w.

(3) the third case which also applies to modal sentences is more complex.

- first we define the set $ID^{\wedge}(o)$ which is the set of sentences that can be constructed as (the consistent) conjunction of sentences in $ID(o)$, the set $ID(o)$ itself, and $\top$.
- we look at all sentences logically implied by sentences of $ID^{\wedge}(o)$. Let φ be such a sentence where $\psi \Rightarrow \varphi$ for $\psi \in ID^{\wedge}(o)$. The truth value of $\phi = \mathrm{I}_o\varphi$ in a world w is that of ψ in w.

This case models that every sentence implied by a sentence of $ID(o)$ or the compatible conjunction of sentences of $ID(o)$ also indexes the object. For instance, if *wine* and *salmon* are in $ID(o)$, then the object is indexed by *wine*, *salmon* AND $wine \wedge salmon$. This is modelled by having in the worlds in which the sentences *wine* and *salmon* are true, the modal sentence $\mathrm{I}_o(wine \wedge salmon)$ true.

(4) we have false in all other cases.

We illustrate the above four cases with an example. Let the object o be of type t_A, where the indexing vocabulary associated with t_A is modelled by the type structure F_{t_A} defined in Section 2.1.3. Suppose that the outcome of the indexing algorithm for o yields $ID(o) = \{a, b \wedge \neg c\}$. Therefore:

$$a, b \wedge \neg c, a \wedge b \wedge \neg c, \top \in ID^{\wedge}(o)$$

We obtain for instance:

(1) Since $\pi_t(w_1^A, a \wedge b) = true$ (see Table 1), then:

$$\pi_o(w_1^A, a \wedge b) = true$$

(2) Since $\pi_t(w_2^A, b \wedge \neg c) = true$ and $b \wedge \neg c \in ID(o)$, then:

$$\pi_o(w_2^A, \mathrm{I}_o(b \wedge \neg c)) = true$$

Since $\pi_t(w_1^A, b \wedge \neg c) = false$, then:

$$\pi_o(w_1^A, \mathrm{I}_o(b \wedge \neg c)) = false$$

(3) Since $a \wedge b \wedge \neg c \in ID^{\wedge}(o)$, $\pi_t(w_2^A, a \wedge b \wedge \neg c) = true$, and $a \wedge b \wedge \neg c \Rightarrow a \wedge b$, then:

$$\pi_t(w_2^A, \mathrm{I}_o(a \wedge b)) = true$$

Since $b \wedge \neg c \in ID^{\wedge}(o)$, $\pi_t(w_3^A, b \wedge \neg c) = false$, and $b \wedge \neg c \Rightarrow \neg c$, then:

$$\pi_t(w_3^A, \mathrm{I}_o\neg c) = false$$

Finally, since $\top$ is in $ID^{\wedge}(o)$, and $\top \Rightarrow \top$, then $\mathrm{I}_o\top$ is true in all worlds of W_t.

(4) the sentence $\mathrm{I}_o\neg a$ does not correspond to any of the first three cases of π_o. So for all worlds $w^A \in W_{t_A}$, $\pi_o(w^A,\mathrm{I}_o\neg a) = false$.

The followings are axioms of evidential reasoning given in [10] that we reformulate in the context of this paper.

Definition 3.4 (Axioms) *Let $\phi, \psi \in S_t$ and $w \in W_t$:*

(A1) If $\mathrm{I}_o\phi$ is true in w, then so is ϕ;

(A2) If $\phi \Rightarrow \psi$ is true then so is $\mathrm{I}_o\phi \Rightarrow \mathrm{I}_o\psi$;

(A3) If ϕ is a tautology (sentence true in all worlds) then so is $\mathrm{I}_o\phi$.

Note that our definition of π_0 satisfies the above axioms. $\mathrm{I}_o\phi$ is true in a world only if ϕ is true in that world (the reverse does not hold) (Axiom A1). Axiom A2 is satisfied from the third case of the definition of π_o. Axiom A3 is satisfied because $\mathrm{I}_o\top$ is true in all worlds.

The definition of π_o enables us to differentiate between two leaf objects o and o' for which $ID(o) = \{a \wedge b\}$ and $ID(o') = \{a \wedge b, a\}$. The indexing process applied to o' produces two sentences, whereas, it produces one sentence when applied to object o. Although we have $a \wedge b \Rightarrow a$, the content of the two objects is different. For object o, there is no *explicit* evidence regarding a. The representations of the two objects must reflect this distinction. This means that, in at least one world of W_t, the truth values of some sentences as given by π_o must differ to their truth values as given by $\pi_{o'}$. Two different indexing structures modelling the indexing of the two objects should be obtained. We illustrate this with an example using the type structure t_A introduced in Section 2.1.3.

- For object o: $a \wedge b$ is true in worlds w_1^A and w_2^A (see Table 1). Since $a \wedge b \in ID(o)$, this means that $\mathrm{I}_o(a \wedge b)$ is true in w_1^A and w_2^A. Since $a \wedge b \Rightarrow a$ holds, and $a \wedge b \in ID^{\wedge}(o)$, then $\mathrm{I}_o a$ is true in worlds w_1^A and w_2^A. However, $\mathrm{I}_o a$ is not true in the other worlds because there is no $\phi \in ID^{\wedge}(o)$ such that ϕ is true in these worlds and $\phi \Rightarrow a$. In all these other worlds, only $\mathrm{I}_o\phi$ is true where ϕ is a tautology, including $\top$.

- For object o': as for the previous case, $\mathrm{I}_{o'}(a \wedge b)$ is true in w_1^A and w_2^A, and $\mathrm{I}_{o'} a$ is true w_1^A, w_2^A. In addition, $\mathrm{I}_{o'} a$ is true in w_3^A, w_4^A because $a \in ID(o')$. In all other worlds, only $\mathrm{I}_{o'}\phi$ is true where ϕ is a tautology, including $\top$.

3.1.3 Example

Suppose that for an object o_1 of type t_A, we have $ID(o_1) = \{a, b \wedge \neg c\}$, and that for an object o_2 of type t_B, we have, $ID(o_2) = \{\neg a, d\}$. Using the type structure F_{t_A} defined in Section 2.1.3 and the type structure F_{t_B} given in Section 2.2.3, the modal sentences true in the worlds W_{t_A} and W_{t_B} are as shown in Table 4. We only show those sentences obtained from the second case of the definition of π_o. The other true modal sentences can be derived by applying the third case of the definition of π_o.

Let us consider the world w_6^A. In this world, the sentences $\neg a, b$ and $\neg c$ are true. The sentence $\neg a \wedge b \wedge \neg c$ (or the corresponding world w_6^A) constitutes a possible description of the content of an object of type t_A (the sentence is true in world w_6^A). For object o_1 (of type t_A), we want to express that from the available evidence, the sentences $a, b \wedge \neg c$, any conjunction of these sentences, or every sentences logically implied by the previously mentioned sentences contribute to the description of the content of the object o_1. Since the sentence a is false in w_6^A, and the sentences b

worlds in W_{t_A}	$\mathrm{I}_{o_1}\phi$	worlds in W_{t_B}	$\mathrm{I}_{o_2}\phi$
w_1^A	$\mathrm{I}_{o_1}a$	w_1^B	$\mathrm{I}_{o_2}d$
w_2^A	I_{o_1}a, $\mathrm{I}_{o_1}(b \wedge \neg c)$	w_2^B	$\mathrm{I}_{o_2}\top$
w_2^A	$\mathrm{I}_{o_1}a$	w_3^B	$\mathrm{I}_{o_2}(\neg a)$,$\mathrm{I}_{o_2}d$
w_4^A	$\mathrm{I}_{o_1}a$	w_4^B	$\mathrm{I}_{o_2}(\neg a)$
w_5^A	$\mathrm{I}_{o_1}\top$		
w_6^A	$\mathrm{I}_{o_1}(b \wedge \neg c)$		
w_7^A	$\mathrm{I}_{o_1}\top$		
w_8^A	$\mathrm{I}_{o_1}\top$		

Table 4: Modal sentences true for object o_1 and object o_2

and $\neg c$ are true in w_6^A, only the evidence with respect to the last two sentences can be modelled. This is done by setting the truth values of $\mathrm{I}_{o_1}b$, $\mathrm{I}_{o_1}\neg c$ and $\mathrm{I}_{o_1}(b \wedge \neg c)$ in w_6^A to *true*. The truth values of other modal sentences are derived from the application of Definition 3.3 and the axioms given in Definition 3.4. This example shows the use of modal operators and the corresponding modal sentences.

For instance, for world $w_2^A \in W_{t_A}$, $\mathrm{I}_{o_1}a$ and $\mathrm{I}_{o_1}(b \wedge \neg c)$ are true because a and $b \wedge \neg c$ are true in w_2^A and $a, b \wedge \neg c \in ID(o_1)$. In world w_7^A, a and $b \wedge \neg c$ are false (the latter because b is false), so neither $\mathrm{I}_{o_1}a$ nor $\mathrm{I}_{o_1}(b \wedge \neg c)$ is true in w_7^A. Only $\top$ (and any tautology) is true in w_7.

We can also derive that $\mathrm{I}_{o_2}(\neg a \wedge d)$ is true in w_3^B. This is because $\neg a \wedge d \in ID^{\wedge}(o_2)$, $\neg a \wedge d \Rightarrow \neg a \wedge d$ and $\neg a \wedge d$ is true in w_3^B.

3.2 Modelling the uncertainty of the indexing

So far we have not mentioned the uncertainty inherent to the representation of objects. The uncertainty is modelled by assigning weights to sentences of S_t to reflect how well they describe the object content.

We describe first which sentences of S_t are weighted. These sentences, referred to as **weighted sentences**, are defined upon the set of sentences $ID^{\wedge}(o)$. Then, we model the **uncertainty** itself. For a leaf object, this means defining a function representing the weights. The weights are assumed to have been computed elsewhere. We give an example of how this can be done in practice.

3.2.1 Weighted sentences

Some sentences of S_t are assigned weights representing how accurate they are at describing the object content. First, we must take into account that sentences can be logically equivalent. When assigning weights to two sentences ϕ and ψ of S_t, if $\phi \Leftrightarrow \psi$, the weight should be the same, and it should be assigned once. We therefore partition the set S_t into sets of logically equivalent sentences. We obtain a group of equivalent classes, and only one sentence (the representative sentence) can be assigned a weight[4].

Definition 3.5 (Frame of discernment) *For a given type $t \in \mathcal{T}$, the set of equivalence classes with respect to $\Leftrightarrow$ is called a* frame of discernment, *and is denoted Φ_t.*

Therefore, only sentences forming Φ_t can be weighted. These sentences are determined upon the sentences forming $ID^{\wedge}(o)$ via the notion of *most specific sentence*.

[4]Such an approach is common in IR, for instance, where terms are stemmed into some base forms [17], or a thesaurus is used to group synonyms together.

Definition 3.6 (Most specific sentence) *A sentence $\psi \in \Phi_t$ is said to be the* most specific sentence *for a world $w \in W_t$ iff for every sentence $\phi \in \Phi_t$, $\mathrm{I}_o\phi$ is true in w iff $\psi \Rightarrow \phi$.*

The function $mss_o : W_t \mapsto \Phi_t$ yields the most specific sentence for a world in W_t, for the object o.

As we will see later in this section, weights are assigned to most specific sentences.

For any world $w \in W_t$, a most specific sentence always exists because it can always be constructed as the conjunction of all sentences $\psi_i \in \Phi_t$ such that $\mathrm{I}_o\psi_i$ is true in w (from Definition 3.3).

Theorem 3.1 *For any world $w \in W_t$, its most specific sentence is unique.*

Proof: Let $w \in W_t$. Suppose that w has two most specific sentences, ϕ and ψ in Φ_t (i.e., $\phi \not\Leftrightarrow \psi$). Hence, for every sentence $\varphi \in \Phi_t$, $\mathrm{I}_o\varphi$ is true in w iff $\phi \Rightarrow \varphi$ and $\psi \Rightarrow \varphi$. Since $\mathrm{I}_o\phi$ and $\mathrm{I}_o\psi$ are true in w (from being most specific sentences), then either $\phi \Rightarrow \psi$, $\psi \Rightarrow \phi$, or $\psi \Leftrightarrow \phi$. The latter case cannot arise since ϕ and ψ are in Φ_t (since $\phi \not\Leftrightarrow \psi$). In either of the first two cases, either ϕ or ψ is not a most specific sentence. Therefore the most specific sentence of a world is unique. □□

The most specific sentences are defined directly from the set $ID^\wedge(o)$ as shown in the following theorem.

Theorem 3.2 *For $w \in W_t$, $mss_o(w) = \phi$ where ϕ is the longest sentence (in terms of the number of conjuncts) in $ID^\wedge(o) \cap \Phi_t$ such that $\pi_o(w, \mathrm{I}_o\phi) = true$.*

We take the intersection of $ID^\wedge(o)$ and Φ_t because only sentences in Φ_t can be weighted.

Proof: Let $w \in W_t$. Let ϕ be the longest sentence $ID^\wedge(o) \cap \Phi_t$ such that $\pi_o(w, \mathrm{I}_o\phi) = true$. Therefore, for all sentences $\psi \in ID^\wedge(o) \cap \Phi_t$ such that $\pi_o(w, \mathrm{I}_o\psi) = true$, we have $\phi \Rightarrow \psi$. Therefore, ϕ is the most specific sentence for w. □□

For instance, for object o_1, $a \wedge b \wedge \neg c$ is the longest sentence in $ID^\wedge(o) \cap \Phi_t$ such that $\mathrm{I}_{o_1}(a \wedge b \wedge \neg c)$ is true in w_2^A (see Table 4). Therefore, $a \wedge b \wedge \neg c$ is the most specific sentence of w_2^A for object o_1; i.e. $mss_{o_1}(w_2^A) = a \wedge b \wedge \neg c$. We also obtain:

$mss_{o_1}(w_1^A) = mss_{o_1}(w_3^A) = mss_{o_1}(w_4^A) = a$

$mss_{o_1}(w_6^A) = b \wedge \neg c$

$mss_{o_1}(w_5^A) = mss_{o_1}(w_7^A) = mss_{o_1}(w_8^A) = \top$

In practice, the use of most specific sentences can be interpreted as follows. Suppose that the indexing process applied to an object produces two sentences *german* and *wine*. The sentences *german*, *wine*, and *german* $\wedge$ *wine* can be shown to be most specific for some worlds. This means that the object is about "german", "wine", or both. Weights will be assigned then to *german*, *wine*, and also *german* $\wedge$ *wine*. The object will be relevant to any query about "wine", "german", or both ("german and wine"). Suppose now that the indexing process produces only one sentence *german* $\wedge$ *wine*. In this case, *german* $\wedge$ *wine* will be a most specific sentence, but neither *german* or *wine* alone will. This means that the object is about "german and wine". In this case, a weight will be assigned to the sentence *german* $\wedge$ *wine*, but not to *german* nor *wine*. The object is relevant to query asking for "german and wine". The object is also relevant to a query about "german" or "wine" because *german* $\wedge$ *wine* $\Rightarrow$ *german* (a document about "german and wine" is also about "german") and *german* $\wedge$ *wine* $\Rightarrow$ *wine* (a document about "german and wine" is also about "wine"). However, the relevance is based on different evidence than with the previous case.

The most specific sentences are sentences built upon $ID(o)$ for which there is explicit evidence that they index the object. Weights will be assigned to them

to capture how well they describe the content of the object. All other sentences indexing the object are being so implicitly. No weight will be assigned to them.

In some cases, $\top$ is the most specific sentence for a world. This is because $\mathrm{I}_o\top$ is the only modal sentence (except for tautology) true in that world. This models *ignorance*, which is discussed in the next section.

We have defined the sentences to which weights are assigned. Next, we model the weights themselves.

3.2.2 Mass function

Weights are assigned to sentences to model the uncertainty of the indexing. Following our previous work [4], this is expressed by a *mass function* defined upon the indexing structure modelling the indexing of the object.

Definition 3.7 (Mass function) *Let $o \in \mathcal{O}$ whose indexing is modelled by the indexing structure $F_o = \langle F_t, S_o, \pi_o \rangle$. The uncertainty of the indexing is modelled by a* mass function $m_o : \Phi_t \mapsto [0,1]$ *such that:*

$$\sum_{\phi \in \Phi_t} m_o(\phi) = 1 \quad \textit{and} \quad m_o(\bot) = 0$$

m_o represents the uncertainty inherent in the representation of the content of the object o. For $\phi \in \Phi_t$, $m_o(\phi)$ is the belief based on explicit evidence that ϕ describes appropriately the object content. The higher $m_o(\phi)$, the higher the sentence ϕ is considered a good description of the object content. If $m_o(\phi) = 0$, then there is no explicit evidence that the object o is about about ϕ. There may be implicit evidence that the object is about ϕ. This happens if there exists a sentence $\psi \in \Phi_t$ such that $m_o(\psi) > o$ and $\psi \Rightarrow \phi$. This can be captured by the belief function associated with the mass function. Belief functions are used to express the relevance of objects to queries, and are discussed in Section 6.

The value $m_o(\top)$ can range from 0 to 1 and models *ignorance*, that is the extent to which we do not know what the object o is about; this can be viewed as the overall uncertainty of the indexing. The value $m_o(\top)$ is referred to as the *uncommitted belief*. We have two extreme cases, $m_o(\top) = 1$, expressing unknown (we do not know what the object is about), and $m_o(\top) = 0$, expressing complete knowledge (we know what the object is not about). The representation of ignorance in IR modelling was discussed in [9, 18, 19].

The sentences that will be weighted (for which m_o is non-null) are those forming the set $MSS(o)$ where $MSS : \mathcal{O} \mapsto \wp(\Phi_t)$ yields the set of most specific sentences associated to the modelling of the indexing of an object in $\mathcal{O}$. Formally:

$$MSS(o) = \bigcup \{\phi \in \Phi_t | \text{ there exists } w \in W_t \text{ such that } mss_o(w) = \phi\}$$

Therefore, for any $\phi \in MSS(o)$, we have $m_o(\phi) > 0$ such that

$$\sum_{\phi \in MSS(o)} m_o(\phi) = 1$$

It is easy to show that the above definition of m_0 leads to a mass function.

To recap, the most specific sentences are sentences indexing the object o based on explicit evidence. Their weights reflect how well they index the object o (they describe the object content).

3.2.3 Construction of the mass function

For a leaf object o, the mass function m_o is constructed from the output of the indexing process applied to the raw data of o. This can be done using standard IR weighting mechanisms but modified so that the mass function values add up to one (see [9, 18, 20]). We illustrate this with an example.

Continuing with our example, the modelling of object o_1 involves four most specific sentences:

$$MSS(o_1) = \{a, a \wedge b \wedge \neg c, b \wedge \neg c, \top\}$$

Suppose that the frequencies [21] of the two propositions a and $b \wedge \neg c$ are, respectively, x and y, and the uncertainty of the indexing of object o_1 is z (the uncommitted belief). Suppose also that frequency of the sentence $a \wedge b \wedge \neg c$ is xy (this value can be computed, for instance, from x and y as $x * y$ when the distributions are assumed independent). Then we can assign:

$m_o(a) = x/N$
$m_o(a \wedge b \wedge \neg c) = xy/N$
$m_o(b \wedge \neg c) = y/N$
$m_o(\top) = z/N$

where $N = x + y + xy + z$.

Note that modelling ignorance consists of assigning a weight to the true sentence $\top$.

Other estimations of x, y and xy can be done through more sophisticated $tf \times idf$ methods [21, 22], machine learning techniques [23] (thus including dependent distributions) or via subjective assignment by manual indexers [19]. Also an estimation of z can be done using residual belief (see [18, 24]). This, however, goes outside the scope of this paper.

3.2.4 Example

Table 5 shows an example of the mass functions modelling the uncertainty of the indexing for objects o_1 and o_2. The most specific sentences (the weighted sentences) are shown in the second and fifth columns for the objects o_1 and o_2, respectively. In the first and fourth columns, the worlds for which the sentences are most specific (for object o_1 and object o_2, respectively) are shown.

worlds in W_{t_A}	$MSS(o_1)$	m_{o_1}	worlds in W_{t_B}	$MSS(o_2)$	m_{o_2}
w_1^A, w_3^A, w_4^A	a	0.3	w_1^B	d	0.2
w_2^A	$a \wedge b \wedge \neg c$	0.4	w_2^B	$\top$	0.3
w_5^A, w_7^A, w_8^A	$\top$	0.1	w_3^B	$\neg a \wedge d$	0.4
w_6^A	$b \wedge \neg c$	0.2	w_4^B	$\neg a$	0.1
$\sum_{\phi \in MSS(o_1)} m_{o_1}(\phi)$		1	$\sum_{\phi \in MSS(o_2)} m_{o_2}(\phi)$		1

Table 5: Mass functions for objects o_1 and o_2

3.3 Summary

In this section, we presented the modelling of the representation of a leaf object: the sentences indexing the object and the uncertainty associated with the indexing. We have shown how the representation is formally expressed, first upon the indexing structure formally modelling the indexing of the object, and then the mass function formally capturing the uncertainty of the indexing.

Next, we present how the content of a composite object is determined as an aggregation operation performed on the representation of its component objects.

4 Modelling the representation of a composite object: the aggregation

As for leaf objects, the content of a composite object is modelled by an indexing structure and a mass function. For a leaf object, the indexing structure and the mass function are constructed from the outcome of the indexing process, whereas, for a composite object, they are constructed from the **aggregation** of the indexing structures and the mass functions modelling the representation of its component objects.

We present the modelling of the representation of a composite object in two parts. First, we describe the **aggregation of indexing structures** which yields the indexing structure modelling the indexing of the composite object. Then, we present the **aggregation of the mass functions** which yields the mass function formalising the uncertainty of the indexing of the composite object.

4.1 Aggregation of the indexing

Let the object $o \in \mathcal{O}$ be composed of objects o_1 and o_2 where $o_1, o_2 \in \mathcal{O}$ and $type(o_1) = t_A$ and $type(o_2) = t_B$ for $t_A, t_B \subseteq \mathcal{T}$. Let $t \subseteq \mathcal{T}$ be the type of the object o ($type(o) = t$). From Definition 2.2 of an aggregated type (see Section 2) $t = t_A \cup t_B$.

Let the type structures for t_A and t_B be, respectively, $F_{t_A} = \langle S_{t_A}, W_{t_A}, v_{t_A}, \pi_{t_A} \rangle$ and $F_{t_B} = \langle S_{t_B}, W_{t_B}, v_{t_B}, \pi_{t_B} \rangle$. The type structure $F_t = \langle S_t, W_t, v_t, \pi_t \rangle$ associated with t is given by Definition 2.11 of the aggregation of type structures (see Section 2.2).

The indexing structure modelling the indexing of the composite object o is defined **syntactically** and **semantically**.

4.1.1 Syntax

Let the indexing structures for o_1 and o_2 be, respectively, $F_{o_1} = \langle F_{t_A}, S_{o_1}, \pi_{o_1} \rangle$ and $F_{o_2} = \langle F_{t_B}, S_{o_2}, \pi_{o_2} \rangle$. We denote I_{o_1} and I_{o_2} the modal operators associated with objects o_1 and o_2, respectively.

Let $F_o = \langle F_t, S_o, \pi_o \rangle$ be the indexing structure modelling the indexing of the composite object o. As for a leaf object, F_o is defined syntactically upon a modal space denoted S_o and its modal operator denoted I_o. The definition of the modal space S_o is identical to that of a leaf object (Definition3.1):

(1) any sentence of S_t is also a sentence of S_o; and

(2) for $\phi \in S_t$, $\mathrm{I}_o\phi$ is a sentence of S_o.

4.1.2 Semantics

To give the semantics of the indexing structure $F_o = \langle F_t, S_o, \pi_o \rangle$, we must construct the mapping π_o. The construction of π_o reflects that the sentences indexing a composite object are based on those indexing its component objects.

Definition 4.1 (Construction of π_o) *Let $\pi_o : W_t \times S_o \mapsto \{true, false\}$. For $\phi \in S_o$ and $w \in W_t$:*

$$\pi_o(w,\phi) = \begin{cases} \pi_t(w,\phi) & \text{if } \phi \in S_t; \\ true & \text{if } \phi = \mathrm{I}_o\psi \text{ and there exist } \psi_1 \in S_{t_A} \text{ and } \psi_2 \in S_{t_B}, \\ & \text{respectively, such that } \mathrm{I}_{o_1}\psi_1 \text{ is true in } w^A \text{ and} \\ & \mathrm{I}_{o_2}\psi_2 \text{ is true in } w^B \text{ for } \oplus(w^A, w^B) = w \text{ and} \\ & \psi_1 \wedge \psi_2 \Rightarrow \psi; \\ false & \text{otherwise.} \end{cases}$$

The truth value of a non-modal sentence (a sentence in S_t) is given by the function π_t (as for the modelling of the indexing of leaf objects).

A modal sentence $\mathrm{I}_o\psi$ is true in a world w if there exist two sentences ψ_1 and ψ_2, respectively, in S_{t_A} and S_{t_B} such that they index, respectively, objects o_1 and o_2 AND their conjunction implies ψ. The worlds, respectively, in W_{t_A} and W_{t_B} in which the sentences ψ_1 and ψ_2 are true must be linked to w via $\oplus$.

For example, suppose that the object o_1 is indexed by *wine* and the object o_2 is indexed by *chardonnay*. The composite object will be indexed by any sentence logically implied by $wine \wedge chardonnay$, including $wine \wedge chardonnay$.

Definition 4.1 also implies that if a component object is indexed by sentence φ, then φ also indexes the composite object. This is because $\mathrm{I}_{o_i}\top$ is true in all worlds (of W_{t_A} and W_{t_B}). So for any sentence $\psi_1 \in S_{t_A}$ and $\psi_2 \in S_{t_B}$, we have $\psi_1 \wedge \top \Rightarrow \psi_1$ and $\top \wedge \psi_2 \Rightarrow \psi_2$. Therefore, if a component object is indexed by *wine*, the composite object is also indexed by *wine*. This also comes from the fact that $\mathrm{I}_o(wine \wedge chardonnay) \Rightarrow \mathrm{I}_o wine$ (Axiom A2).

We can give now the formal definition of the aggregation of indexing structures.

Definition 4.2 (Aggregation of indexing structures) *Let $o \in \mathcal{O}$ where $type(o) = t \subseteq \mathcal{T}$. The indexing structure $F_o = \langle F_t, S_o, \pi_o \rangle$ modelling the indexing of the composite object o is defined as follows:*

(1) F_t is as given in Definition 2.11;

(2) S_o is as given in Definition 3.1;

(3) π_o is as given in Definition 4.1

4.1.3 Example

Applied to our example, the truth values assigned by π_o to modal sentences (sentences indexing the composite object o) in the worlds forming W_t (given in Table 3) are shown in Table 6, third column. The worlds $w_j^A \in W_{t_A}$ and $w_k^B \in W_{t_B}$ such that $\oplus(w_j^A, w_k^B) = w_i$ for $w_i \in W_t$ are shown in the second column. The modal sentences true in w_j^A and w_k^B leading to the modal sentences true in w_i are shown in the fourth and fifth columns. We only show the modal sentences that cannot be derived from the logical implication in, respectively, W_t, W_{t_A} and W_{t_B}.

For instance, we have $\mathrm{I}_{o_1}a$ and $\mathrm{I}_{o_1}(b \wedge \neg c)$ true in w_2^A, and $\mathrm{I}_{o_2}d$ true in w_1^B. Therefore $\mathrm{I}_o(a \wedge d)$ and $\mathrm{I}_o(b \wedge \neg c \wedge d)$ are true in w_2.

We have $\mathrm{I}_{o_1}a$ true in w_4^A and $\mathrm{I}_{o_2}\top$ true in w_2^B so $\mathrm{I}_o a$ is true in w_8.

We have seen how to construct the indexing structure modelling the indexing of a composite object as an aggregation operation performed on the indexing structures modelling the indexing of its component objects. The construction was general, so it captures the cases where the indexing vocabularies of the component objects are identical or different. Therefore, we can determine the indexing of a composite

world w_i in W_t	(w_j^A, w_k^B)	Modal sentences true in w_i	Modal sentences true in w_j^A	Modal sentences true in w_k^B
w_1	(w_1^A, w_1^B)	$a \wedge d$	a	d
w_2	(w_2^A, w_1^B)	$a \wedge d, b \wedge \neg c \wedge d$	$a, b \wedge \neg c$	d
w_3	(w_3^A, w_1^B)	$a \wedge d$	a	d
w_4	(w_4^A, w_1^B)	$a \wedge d$	a	d
w_5	(w_1^A, w_2^B)	a	a	$\top$
w_6	(w_2^A, w_2^B)	$a, b \wedge \neg c$	$a, b \wedge \neg c$	$\top$
w_7	(w_3^A, w_2^B)	a	a	$\top$
w_8	(w_4^A, w_2^B)	a	a	$\top$
w_9	(w_5^A, w_3^B)	$b \wedge \neg c \wedge \neg a, b \wedge \neg c \wedge d$	$b \wedge \neg c$	$\neg a, d$
w_{10}	(w_6^A, w_3^B)	$\neg a, d$	$\top$	$\neg a, d$
w_{11}	(w_7^A, w_3^B)	$\neg a, d$	$\top$	$\neg a, d$
w_{12}	(w_8^A, w_3^B)	$\neg a, d$	$\top$	$\neg a, d$
w_{13}	(w_5^A, w_4^B)	$\neg a$	$\top$	$\neg a$
w_{14}	(w_6^A, w_4^B)	$b \wedge \neg c \wedge \neg a$	$b \wedge \neg c$	$\neg a$
w_{15}	(w_7^A, w_4^B)	$\neg a$	$\top$	$\neg a$
w_{16}	(w_8^A, w_4^B)	$\neg a$	$\top$	$\neg a$

Table 6: Example of the aggregation of indexing structures

object whether its component objects *are or are not* of the same type (medium, site, or language). In the model developed in [4], only the first option was allowed. The aggregation is also defined such that the informational relatedness of the indexing vocabularies, and hence the elements indexing the objects, can be taken into account.

Next, we present how the uncertainty of the indexing of the composite object is determined.

4.2 Aggregation of the uncertainty

Modelling the uncertainty of the indexing of a composite object consists of computing the mass function for the composite object. We recall that the mass function expresses, using weights, how the elements of the indexing vocabulary appropriately describe the content of (index) an object.

In [4], the mass function of the composite object o is defined as the aggregation of the mass functions of the components objects o_1 and o_2, as given by the Dempster's combination rule. The combination rule was both effective and efficient in determining the representation of the composite object. As discussed in the introduction of this paper, the rule can, however, only be applied if a uniform indexing vocabulary is defined for all component objects. The combination rule as provided by evidential reasoning is more general, because it can apply to objects indexed by elements from different indexing vocabularies, that is objects of different types (medium, site, or language).

In [10], a mass function is defined upon a probability function defined on an algebra defined on the set of possible worlds. Let m_o be the mass function for a composite object o. m_o is defined in terms of a probability function $\hat{\Pr}_o$, which is itself defined in terms of second probability function $\Pr_o$. $\Pr_o$ is the probability function representing the uncertainty that objects o_1 and o_2 are indexed by a sentence in Φ_{t_A} and a sentence in Φ_{t_B}, respectively. In other words, $\Pr_o$ expresses the uncertainty prior to the aggregation, whereas $\hat{\Pr}_o$ represents the uncertainty after aggregation. It is the result of constraining probabilistic knowledge in $W_{t_A} \times W_{t_B}$ to

those worlds that are possible after the aggregation (for details, the reader should refer to [10]).

In this work, the representations of objects o_1 and o_2 are independent. That is, the indexing process applied to object o_1 is done independently of that applied to object o_2. Such a scenario will be mostly the case for heterogeneous structured documents[5]. If the representations of the two objects are not independent, then a more complex formulation should be used to aggregate the uncertainty of the two representations. This formulation can be found in [10]. In previous work, we used the Dempster-Shafer's theory of evidence (a special case of evidential reasoning) to aggregate objects representations. We have carried out two sets of experiments, one using standard test collections [9], and one using web documents [25]. In both, assuming independence did not seem to degrade retrieval effectiveness. Nevertheless, in future work, we will investigate at both theoretical and experimental levels, dependent objects representation (in particular to dead with video data).

With the independence assumption, the technical details describing the construction of $\hat{\Pr}_o$ which yields m_o in terms of $\Pr_o$ are not necessary for the understanding of this paper, and hence are omitted. Furthermore, the calculation of the mass function associated with the composite object is straightforward.

However, it should be noted that evidential reasoning as developed in [10] allows for the dependent case to be taken into account.

As for leaf object, first we determine the **weighted sentences**, the sentences that are assigned weights, then the **mass function** itself.

4.2.1 Weighted sentences

As for leaf object, weighted sentences correspond to most specific sentences. The definition of a most specific sentence in the worlds forming W_t is the same as that for leaf objects (see Definition 3.6).

We give next a theorem that relates most specific sentences of a composite object to those of its component objects. Since we are assigning weights, we work with the frame of discernment Φ_t.

Theorem 4.1 *A sentence $\phi \in \Phi_t$ is the most specific sentence for world $w \in W_t$ iff there exist $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$, each most specific sentence for world $w^A \in W_{t_A}$ and $w^B \in W_{t_B}$, respectively, where $w = \oplus(w^A, w^B)$ and $\phi^A \wedge \phi^B \Leftrightarrow \phi$.*

Proof: Let $\phi \in \Phi_t$ be the most specific sentence for world $w \in W_t$. We show that there exist $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$, each most specific sentence for world $w^A \in W_{t_A}$ and $w^B \in W_{t_B}$, respectively, such that $w = \oplus(w^A, w^B)$ and $\phi^A \wedge \phi^B \Leftrightarrow \phi$.
Let $w \in W_t$ such that $w = \oplus(w^A, w^B)$. Let $\phi \in \Phi_t$ such that $\mathrm{I}_o\phi$ is true in w. From Definition 4.1, it must be the case that we have two sentences $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$ such that $\phi^A \wedge \phi^B \Rightarrow \phi$ (in W_t), $\mathrm{I}_{o_1}\phi^A$ is true in w^A, and $\mathrm{I}_{o_2}\phi^B$ is true in w^B. We also have $\mathrm{I}_o\phi^A$ and $\mathrm{I}_o\phi^B$ true in w. Suppose now that ϕ is the most specific sentence in w. Therefore, it must be the case that $\phi \Rightarrow \phi^A$ (in W_t) and $\phi \Rightarrow \phi^B$ (in W_t). Thus, $\phi \Rightarrow \phi^A \wedge \phi^B$ (in W_t). We conclude that $\phi \Leftrightarrow \phi^A \wedge \phi^B$ (in W_t). If ϕ^A and ϕ^B are not most specific for w^A and w^B respectively, then there exists $\phi_0^A \in \Phi_{t_A}$ such that $\phi_0^A \Rightarrow \phi^A$ (in W_{t_A}) and $\mathrm{I}_{o_1}\phi_0^A$ is true in w^A, and $\phi_0^B \in \Phi_{t_B}$ such that $\phi_0^B \Rightarrow \phi^B$ (in W_{t_B}) and $\mathrm{I}_{o_2}\phi_0^B$ is true in w^B. Therefore, $\mathrm{I}_{o_1}\phi_0^A$ and $\mathrm{I}_{o_2}\phi_0^B$ are true in w. Then $\phi_0^A \wedge \phi_0^B \Rightarrow \phi^A \wedge \phi^B$ (in W_t). As a result we have $\phi_0^A \wedge \phi_0^B \Rightarrow \phi$ (in W_t) but since ϕ is a most specific sentence for w then we have a contradiction. Therefore ϕ^A and ϕ^B must be most specific sentences in w^A and w^B, respectively.

[5] Independent representation of objects may not happen when the objects are related for instance via a temporal relationship, such as in video data.

We prove now the reverse. Let $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$, each most specific sentence for world $w^A \in W_{t_A}$ and $w^B \in W_{t_B}$, respectively, such that $w = \oplus(w^A, w^B)$ and $\phi^A \wedge \phi^B \Leftrightarrow \phi$. We show that $\phi \in \Phi_t$ is the most specific sentence for world w. Let $w \in W_t$ such that $w = \oplus(w^A, w^B)$. Suppose that $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$ are most specific sentences in w^A and w^B, respectively such that $\phi^A \wedge \phi^B \Leftrightarrow \phi$. If ϕ is not most specific for w , then there exists $\phi_0 \in \Phi_t$ such that $\phi_0 \Rightarrow \phi$ (in W_t) and $\mathrm{I}_o\phi_0$ is true in w. Therefore, $\phi_0 \Rightarrow \phi^A \wedge \phi^B$. Also there must exist two sentences $\phi_0^A \in \Phi_{t_A}$ and $\phi_0^B \in \Phi_{t_B}$ such that $\phi_0^A \wedge \phi_0^B \Rightarrow \phi_0$ (in W_t), $\mathrm{I}_{o_1}\phi_0^A$ is true in w^A, and $\mathrm{I}_{o_2}\phi_0^B$ is true in w^B. This implies that $\phi_0^A \Rightarrow \phi^A$ (in W_{t_A}) and $\phi_0^B \Rightarrow \phi^B$ (in W_{t_B}). This contradicts the fact that ϕ^A and ϕ^B are most specific sentences in w^A and w^B, respectively. Therefore, ϕ must be a most specific sentence. □□

4.2.2 Example

Applied to our example, the most specific sentences for the worlds in W_t are given in Table 7.

worlds in W_t	Most specific sentences
w_1, w_3, w_4	$a \wedge d$
w_2	$a \wedge b \wedge \neg c \wedge d$
w_5, w_7, w_8	a
w_6	$a \wedge b \wedge \neg c$
w_9, w_{11}, w_{12}	$\neg a \wedge d$
w_{10}	$b \wedge c \wedge \neg a \wedge d$
w_{13}, w_{15}, w_{16}	$\neg a$
w_{14}	$b \wedge \neg c \wedge \neg a$

Table 7: Most specific sentences for worlds in W_t

For world w_2, we have $\mathrm{I}_o(a \wedge d)$ and $\mathrm{I}_o(b \wedge \neg c \wedge d)$ true. Therefore, $mss_o(w_2) = a \wedge b \wedge \neg c \wedge d$. We can see (from Table 5) that $mss_o(w_2) = mss_{o_1}(w_2^A) \wedge mss_{o_2}(w_1^B)$ where $mss_{o_1}(w_2^A) = a \wedge b \wedge \neg c$ and $mss_{o_2}(w_1^B) = d$, and $\oplus(w_2^A, w_1^B) = w_2$.

4.2.3 Mass function

The aggregation of the mass functions modelling the uncertainty of the indexing for the objects o_1 and o_2 yields the mass function of the composite object. This is defined as follows.

Definition 4.3 (Aggregation of mass functions) *Let m_o be the mass function associated with the composite object $o \in \mathcal{O}$. Let m_{o_1}, m_{o_2} be the mass functions associated with the component objects $o_1 \in \mathcal{O}$ and $o_2 \in \mathcal{O}$, respectively. Let Φ_{t_A} and Φ_{t_B} be the frames of discernment associated with the objects o_1 and o_2, respectively. For $\phi \in \Phi_t$:*

$$m_o(\phi) = \mathcal{K} * \sum_{(\phi^A, \phi^B) \in \Gamma(\phi)} m_{o_1}(\phi^A) * m_{o_2}(\phi^B)$$

where

(1) $\mathcal{K} = \sum_{(\phi^A, \phi^B) \in \Phi_{t_A} \times \Phi_{t_B}} m_{o_1}(\phi^A) * m_{o_2}(\phi^B)$, ensuring that m_o is a mass function.

(2) The function $\Gamma : \Phi_t \mapsto \wp(\Phi_{t_A} \times \Phi_{t_B})$ maps every sentence ϕ in Φ_t to a subset of sentence pairs (ϕ^A, ϕ^B) with $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$, such that $\phi^A \wedge \phi^B \Leftrightarrow \phi$ in W_t.

Given a sentence $\phi \in \Phi_t$, if ϕ is a most specific sentence with respect to some worlds of W_t, then its weight $m_o(\phi)$ is computed upon the weights of pairs of sentences $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$ such that $\phi^A \wedge \phi^B$ is logically equivalent to ϕ. Such pairs of sentences is given by the set $\Gamma(\phi)$.

The above formula when given for $\Phi_{t_A} = \Phi_{t_B}$ corresponds to the Dempster's combination formula [6].

For a leaf object, any most specific sentence has a non-null mass value. We show that this also holds for a composite object.

Theorem 4.2 *For any $\phi \in MSS(o)$ (the set of most specific sentences for object o), $m_o(\phi) > 0$.*

Proof: Let $\phi \in MSS(o)$. Let $w \in W_t$ be such that ϕ is the most specific sentence in w. From Theorem 4.1, there exits $\phi^A \in \Phi_{t_A}$ and $\phi^B \in \Phi_{t_B}$ each most specific sentence for world w^A and w^B, respectively, where $w = \oplus(w^A, w^B)$ and $\phi^A \wedge \phi^B \Leftrightarrow \phi$. Therefore, $(\phi^A, \phi^B) \in \Gamma(\phi)$, and hence, $m_{o_1}(\phi^A) * m_{o_2}(\phi^B)$ contributes towards the value of $m_o(\phi)$. Since ϕ^A and ϕ^B are most specific sentences, by definition of the mass function for leaf objects (if o_1 and o_2 are leaf objects), $m_{o_1}(\phi^A) > 0$ and $m_{o_2}(\phi^B) > 0$. Therefore, $m_o(\phi) > 0$. The proof can be done by induction for the case where o_1 or o_2 are not leaf objects. □□

4.2.4 Example

Applied to our working example, the mass function for the composite object is given in Table 8. The weighted sentences were defined in Table 7:

$$MSS(o) = \{a \wedge d, a \wedge b \wedge \neg c \wedge d, a, a \wedge b \wedge \neg c, \neg a \wedge d, b \wedge c \wedge \neg a \wedge d, \neg a, b \wedge \neg c \wedge \neg a\}$$

The calculation of the mass function is shown in the table, i.e., the pair of sentences forming $\Gamma(\phi)$ for $\phi \in MSS(o)$ and the values of the mass functions for these sentences. In our case, $\mathcal{K} = 0.63$.

Most specific sentences ϕ for o	$\Gamma(\phi)$: (ϕ^A, ϕ^B)	$m_{o_1}(\phi^A) \times m_{o_2}(\phi^B)$	$m_o(\phi)$
$a \wedge d$	(a, d)	$0.3 \times 0.2 = 0.06$	0.096
$a \wedge b \wedge \neg c \wedge d$	$(a \wedge b \wedge \neg c, d)$	$0.4 \times 0.2 = 0.08$	0.126
a	$(a, \top)$	$0.3 \times 0.3 = 0.18$	0.285
$a \wedge b \wedge \neg c$	$(a \wedge b \wedge \neg c, \top)$	$0.4 \times 0.3 = 0.12$	0.191
$\neg a \wedge d$	$(\top, \neg a \wedge d)$	$0.1 \times 0.4 = 0.04$	0.064
$b \wedge \neg c \wedge \neg a \wedge d$	$(b \wedge \neg c, \neg a \wedge d)$	$0.2 \times 0.4 = 0.08$	0.126
$\neg a \wedge d$	$(\top, \neg a \wedge d)$	$0.1 \times 0.1 = 0.04$	0.064
$\neg a$	$(\top, \neg a)$	$0.1 \times 0.1 = 0.01$	0.016
$b \wedge \neg c \wedge \neg a$	$(b \wedge \neg c, \neg a)$	$0.2 \times 0.1 = 0.02$	0.032
$\sum_{\phi \in MSS(o)} m_o(\phi)$		0.63	1

Table 8: Mass function for the composite object and its calculation

The values obtained for the mass function m_0 seem intuitive, although nothing can be said about how effective they are at reflecting the uncertainty of the indexing of the composite object. In our previous work [8, 9], we used the Dempster's combination rule to derive the values of the mass function for a composite object. Our experiments showed that the rule led to a correct modelling of the uncertainty of the indexing. We expect the combination rule as provided by evidential reasoning to be as effective.

4.3 Summary

In this section, we have modelled the representation of a composite object. The representation was obtained as the aggregation of the representation of its components objects. The aggregation was defined at two levels: the indexing and the uncertainty of the indexing. This leads to a general model for heterogeneous structured documents, where the component objects can be of different media, distributed over several sites, or written in various languages.

5 Property of the aggregation

Efficient retrieval of heterogeneous structured documents is possible if we can minimise the number of objects in a structured document to be considered when calculating object relevance. One approach allowing this minimum search is to impose the following property on the representation of the objects composing a structured document: if a composite object is not relevant to the query, then none of its component objects are relevant to the query. Therefore, there is no point in going further down in the structure to seek for more relevant objects. This is referred to as *focussed retrieval.* Chiaramella et al [1] show that this property can be implemented if the aggregation operation satisfies the so-called **dependency constraint**:

> The representation of a composite object "implies" the representations of its component objects.

In practice, this means that elements indexing the component objects are "implied" by the elements indexing the composite object. For example, let o_1 and o_2 be two objects indexed by the sentences, respectively, *wine* and *grape*. If the object o is composed of the objects o_1 and o_2, then its representation should "imply" both *wine* and *grape*. In other words, the sentences *wine* and *grape* must be somewhat present in the representation of o.

We show that in the model presented in this paper, the dependency constraint holds if two assumptions are made. These are expressed in the following two propositions.

Proposition 5.1 *For all worlds $w^A \in W_{t_A}$ there exists a world $w^B \in W_{t_B}$ for which we can create a world $w \in W_t$ such that $w = \oplus(w^A, w^B)$. For all worlds $w^B \in W_{t_B}$ there exists a world $w^A \in W_{t_A}$ for which we can create a world $w \in W_t$ such that $w = \oplus(w^A, w^B)$.*

Proposition 5.1 can be interpreted as follows. We can always combine a sentence of S_{t_A} to at least one sentence of S_{t_B}. This should be satisfied otherwise, we cannot determine "completely" the representation of a composite object. If for a world $w^A \in W_{t_A}$, we cannot found a compatible world $w^B \in W_{t_B}$, then the sentences true in w^A may never be used to represent the content of the composite object. This case should definitively not happen.

Proposition 5.2 *Any most specific sentence has a non-null mass value value.*

A most specific sentence is a sentence that is used to represent the content of the object. Furthermore, it is a sentence for which a weight measuring its uncertainty in indexing an object is attached.

In constructing the mass function for a leaf object, Proposition 5.2 is satisfied. The most specific sentences are derived from the set of sentences yielded by the indexing process applied to the leaf object, so they are (the most concise) sentences

for which we have explicit evidence that they describe the content of the object. Hence their mass values should indeed be non-null.

The fact that $\top$ is a most specific sentence indicates that there is some ignorance about the content of the object. Therefore, if $\top$ is a most specific sentence, then its mass value should also be non-null.

For a composite object, we have shown with Theorem 4.2 that all most specific sentences for the composite object have non-null mass values, thus satisfying Proposition 5.2.

We show next that if Propositions 5.1 and 5.2 are satisfied, the dependency constraint holds. This means that if a sentence ϕ indexes a component object, then there exists a sentence ψ that indexes the composite object such that $\psi \Rightarrow \phi$. This is formally expressed in the following theorem.

Theorem 5.1 (Dependency constraint)

Let the indexing structures for o_1 and o_2 be, respectively, $F_{o_1} = \langle F_{t_A}, S_{o_1}, \pi_{o_1} \rangle$ and $F_{o_2} = \langle F_{t_B}, S_{o_2}, \pi_{o_2} \rangle$. Let $F_o = \langle F_t, S_o, \pi_o \rangle$ be the indexing structure modelling the indexing of the composite object o.

Let Φ_{t_A} and Φ_{t_B} be the frames of discernment associated with the objects o_1 and o_2. Let Φ_t be the frame of discernment associated with the composite object o.

Let m_{o_1}, m_{o_2} be the mass functions associated with the component objects o_1 and o_2, respectively. Let m_o be the mass function associated with the composite object o.

(1) Let $\psi \in \Phi_{t_A}$ such that $m_{o_1}(\psi) > 0$. Then there exists $\phi \in \Phi_t$ such that $\phi \Rightarrow \psi$ and $m_o(\phi) > 0$.

(2) Let $\psi \in \Phi_{t_B}$ such that $m_{o_2}(\psi) > 0$. Then there exists $\phi \in \Phi_t$ such that $\phi \Rightarrow \psi$ and $m_o(\phi) > 0$.

Proof: The proofs for parts (1) and (2) are the same. We only give the proof for (1).

Let $\psi \in \Phi_{t_A}$ such that $m_{o_1}(\psi) > 0$. This means that $\psi \in MSS(o_1)$. Let $w^A \in W_{t_A}$ such that ψ is true in w^A (i.e. $\pi_{o_1}(w^A, \mathrm{I}_{o_1}\psi) = true$). From Proposition 5.1, there exists a world $w^B \in W_{t_B}$ such that $\oplus(w^A, w^B)$ exists. Let $\oplus(w^A, w^B) = w \in W_t$. From Definition 4.1, we must have $\mathrm{I}_o\psi$ true in w. Let $mss_{o_2}(w^B) = \psi'$. We also have $\mathrm{I}_o\psi'$ true in w (from Definition 4.1). That is, $\mathrm{I}_o\psi$ and $\mathrm{I}_o\psi'$ are both true in w. Therefore, $\mathrm{I}_o(\psi \wedge \psi')$ is true in w (Definition 4.1). Let $\phi \Leftrightarrow \psi \wedge \psi'$. That is, there exists $\phi \in S_t$ such that $\phi \Rightarrow \psi$.

What remains to be shown to prove Theorem 5.1 is that such ϕ is in $MSS(o)$. For this, it is sufficient to show that $mss_o(w) = \phi = \psi \wedge \psi'$. By construction, both ψ and ψ' are most specific sentences for w^A and w^B, respectively. Therefore, from Proposition 5.2 and Theorem 4.1, the most specific sentence for w must be $\psi \wedge \psi'$, so ϕ is a most specific sentence. From Theorem 4.2, it must be the case that $m_o(\phi) > 0$. □□

6 Retrieval

Given a structured document, retrieval must return to the user those objects (if they exist) in the document that are most relevant to his or her information need. The returned object may be a leaf (only that object concerns the query), a composite object (all the components of that object concerns the query), the root object (the whole document concerns the query). The returned objects are displayed to the user, and then constitute *access points* from where the user can decide to browse the structure if needed. An object being displayed to a user means that most of

its component objects, direct or indirect are considered relevant to the information need. The object is displayed to the user, with a summary of its content as computed by the aggregation operator.

Consider a hypermedia system, such as the world-wide-web, in which documents are hierarchically structured. Hypermedia documents and hyper-links would correspond to objects and the containment relationship between objects, respectively. For web documents, XML meta-data would provide information about the types of the objects. Our model would allow to target the best access points (web documents) to the web site, who can then be browsed up or down by users. We have implemented a subset of this model using text data only, the Dempster-Shafer's theory of evidence, and a web museum site [25]. We are currently pursuing a full implementation of the model presented in this paper.

The retrieval process is very similar to that described in our previous work [4, 8, 9, 25]. The main addition is the modelling of the query, since now objects can be of different types. This is described in Section 6.1. The expression of the relevance of an object to the query is described in Section 6.2. One main asset of our approach is that retrieval can be focussed to those objects that are composed of relevant objects. This is discussed in Section 6.3. Returning retrieving objects independently of their structure is not sufficient [2]. Several objects may be retrieved as answers to a query which belong to the same structured document. How the relationships between retrieved objects are taken into account (thus reducing cognitive overload) is discussed in Section 6.4.

6.1 Modelling a query

An information need, as phrased in a query, is represented as a sentence q. The question is to which sentence space the sentence q belongs. To evaluate the relevance of an object at any level in the structure, the sentence space must define (syntactically) the indexing vocabulary that corresponds to the aggregation of all indexing vocabularies. This is the indexing vocabulary associated with the aggregated type defined over all the types in $\mathcal{T}$. The aggregated type can be viewed as the type of a *fictitious* object composed of all root document objects. We refer to this type as t_{top}, and to the sentence space as S_{top}. Therefore $q \in S_{top}$.

In practice, symbolising the indexing vocabulary associated with the type t_{top} is not necessary, because, when the relevance of an object of type t is computed, the query sentence is transformed to one that belongs to S_t. The reason is that for such an object, only sentences that can describe its content or the content of its component objects can be used. For instance, seeking objects about "wine" should be with respect to those objects that can be indexed by *wine* or by equivalent sentences (e.g., *vin*).

We define a projection operator that transforms a sentence ϕ of S_{top} to one of S_t where S_t is the sentence space associated with a type t.

Definition 6.1 (Projection) *Let $t \in \mathcal{T}$ be a type and S_t its associated sentence space. The function $\Pi_t : S_{top} \mapsto S_t$ maps a sentence of S_{top} to a sentence of S_t as follows (here ϕ and ψ are sentences of S_{top}, and p is a proposition of P_{top}):*

(1) $\Pi_t(\phi \wedge \psi) = \Pi_t(\phi) \wedge \Pi_t(\psi)$;

(2) $\Pi_t(\phi \vee \psi) = \Pi_t(\phi) \vee \Pi_t(\psi)$;

(3) $\Pi_t(\neg\phi) = \neg\Pi_t(\phi)$;

(4) The final case is as follows:

$$\Pi_t(p) = \begin{cases} p & \text{if } p \in P_t, \\ \varphi & \text{if } \varphi \in S_t \text{ and } \varphi \text{ and } p \text{ are informationally equivalent,} \\ \bot & \text{otherwise} \end{cases}$$

Suppose that the user is looking for objects about "fish and wine". The query sentence is $fish \wedge wine$ which belongs to S_{top}. Let o_1 and o_2 be two objects where $type(o_1) = t_A$ and $type(o_2) = t_B$. Suppose that the associated sentence spaces S_{t_A} and S_{t_B} contain the sentences $fish \wedge wine$ and $fish$, respectively. We assume that there are no informationally equivalent sentences to $wine$ in S_{t_B}. Therefore:

$$\begin{aligned} \Pi_{t_A}(fish \wedge wine) &= fish \wedge wine \\ \Pi_{t_B}(fish \wedge wine) &= \bot \end{aligned}$$

The queries used to evaluate the relevance of the objects o_1 and o_2 are $fish \wedge wine$ and $\bot$, respectively. The use of $\bot$ is appropriate because the object o_2 cannot be about "wine", so it cannot be about "wine and fish". If the original query was $fish \vee wine$, then:

$$\begin{aligned} \Pi_{t_A}(fish \vee wine) &= fish \vee wine \\ \Pi_{t_B}(fish \vee wine) &= fish \end{aligned}$$

The query sentence for object o_2 is $fish$. This is correct since the object can be about "fish", so it can be about "wine or fish".

We discuss negation (i.e., $\Pi_t(\neg\phi) = \neg\Pi_t(\phi)$). In IR, there are two interpretations of negation: explicit and implicit. For an object to not be about for example "wine", explicit negation means that the object must be indexed by a sentence that logically implies $\neg wine$, whereas implicit negation means that the object is not indexed by a sentence that implies "wine", including the sentence "wine" (this is the closed-world assumption). We use explicit negation in this work. The definition of Π_t is compatible with this interpretation. For an object $o \in \mathcal{O}$ of type $t \in \mathcal{T}$ to not be about "wine" (the query is $\neg wine$), it must be the case that the object is indexed by a sentence that logically implies $\neg wine$ or the negation of a sentence equivalent to $wine$. Formally, there must exist a world $w \in W_t$ and a sentence $\Phi \in S_t$ such that $\phi \Rightarrow \Pi_t(\neg wine)$ (where $\Pi_t(\neg wine) = \neg\Pi_t(wine)$) and $\pi_o(w, \mathrm{I}_o\phi) = true$. For instance, if $\Pi_t(wine) = vin$ ("vin" is the French word for "wine"), and the object is indexed by $\neg vin$ (ϕ is $\neg vin$), then the object is not about "wine".

6.2 Relevance of an object to an information need

Given the representation of a query, we describe next how to express the relevance of an object to the query. In previous work [4], we use the belief function [6] of the Dempster-Shafer theory of evidence for this purpose. The same function can be used with a model based on evidential reasoning.

Definition 6.2 (Belief function) *Let $o \in \mathcal{O}$ be an object of type $t \in \mathcal{T}$ and whose indexing is modelled by the indexing structure $F_o = \langle F_t, S_o, \pi_o \rangle$ and let m_o be its associated mass function. Given a sentence $\phi \in S_t$, the* total belief that the object supports *that sentence is modelled by the belief function $Bel_o : S_t \mapsto [0, 1]$ defined as follows:*

$$Bel_o(\phi) = \sum_{\psi \Rightarrow \phi, \psi \in MSS(o)} m(\psi)$$

The quantity $Bel(\phi)$, if not null, indicates that the object contains information that concerns ϕ. This is because $Bel_o(\phi)$ is based on the sentences that explicitly index the object o (the most specific sentences of o, $MSS(o)$) and that support the sentence ϕ. It also takes into account the beliefs associated to their use; the higher their beliefs, the higher the relevance. Also, the greater their number, the higher the relevance. Belief functions are therefore used to evaluate the relevance of an object to a query. The general definition is a s follows.

Definition 6.3 (Relevance of an object to a query) *Let $o \in \mathcal{O}$ be an object of type $t \in \mathcal{T}$ and whose indexing is modelled by the indexing structure $F_o = \langle F_t, S_o, \pi_o \rangle$ and let Bel_o be its associated belief function. Given a query $q \in S_{top}$, the relevance of the object o to the query is given by the function $Rel : \mathcal{O} \times S_{top} \mapsto [0,1]$ defined as follows:*

$$Rel(o, q) = Bel_o(\Pi_t(q))$$

Π_t transforms the query sentence to one that can be evaluated with respect to an object of type t.

For any two objects o and o', if $Rel(o, q) < Rel(o', q)$, the object o' contains more information pertinent to the query q than does the object o, so is more relevant to the query than o' is. Objects can then be ranked according to R.

We illustrate the use of the belief function to express object relevance with an example using objects o_1 and o_2 whose mass functions are given in Table 5. We use the following two queries: $q_1 = a$ and $q_2 = b \vee d$. Table 9 shows the transformation of the query sentences to sentences of the sentence spaces S_{t_A} and S_{t_B}.

Queries q_i	$\Pi_{t_A}(q_i)$	$\Pi_{t_B}(q_i)$
$q_1 = a$	a	a
$q_2 = b \vee d$	b	d

Table 9: Transformation of queries

For query q_2, the first disjunct b can be supported by an object of type t_A whereas the second disjunct d can be supported by an object of type t_B. The relevance of each object for the two queries is given in Table 10.

Queries q_i	$Rel(o_1, q_i)$	$Rel(o_2, q_i)$
$q_1 = a$	0.8	0.3
$q_2 = b \vee d$	0.7	0.9

Table 10: Relevance values for objects o_1 and o_2

The objects o_1 and o_2 are hence ranked for each query as shown in Table 11.

Queries	Rank
$q_1 = a$	o_1, o_2
$q_2 = b \vee d$	o_2, o_1

Table 11: Ranking for object o_1 and o_2

Both objects are relevant to query q_1. Object o_1 is more relevant than is object o_2 because the support for a by object o_1 is higher to the support for a by object o_2.

Both objects are relevant to query q_2, object o_1 because of b and object o_2 because of o_2. Object o_2 is more relevant than is object o_1 because the support for d by object o_2 is higher to the support for b by object o_1.

6.3 Focussed retrieval

We use the criterion of the dependency constraint discussed in Section 5 to limit the number of objects to be considered in the retrieval process. We traverse the structured document commencing from the root object. Let o be the object whose relevance is being currently investigated. If the object is not relevant (there are no sentences $\phi \in MSS(o)$ such that ϕ logically implies the query sentence $\Pi_t(q)$, i.e., $Rel(o,q) = Bel_o(\Pi_t(q)) = 0$), then there are no objects composing o that logically imply the query sentence. Therefore, there is no point traversing the document structure further down. This strategy was extensively discussed in [1, 4] and implemented in [9].

6.4 Displaying the most relevant objects

We know how to estimate the relevance of any object to a query. The next step is to use the values obtained to determine among the relevant objects, which objects should be displayed to the user, taking into account that the objects can be related. We use the same approach developed in our previous work [9], the difference being that now heterogeneous objects can be manipulated.

Let q be a query. Let $o_1, \ldots, o_n$ be the objects composing a structured document. For each object o_i, $Rel(o_i, q)$ is the relevance of the object to the query q. The *most* optimal access point for browsing is the object most relevant to the information need as given by R (i.e., the object with the highest belief value). The *next* optimal access point is the object with the next highest belief that is not the descendant of any object with higher belief value. However, if one such object, let us say o_k is a descendant of any object already identified as optimal starting point, then o_k is not an optimal starting point. If for two objects o and o' we have that $Rel(o,q) = Rel(o',q)$, the object deeper in the structure is considered first. This strategy was successfully implemented in [9].

Let object o be composed of object o_1 and o_2. We consider the two queries given in Section 6.2. The relevance values for the three objects o, o_1 and o_2 to the two queries are shown in Table 12.

Queries q_i	$Rel(o_1, q_i)$	$Rel(o_2, q_i)$	$Rel(o, q_i)$
$q_1 = a$	0.8	0.3	0.698
$q_2 = b \vee d$	0.7	0.9	0.731

Table 12: Relevance values for objects o_1 and o_2

The optimal access point for query q_1 and q_2 are objects o_1 and o_2, respectively. The relevance of object o_2 to query q_1 comes from ignorance ($m_{o_2}(\top) = 0.3$), so it seems intuitive that the document should be accessed via object o_1 first. The relevance of objects o_1 and o_2 is due to various sentences supported by the two objects. The highest support comes from object o_2, which hence constitutes the document access point for browsing.

6.4.1 Summary

Based on the representation of the objects forming heterogeneous structured documents developed in the previous sections, our retrieval strategy is as follows:

- the relevance of the query at any level in the structure is calculated;
- objects that are definitely not relevant are discarded early (focussed retrieval); and
- the most optimal access points to the document are displayed to the user, thus reducing cognitive overload.

7 Related work

Research with similar or complementary aims falls into three main areas: information retrieval, hypertext and database.

The approaches developed in *information retrieval* can be classified into four groups. The first group, which follows approaches most similar to ours is that of [1], [26], [27] and [28]. They all propose model to retrieve documents that have an underlying structure. Our work is an extension of Chiaramella et al's model [1] (the part dealing with content-based retrieval). We have added uncertainty to their model. The main difference between our model and Roelleke's model [26] is the formalism used to express the model. Roelleke uses a four-valued logic with a probabilistic approach. The representation of the content of an object is defined in terms of an aggregation of the representation of the content of its component objects. The work also supports multimedia and distributed structured documents. Myaeng et al [27] use an inference network model that is applied to SGML documents. The central idea is to represent SGML objects, of various granularities, in a network. The degree to which an object, at any level, supports the query is calculated by considering its component objects (probabilities are propagated along the network). The model however does not build a representation of the content of an object based on that of its component objects. In [28], the relevance value of an object is computed as the combination of the relevance values of its component objects using probability theory. The work was applied to distributed documents. Neither of the models proposed in [26], [27], or [28] aims at providing focussed retrieval.

The second group of approaches uses *passage retrieval* which aims at retrieving documents based on the most relevant part of a text document. [29] presents an approach based on retrieval by fixed-length passage (a text window of 150-300 words in length). These can be compared against a query to obtain a series of scores for overlapping passages. Ranking the documents by the highest-scoring passage yielded significantly better results than retrieval by whole document score. However the most significant results in this work came from combining document-level matching with passage-level matching information. Similar results were obtained in [30]. [31] also demonstrated the utility of combining evidence from different sections and, more importantly from different levels of structure (sentences, paragraphs, etc). [32] proposed an alternative technique, TextTiling, that is capable of retrieving documents by topical structure. TextTiling imposes a structure on full-length documents by splitting them into coherent multi-paragraph segments that represent subtopics in the documents. [32] examined various methods for ranking documents based on subtopic structuring. None of the passage retrieval approaches discusses the possibility of retrieving aggregated objects (other than the document itself), that is, objects whose sub-objects are all relevant.

In the third group, data models representing the semantic structure of documents (e.g., title vs section) are developed [33, 34, 35]. For instance, Burkowski's data model [34] is expressed by an algebra. Retrieval is done via a query language defined upon the data model. [34] allows, in addition, ranking of components. These approaches are very specific to the content and the structure of documents. It is

then difficult to generalise them to deal with other aspects of structured documents (e.g., returning aggregated objects, or non-text objects).

The fourth group of approaches dealing with structured documents aims at constructing indexes that not only locate keywords in a text, but also structure data (e.g., beginning and end of a section, or title). An example of such an approach is described in [36]. For each keyword and structure data, a list of the locations of their occurrence in the text documents is compiled. An expressive query language is thus defined to search with respect to content and structure. This approach is not intended for the retrieval of aggregated objects, and hence does not allow for focussed retrieval.

Hypertext [37] is a medium for presenting related information units. Hence, hypertext retrieval methods can be used to retrieve structured documents. The work described by Frisse [38] illustrates how hypertext can be used to provide a means of navigating through long, related texts. Frisse defines an hypertext query processing mechanism which, given relevant objects, selects those objects to be displayed to the user. The approach determines the optimal objects to be displayed to a user. An optimal object is one most relevant to the information need and in addition is an optimal starting point for browsing. The most optimal starting point for browsing is the object most relevant to the information need as given by the retrieval function. The next optimal starting point is the object with the next highest relevance that is not the descendant of any object with higher relevance. This approach takes into account the relationships between objects, thus attempting to reduce cognitive overload. As in [28], the relevance of an object is based on the relevance of its components objects. Our approach can be viewed as a means to implement such a strategy. A main difference is that our approach computes the relevance of an object to a query based on the representation of the objects. In [38], relevance values are combined. As a result, the approach does not implement focussed retrieval.

Research in the *database* area also deals with structured document retrieval. The aim is to extend existing database technology to deal with structures. A particular application of this work is text documents with an underlying structure specified by a mark-up language such as SGML. An example of such an approach is that of [39] who use object-oriented databases. The query language offers so-called containment operators for matching attribute values. For example, they allow users to retrieve component parts (a section, a chapter) that contain a particular set of keywords, or that contain a sub-part that contains a particular set of keywords. Typical for database query language, the underlying schema must be known by the user formulating the query. To remedy this problem, path expressions can be used [39]. Database approaches, to be effective require an expressive query language, whereas in IR, content-based retrieval is usually performed by submitting to the system a set of keywords as a query. Also, none of them discuss the possibility of implementing focussed retrieval.

The strength of our work, then, is that it provides a general framework for heterogeneous structured document retrieval that, as well as being media, site and language independent, considers the relationships between retrieved document parts, encapsulates the notions of aggregation and focussed retrieval, utilises the structure of the document without extending the query language.

8 Conclusion

Heterogeneous structured documents are documents whose components can be of different **types**: various media, located in a number of sites, or written in several languages. Such documents are becoming increasingly more preponderant

in todays information systems (e.g. web documents which contain text, images, sounds, etc; digital libraries which consist of documents distributed among several databases; multilingual documents such as those stored in the European Commission, etc.). Having document components of various types mean that **different indexing vocabularies** are involved in representing the content of a document. We need a model that can encompass the disparity of indexing vocabularies.

In this paper, we presented a formal model for representing heterogeneous structured documents based on **evidential reasoning**. We can model the following aspects necessary for the representation of heterogeneous structured documents:

- the indexing vocabularies associated with document components;
- the representation of document components;
- the aggregation operation which determines the representation of composite objects based on the representations of their component objects;
- the informational relatedness of the indexing vocabularies.

By being formal, the model can be used to study various properties inherent in the representation of heterogeneous structured document; thus leading to more effective representation and retrieval of heterogeneous structured documents. We have already shown that one property, the dependency constraint, enables a focussed retrieval of objects.

This paper does not present the final word in modelling the representation and retrieval of heterogeneous structured documents. We have yet to investigate:

- how our model can be included in the more general model proposed by Chiaramella et al [1] to provide for structure- and attribute-based retrieval as well as content-based retrieval, and
- how our model can be effectively and efficiently implemented using knowledge relating different indexing vocabularies.

However we believe that we have provided a credible, formal framework for further investigations.

Acknowledgement

Thanks to Ian Ruthven, Thomas Roelleke and the anonymous referees for their insightful comments.

References

[1] Y. Chiaramella, P. Mulhem, and F. Fourel. A model for multimedia information retrieval. Technical Report Fermi ESPRIT BRA 8134, University of Glasgow, 1996.

[2] Y. Chiaramella. Browsing and querying: two complementary approaches for multimedia information retrieval. In *Proceedings Hypermedia - Information Retrieval - Multimedia*, Dortmund, Germany, 1997.

[3] Y. Chiaramella and A. Kheirbek. An integrated model for hypermedia and information retrieval. *Information Retrieval and Hypertext*, 1996.

[4] M. Lalmas. Dempster-Shafer's theory of evidence applied to structured documents: modelling uncertainty. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 110–118, Philadelphia, PA, USA, 1997.

[5] A. P. Dempster. A generalization of the Bayesian inference. *Journal of Royal Statistical Society*, 30:205–447, 1968.

[6] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[7] M. Lalmas and Y. Chiaramella. Dempster-shafer's theory of evidence applied to structured documents. Technical Report Fermi Esprit BRA 8134, University of Glasgow, 1997.

[8] M. Lalmas and I. Ruthven. A model for structured document retrieval: empirical investigations. In *Proceedings Hypermedia - Information Retrieval - Multimedia*, pages 53–66, Dortmund, Germany, 1997.

[9] M. Lalmas and I. Ruthven. Representing and retrieving structured documents with Dempster-Shafer's theory of evidence: Modelling and evaluation. *Journal of Documentation*, 54(5):529–565, 1998.

[10] E. H. Ruspini. The logical foundations of evidential reasoning. Technical Report 408, SRI International, 1986.

[11] E. H. Ruspini. Epistemic logics, probability, and the calculus of evidence. In *Proceedings of the Joint Conference on Artificial Intelligence*, pages 924–931, Menlo Park, CA, 1987.

[12] E. H. Ruspini. The semantics of vague knowledge. *Revue Internationale de Systemique*, 3:387–420, 1989.

[13] E. H. Ruspini. Approximate reasoning: Past, present, and future. *Information Sciences*, 57-58:297–317, 1991.

[14] S. A. Kripke. Semantic analysis of modal logic. *Zeitschrift fur Mathematische Logik und Grundlagen der Mathematik*, 9:67–96, 1963.

[15] L.T.F. Gamut. *Logic, Language and Meaning*, volume I, Intensional logic and logical grammar. The University of Chicago Press, 1991.

[16] L.T.F. Gamut. *Logic, Language and Meaning*, volume II, Introduction to Logic. The University of Chicago Press, 1991.

[17] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[18] Marcos Theophylactou. Document retrieval using natural language processing and the dempster-shafer theory of evidence. Master's thesis, University of Glasgow, 1997.

[19] S. S. Schoken and R.A. Hummel. On the use of the Dempster-Shafer model in information indexing and retrieval applications. *International Journal of Man-Machine Studies*, 39:1–37, 1993.

[20] W. Teixeira de Silva and R. L. Milidiu. Belief function model for information retrieval. *Journal of the American Society for Information Science*, 44(1):10–18, 1993.

[21] C. J. van Rijsbergen. *Information Retrieval.* Butterworths, London, 2 edition, 1979.

[22] G. Salton and M. J. McGill. *Introduction to modern information retrieval.* McGraw-Hill Book Company, 1980.

[23] N. Fuhr and C. Buckley. A probabilistic learning approach for document indexing. *ACM Transaction of Information Systems*, 9(3):223–248, 1991.

[24] M. Theophylactou and M. Lalmas. A Dempster-Shafer belief model for document retrieval using noun phrases. In *Proceedings of BCS Information Retrieval colloquium, Grenoble*, 1998.

[25] E. Moutogianni. A Dempster-Shafer model for structured document retrieval: Implementation and experiments on a Web Museum collection. Master's thesis, Queen Mary and Westfield College, 1999.

[26] T. Roelleke. *POOL: Probabilistic Object-Oriented Logical Representation and Retrieval of Complex Objects - A Model for Hypermedia Retrieva.* PhD thesis, University of Dortmund, Germany, 1999.

[27] S.H. Myaeng, D. H. Jang, M. S. Kim, and Z. C. Zhoo. A flexible model for retrieval of SGML documents. In *Proceedings of ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 138–145, Melbourne, Australia, 1998.

[28] C. Baumgarten. A probabilistic model for distributed information retrieval. In *Proceedings of ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 258–266, Philadelphia, USA, 1997.

[29] J. Callan. Passage-level evidence in document retrieval. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 302–310, Dublin, Ireland, 1994.

[30] R. Wilkinson. Effective retrieval of structured documents. In *Proceedings of ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 311–317, Dublin, Ireland, 1994.

[31] G. Salton, J. Allan, and C. Buckley. Approaches to passage retrieval in full text information systems. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 49–58, Pittsburgh, USA, 1993.

[32] M. A. Hearst and C. Plaunt. Subtopic structuring for full-length document access. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 59–68, Pittsburgh PA, USA, 1993.

[33] I.A. Macleod. Storage and retrieval of structured documents. *Information Processing and Management*, 26(2):197–208, 1990.

[34] F.J. Burkowski. Retrieval activities in a database consisting of heterogeneous collections of structured texts. In *Proceedings of ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 112–125, Copenhagen, Denmark, 1992.

[35] G. Navarro and R. Baeza-Yates. A language for queries on structured and contents of textual databases. In *Proceedings of ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 93–101, Seattle, USA, 1995.

[36] T. Dao. An indexing model for structured documents to support queries on content, structure and attributes. In *Advances in Digital Libraries Conference*, 1998.

[37] J. Conklin. Hypertext: An introduction and survey. *IEEE Computer*, 29(9):17–41, 1987.

[38] M.E. Frisse. Searching for information in a hypertext medical handbook. *Communications of the ACM*, 31(7):880–886, 1988.

[39] S. Abiteboul, S. Cluet, V. Christophides, T. Milo, G. Moerkotte, and J. Simeon. Querying documents in object databases. *International Journal on Digital Libraries*, 1:1–9, 1997.

Appendix: Overview of the notations

$\mathcal{O}$	set of objects
$Leaf(\mathcal{O})$	set of leaf objects
$\mathcal{T}$	set of types
$type : O \mapsto \wp(\mathcal{T})$	type of an object
P_t	proposition space associated with type t
S_t	sentence space associated with type t
W_t	set of possible worlds associated with type t
$v_t : W_t \times P_t \mapsto \{true, false\}$	assigns a truth value to a proposition in a world
$S_t : W_t \times S_t \mapsto \{true, false\}$	assigns a truth value to a sentence in a world
$F_t = \langle S_t, W_t, v_t, \pi_t \rangle$	type structure associated with type t (models indexing vocabulary)
$\Rightarrow$	logical implication
$\Leftrightarrow$	logical equivalence
$\oplus : W_{t_A} \times W_{t_B} \mapsto W_t$	relates pair of worlds in $W_{t_A} \times W_{t_B}$ to created world in W_t
I_o	modal operator for object o
S_o	modal space for object o
$F_o = \langle F_t, S_o, \pi_o \rangle$	indexing structure for object o (models object representation)
$ID : Leaf(\mathcal{O}) \mapsto \wp(S_t)$	sentences obtained from the indexing process for leaf objects
$ID^{\wedge} : Leaf(\mathcal{O}) \mapsto \wp(S_t)$	sentences implied by sentences obtained from the indexing process for leaf objects
Φ_t	frame of discernment associated with type t
$mss_o : W_t \mapsto \Phi_t$	most specific sentence for a world
$MSS : \mathcal{O} \mapsto \wp(\Phi_t)$	set of most specific sentences for an object
$m_o : \Phi_t \mapsto [0, 1]$	mass function for object o
$\Gamma : \Phi_t \mapsto \wp(\Phi_{t_A} \times \Phi_{t_B})$	sentence pairs in $\Phi_{t_A} \times \Phi_{t_B}$ whose conjunction is equivalent to sentence in Φ_t
$\Pi_t : S_{top} \mapsto S_t$	transforming a query sentence to a sentence in S_t
$Bel_o : S_t \mapsto [0, 1]$	belief function for object o
t_{top}	top most type
$Rel : \mathcal{O} \times S_{top} \mapsto [0, 1]$	relevance of an object to a query