In *Proceedings of ACM SIGIR-96, 19th International Conference on Research and Development in Information Retrieval, Zurich, CH, 18-22 August 1996,* pages 197–205.

A Relevance Terminological Logic for Information Retrieval

Carlo Meghini and Umberto Straccia
{meghini,straccia}@iei.pi.cnr.it
Consiglio Nazionale delle Ricerche
Istituto di Elaborazione dell'Informazione
Via S. Maria, 46 - I-56126 Pisa, Italy

Abstract

A Terminological Logic is presented as an information retrieval model, with a four-valued semantics that gives to its inference relation the flavor of relevance, that is a strict connection in meaning between the premises and the conclusion of the arguments licensed by the logic. The logic also permits the expression of meta-knowledge enforcing a closed-world reading of the knowledge concerning specified individuals and primitive concepts. A Gentzen-style, sound and complete calculus for reasoning in the logic is given, thus establishing the basis for an information retrieval engine.

1 Introduction

Recently, *Terminological Logics* (TLs, for short) have been proposed as a logical model for Multimedia Information Retrieval (MIR) [Meghini *et al.*, 1993; Sebastiani, 1994] by interpreting the retrieval task in terms of logical implication: given a set of assertions Σ (the document base containing the document descriptions) and a query concept Q (a description of the class of documents to be retrieved), retrieve the documents d such that Σ *logically implies* $Q(d)$, in symbols, $\Sigma \models Q(d)$.

This view, termed as the terminological model (TM), combines the logical approach to information retrieval (IR) modelling with the conceptual modelling approach to information systems. While the former indicates a logic as the most suitable tool to capture the inference intrinsic in the retrieval process, the latter proposes the explicit representation of knowledge as the foundational principle of any intelligent system supporting information-intensive applications. Conceptual modelling can be employed in text retrieval only for the representation of domain knowledge and the articulation of queries, because its application to document representation would imply manual indexing, an unrealistic assumption for most applications. However, when multimedia documents are considered, conceptual modelling is likely to play a major role also at the document representation level, given the enormous difficulty of automatically constructing adequate content representations of this kind of documents.

The TM proposed in [Meghini *et al.*, 1993] has been thoroughly investigated from several point of views, and certain requirements have been identified, which have led to the definition of the logic presented in this paper, named $\mathcal{AL}$MIR ($\mathcal{AL}$, the name of the family of TLs on which it is based + MIR).

The basic requirement concerns the capturing of relevance. The classical implication relation does not take into account the relevance of premises to the conclusions of its licensed arguments, what is instead deemed as essential to the whole IR task. In order to overcome this problem and define an inference relation closer to the spirit of IR, the classical, two-valued semantics has been replaced by a four-valued semantics. This semantics, borrowed from relevance logic[Anderson and Belnap, 1975], permits to define an implication relation that requires a tight connection in meaning between a query and the documents that are retrieved in response to it. Happily, the connection in meaning captured by the semantics is tight enough to also handle inconsistency in a way that is appropriate to IR. In particular, a contradictory document base does not entail every query, as it happens with classical semantics. It is important to notice that the inconsistency among the content representations of different documents may well be the rule rather than the exception.

The second requirement that has emerged from our studies is, if possible, even more fundamental than the previous one, and relates to the usage of a TL for modelling documents. Classical logics adopt the so-called open-world assumption, that is they interpret a set of logical sentences as a description of a state of affairs that may be *partial*, in the sense that it may lack knowledge about some aspects. Thus, an incomplete document base may not entail a sentence α nor its negation $\neg\alpha$. The open-world assumption may turn out to be extremely inconvenient when retrieving documents. Because in order to obtain the desired behavior, the document indexer must specify not only what documents are, but also what they are not, what usually amounts to an overwhelming number of negative assertions. This has led us to conclude that, under certain circumstances, an IR system should adopt a closed-world view of the underlying document base, using the inability of establishing a fact as evidence of the contrary.

A third factor that has led to the definition of $\mathcal{AL}$MIR are the serious computational problems from which the initial logic has been discovered to suffer [Buongarzoni *et al.*, 1995].

Construct	Denotation in an interpretation $\mathcal{I}$
A	$A^{\mathcal{I}}$
$\top$	$\Delta^{\mathcal{I}}$
$\bot$	$\emptyset$
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$\forall R.C$	$\{d \in \Delta^{\mathcal{I}} \mid \forall d', (d, d') \in R^{\mathcal{I}} \rightarrow d' \in C^{\mathcal{I}}\}$
$\exists R.C$	$\{d \in \Delta^{\mathcal{I}} \mid \exists d', (d, d') \in R^{\mathcal{I}}$ & $d' \in C^{\mathcal{I}}\}$
P	$P^{\mathcal{I}}$
$R \circ Q$	$\{(d_1, d_2) \in \Delta^{\mathcal{I}} \mid \exists d', R^{\mathcal{I}}(d_1, d')$ & $Q^{\mathcal{I}}(d', d_2)\}$

Table 1: Syntax and two-valued semantics of $\mathcal{ALC}$MIR.

Next Section introduces syntax and standard semantics of $\mathcal{ALC}$MIR, as well as the fundamental concepts of the TM of IR. Section 3 presents the four-valued semantics for $\mathcal{ALC}$MIR and discusses its rationale and formal properties. Section 4 extends the four-valued semantics by means of closed world capabilities and overviews the properties of our final logic. Section 5 gives a sound and complete calculus for $\mathcal{ALC}$MIR. Section 6 concludes.

2 Introducing $\mathcal{ALC}$MIR

The syntax of $\mathcal{ALC}$MIR and its standard, two-valued semantics are introduced in Table 1[1]. The building blocks of the language are primitive concepts (denoted by the letter A in Table 1), which represent basic classes of the application domain, and primitive roles (denoted by P), which represent basic properties of classes. Complex concepts (C and D) and roles (R and Q) are built out of primitive symbols via the language constructors. For example, the complex concept:

$$\texttt{Order} \sqcap \forall \texttt{Sender}.\texttt{CarVendor}$$

is obtained combining the primitive concepts `Order` and `CarVendor` and the primitive role `Sender` by the conjunction ($\sqcap$) and the universal quantification ($\forall$) constructors. The concept describes the set of orders whose sender is a car vendor. From a logical point of view, concepts can be seen as unary predicates, whereas roles are binary predicates.

The classical, two-valued semantics of TLs is based on the notion of interpretation. An interpretation $\mathcal{I}$ consists of a domain $\Delta^{\mathcal{I}}$, a non-empty set, and of a function mapping primitive concepts into subsets of $\Delta^{\mathcal{I}}$ and primitive roles into subsets of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. As shown in Table 1, the interpretation of complex concepts and roles is obtained by appropriately combining the interpretation of their primitive components. It can be verified that the interpretation of the above concept is:

$$\texttt{Order}^{\mathcal{I}} \cap \{d \in \Delta^{\mathcal{I}} \mid \forall p, (d, p) \in \texttt{Sender}^{\mathcal{I}} \rightarrow p \in \texttt{CarVendor}^{\mathcal{I}}\}.$$

In addition to that of primitive concepts and roles, we assume an alphabet $\mathcal{O}$ of symbols called *individuals*, denoted by a and b, and an alphabet $\mathcal{V}$ of *variables* (denoted by x and y). The alphabet of *objects*, written $\mathcal{O}^+$, is $\mathcal{O}^+ = \mathcal{O} \cup \mathcal{V}$, (objects are denoted by o and o').

[1]Parentheses are used only to disambiguate concept expressions. For example, we will write $(\forall R.C) \sqcap D$ to mean that the concept D is not in the scope of $\forall R$.

Objects are related to concepts and roles via individual assertions. For instance, the assertion `Order(o12)` says that the individual `o12` is an `Order`, while `Sender(o12,cv1)` says that the sender of `o12` is `cv1`. Another type of assertions assess the subset relationship between concept extensions; for instance, `Ferrari` $\sqsubseteq$ `SportsCar`, asserts that the class of `Ferrari` is a subclass of the class `SportsCar`. Formally, an *assertion* is an expression having one of the following forms:

- $C(o)$, meaning that o is an instance of C, where o is an object and C is a $\mathcal{ALC}$MIR concept;
- $R(o, o')$, meaning that o is related to o' by means of R, where o and o' are objects and R is a $\mathcal{ALC}$MIR role;
- $T \sqsubseteq T'$, meaning that T is a subclass of T', where T and T' are both concepts or both roles.

An assertion made out of a primitive symbol is called *primitive assertion*. An assertion made out of a negated primitive symbol is called *negated primitive assertion*.

Semantically, interpretations map objects into domain elements and may or may not satisfy assertions, depending on the conditions spelled out in Table 2. For instance, the interpretation $\mathcal{I}$ satisfies the individual assertion $C(o)$ just in case the domain element onto which o is mapped by $\mathcal{I}$, $o^{\mathcal{I}}$, is in the extension of C in $\mathcal{I}$, $C^{\mathcal{I}}$.

A *document base* (DB) is a finite set of assertions. A *query* is an assertion of type $C(o)$. Given a DB Σ and a query $Q(o)$, the decision problem in TLs which is relevant to the present context is instance checking, *i.e.* whether every interpretation satisfying all the assertions in Σ also satisfies $Q(o)$. This is just another way of saying that $Q(o)$ is a logical consequence of Σ, written $\Sigma \models Q(o)$. For example, it is easily verified that `CarVendor(v1)` is a logical consequence of $\{(\texttt{Order} \sqcap \forall \texttt{Sender}.\texttt{CarVendor})(\texttt{o1}), \texttt{Sender}(\texttt{o1}, \texttt{v1})\}$. The toy DB Σ_1, which will be used throughout this paper, is given in Table 3. Σ_1 describes three documents, the `Orders` `o1, o2` (indexed by the index terms `f40, red` and `c`) and the `Invoice i`. The domain knowledge in Σ_1 states some connections about roles and concepts; for instance,

$$\texttt{IndTermList} \circ \texttt{IndTerm} \sqsubseteq \texttt{About}$$

means that if document d has a term list l and the token t is indexed in l ($\texttt{InTermList}(d, l)$ and $\texttt{IndTerm}(l, t)$) then d is about t ($\texttt{About}(d, t)$). As it can be easily verified,

$$\Sigma_1 \models \texttt{Doc} \sqcap \exists \texttt{About}.\texttt{Ferrari}(\texttt{o1}).$$

3 Relevance semantics for $\mathcal{ALC}$MIR

This section presents the semantics of $\mathcal{ALC}$MIR, and is divided in three main parts. The first one provides the rationale for the successive technical developments, discussing, at a pre-theoretic, intuitive level, the connection between IR and

Assertion	Satisfiability in an interpretation $\mathcal{I}$
$C(o)$	$o^{\mathcal{I}} \in C^{\mathcal{I}}$
$R(o, o')$	$(o^{\mathcal{I}}, o'^{\mathcal{I}}) \in R^{\mathcal{I}}$
$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
$R \sqsubseteq Q$	$R^{\mathcal{I}} \subseteq Q^{\mathcal{I}}$

Table 2: Syntax and semantics of assertions.

About Order o1
Order(o1), Sender(o1,v1), CarVendor(v1), IndTermList(o1,l1), IndexList(l1), IndTerm(l1,f40), IndTerm(l1,red), Ferrari(f40), Red(red), HasImage(o1,i2), Image⊓∃(Thereis.Car)(i2) Comp(o1,t1), Comp(o1,t2), Transl(t1,t2), Transl(t2,t3), ItalianText(t1), ¬ItalianText(t3)
About Order o2
Order(o2), Sender(o2,v2), CarVendor(v2), IndTermList(o2,l2), IndexList(l2), Car(c), HasImage(o2,i1), IndTerm(l2,c), (Image ⊓∃(Thereis.Lamborghini ⊓∃Color.Green))(i1)
About Invoice i
Invoice(i), RelatedTo(i,o1), RelatedTo(i,o2),
Domain Knowledge Base
Order ⊑ Doc, Invoice ⊑ Doc, IndexList ⊑ ∀IndTerm.Token, Ferrari ⊑ SportsCar, Lamborghini ⊑ SportsCar, SportsCar ⊑ Car, ¬ItalianText ⊑ EnglishText, CarVendor ⊑ Reseller, PrivateVendor ⊑ ¬Reseller, IndTermList∘IndTerm ⊑ About, About∘RelatedTo ⊑ About, HasImage∘Thereis ⊑ About, Sender ⊑ From, RelatedTo∘From ⊑ From

Table 3: A simple document base

relevance logics[2]. The second part formally specifies the semantics, while the third one presents the main properties of the resulting implication relation, to the end of showing its compliance with the driving intuition.

3.1 IR and relevance logics

MIR is often described in terms of *relevance*: its task is to find, among the documents in a given collection, those that are relevant to a given query. Unfortunately, the primacy of relevance in the whole IR discipline is also the primary cause that has hindered, up to now, the development of a *theory* of IR. In fact, relevance is not a formally and clearly defined notion; what relevance is, in other words, is defined by the user from time to time and from experiment to experiment, and is then heavily dependent on judgments where highly subjective and scarcely reproducible factors are brought to bear. The very possibility of a theory of IR is then dependent on the possibility of giving a *formal* definition of what relevance is, a definition capable of abstracting from the subjective and contingent factors inherent in the *operational* view of relevance described above.

Some works (see e.g. [van Rijsbergen, 1989]) have thus addressed the foundational problem of IR by trying to give a formal notion of relevance based on mathematical logic. These researches have shown how the relevance of a document d to a query q may naturally be understood in terms of a *conditional* (sometimes also called an *implication*) $d \rightarrow q$, where the "$\rightarrow$" symbol is the particular conditional notion formalized by a given logic. The foundational problem has then become the problem of singling out the logic (or those logics) whose conditional takes into account "relevance" as a critical factor.

The history of logic has seen a flurry of logics motivated by the need to give a natural account of the conditional. Classical logic itself possesses a well-known conditional notion, *material implication* (denoted by the symbol "$\supset$"). However, material implication has often been criticized, on the account that it licenses paradoxical sentences as theorems of the pure calculus; for instance, the sentence $(a \supset (b \supset a))$ (asserting that a true proposition is implied by any proposition) is a theorem of classical logic, a state of affairs that is questionable at best. It is interesting to note that some of the paradoxical sentences belonging to classical logic (and modal logics, too) are actually conditional sentences that suffer from *fallacies of relevance*: in other words, they are theorems of the given logic *even if their premise is not relevant to their conclusion*. For instance, the fact that $(a \supset (b \supset a))$ is valid in classical logic should strike one as peculiar, in that in any of these cases the fact that b holds does not have any "relevance" to the fact that a holds!

Among the first to take such a stand, Nelson [Nelson, 1933] has argued that, in order for any conditional notion "$\rightarrow$" to be adequate, a sentence such as $a \rightarrow b$ should be valid only if there be "some connection of meaning between a and b". To the surprise of many orthodox logicians, the idea of a "connection of meaning between a and b" (or, more generally, the idea of a being *relevant* to b) has been shown to be amenable to formal treatment by a number of logicians who have defined a class of logical calculi called *relevance* or *relevant logics* [Anderson and Belnap, 1975].

Relevance logics attempt to formalize a conditional notion in which relevance is a primary concern. By doing this, they challenge classical logic in a number of ways, i.e. by introducing a new, non truth-functional connective (denoted by "$\rightarrow$") into the syntactic apparatus of classical logic, by rejecting some classical rules of inference for classical connectives, and by changing the notion of validity itself by "wiring" into it considerations of relevance.

The rationale of the present TM model is that relevance logics are a very valuable source of inspiration for logics of information retrieval. In fact, even a brief analysis of the motivations put forth by relevance logicians and by IR theorists, respectively, indicates a surprising coincidence of underlying tenets and purposes. Therefore, it seems just natural to think that, if we view retrieval as essentially consisting of a disguised form of logical inference [van Rijsbergen, 1989], IR and relevance logic might constitute the engineering side and the theoretical side of the same coin.

As with modal logics, there are many relevance logics; some of them are ordered with respect to expressive power, while some of them are incommensurable with respect to this dimension; more importantly, different relevance logics formalize a different notion of relevance. The relevance logic that seems to comply with the requirements of the IR world is the logic $\mathbf{E_{fde}}$ of *tautological entailment* [Dunn, 1976], the fragment of the relevance logics $\mathbf{E}$ and $\mathbf{R}$ that deals with *first degree entailment* only, i.e. pairs of propositional (classical) formulae separated by one "$\rightarrow$" symbol. This logic seems well suited to formalize a state of affairs in which both document and query have a boolean representation, and in which the relevance of one to the other is the parameter of interest. In addition, $\mathbf{E_{fde}}$, has a denotational semantics, namely a four-valued semantics, independently developed by Belnap [Belnap, 1975] and Dunn [Dunn, 1976], which make it amenable to the various extensions needed for modelling IR. Finally, the computational properties of $\mathbf{E_{fde}}$ have been investigated. While deciding entailment in the general case is likely to be intractable (technically, the problem is co-NP-complete [Patel-Schneider, 1987b]), whenever α and β are

[2]We thank Fabrizio Sebastiani for his contribution to this section.

formulae in Conjunctive Normal Form, there exists a $O(|\alpha| \cdot |\beta|)$ algorithm that tests if $\alpha \rightarrow \beta$ holds [Levesque, 1984]. *Relevance TLs*, *i.e.* four-valued TLs based on some relevance logic, have already been used in Knowledge Representation and proven to have a generally better computational behavior than their two-valued analogues [Levesque, 1984; Patel-Schneider, 1987a; Patel-Schneider, 1989].

For all these reasons, we view tautological entailment as a major source of inspiration for incorporating a form of relevance into the inference relation of $\mathcal{ALC}$MIR. In particular, the four-valued semantics of $\mathcal{ALC}$MIR is a variation of tautological entailment, in which the implication relation has been strengthened to better cope with the reasoning tasks of MIR.

3.2 Semantics

In a relevance TL, assertions can be not only *true* or *false* in an interpretation, but also neither true nor false (a state of affairs which is known as *unknown*), and also both true and false (a state of affairs which is known as *contradiction*). Formally, the four truth values are the elements of the powerset of $\{t, f\}$, *i.e.* $\{t, f\}$, $\{\}$, $\{t\}$ and $\{f\}$, and are best understood as epistemic states of a reasoning system. Under this view, if the truth value of a proposition contains t, then the system has evidence to the effect – or believes – that the proposition is true. Similarly, if the truth value of a proposition contains f, then the system believes that the proposition is false. The truth value $\{\}$ corresponds to lack of knowledge, and the truth value $\{t, f\}$ corresponds to inconsistent knowledge.

Definition 1 *An* interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *consists of a non empty set* $\Delta^{\mathcal{I}}$ *(the* domain *of* $\mathcal{I}$*) and a function* $\cdot^{\mathcal{I}}$ *(the* interpretation function *of* $\mathcal{I}$*) such that*

1. $\cdot^{\mathcal{I}}$ *maps every concept into a function from* $\Delta^{\mathcal{I}}$ *to the powerset of* $\{t, f\}$*;*
2. $\cdot^{\mathcal{I}}$ *maps every role into a function from* $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ *to the powerset of* $\{t, f\}$*;*
3. $\cdot^{\mathcal{I}}$ *maps every object into* $\Delta^{\mathcal{I}}$*;*
4. $a^{\mathcal{I}} \neq b^{\mathcal{I}}$*, if* $a \neq b$*.* ∎

The interpretation function can best be understood as the union of two separate two-valued extensions: the positive extension and the negative extension. Let $\mathcal{I}$ be an interpretation. The *positive extension* of a concept C, written $C^{\mathcal{I}}_+$, is defined as the set $\{d \in \Delta^{\mathcal{I}} : t \in C^{\mathcal{I}}(d)\}$, whereas the *negative extension* of a concept C, written $C^{\mathcal{I}}_-$,is defined as the set $\{d \in \Delta^{\mathcal{I}} : f \in C^{\mathcal{I}}(d)\}$. The positive and negative extension of roles are defined similarly.

A *two-valued standard interpretation* is an interpretation $\mathcal{I}$ such that for every concept C, $C^{\mathcal{I}}_- = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}_+$ and for all roles R, $R^{\mathcal{I}}_- = \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}}_+$.

Unlike standard semantics, the positive and the negative extension of the same concept need not be the complement of each other. Domain elements that are members of neither set are not known to belong to the concept and are not known not to belong to the concept. This is a perfectly reasonable state for a system that is not a perfect reasoner or does not have complete information. Domain elements that are members of both sets can be thought of as inconsistent with respect to that concept in that there is evidence to indicate that they are in the extension of the concept and, at the same time, not in the extension of the concept. This is a slightly harder state to rationalize but can be considered a possibility in the light of inconsistent information.

The extensions of concepts and roles have to meet certain restrictions, designed so that the formal semantics respects the informal meaning of constructors. For example, the positive extension of the concept C⊓D must be the intersection of the positive extension of C and D and its negative extension must be the union of their negative extensions, thus formalizing the intuitive notion of conjunction in the context of the four-valued semantics.

Definition 2 *Let* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *be an interpretation. The interpretation function* $\cdot^{\mathcal{I}}$ *has to meet the following equations for concepts and roles: for each* $d, d' \in \Delta^{\mathcal{I}}$

$$
\begin{array}{rll}
t \in (C \sqcap D)^{\mathcal{I}}(d) & \textit{iff} & t \in C^{\mathcal{I}}(d) \textit{ and } t \in D^{\mathcal{I}}(d) \\
f \in (C \sqcap D)^{\mathcal{I}}(d) & \textit{iff} & f \in C^{\mathcal{I}}(d) \textit{ or } f \in D^{\mathcal{I}}(d) \\
t \in (C \sqcup D)^{\mathcal{I}}(d) & \textit{iff} & t \in C^{\mathcal{I}}(d) \textit{ or } t \in D^{\mathcal{I}}(d) \\
f \in (C \sqcup D)^{\mathcal{I}}(d) & \textit{iff} & f \in C^{\mathcal{I}}(d) \textit{ and } f \in D^{\mathcal{I}}(d) \\
t \in (\neg C)^{\mathcal{I}}(d) & \textit{iff} & f \in C^{\mathcal{I}}(d) \\
f \in (\neg C)^{\mathcal{I}}(d) & \textit{iff} & t \in C^{\mathcal{I}}(d) \\
t \in (\forall R.C)^{\mathcal{I}}(d) & \textit{iff} & \forall\, e \in \Delta^{\mathcal{I}}, \\
 & & t \in R^{\mathcal{I}}(d, e) \textit{ implies } t \in C^{\mathcal{I}}(e) \\
f \in (\forall R.C)^{\mathcal{I}}(d) & \textit{iff} & \exists\, e \in \Delta^{\mathcal{I}}, \\
 & & t \in R^{\mathcal{I}}(d, e) \textit{ and } f \in C^{\mathcal{I}}(e) \\
t \in (\exists R.C)^{\mathcal{I}}(d) & \textit{iff} & \exists\, e \in \Delta^{\mathcal{I}}, \\
 & & t \in R^{\mathcal{I}}(d, e) \textit{ and } t \in C^{\mathcal{I}}(e) \\
f \in (\exists R.C)^{\mathcal{I}}(d) & \textit{iff} & \forall\, e \in \Delta^{\mathcal{I}}, \\
 & & t \in R^{\mathcal{I}}(d, e) \textit{ implies } f \in C^{\mathcal{I}}(e) \\
t \in (R \circ Q)^{\mathcal{I}}(d, d') & \textit{iff} & \exists\, e \in \Delta^{\mathcal{I}}, \\
 & & t \in R^{\mathcal{I}}(d, e) \textit{ and } t \in Q^{\mathcal{I}}(e, d') \\
f \in (R \circ Q)^{\mathcal{I}}(d, d') & \textit{iff} & \forall\, e \in \Delta^{\mathcal{I}}, \\
 & & f \in R^{\mathcal{I}}(d, e) \textit{ or } f \in Q^{\mathcal{I}}(e, d')
\end{array}
$$

Moreover, $\top^{\mathcal{I}}_+ = \bot^{\mathcal{I}}_- = \Delta$ *and* $\top^{\mathcal{I}}_- = \bot^{\mathcal{I}}_+ = \emptyset$*.* ∎

The intuition behind this definition is to guarantee that, for example, $(C \sqcap D)^{\mathcal{I}}_+ = C^{\mathcal{I}}_+ \cap D^{\mathcal{I}}_+$, *i.e.* an object is known to be an instance of $C \sqcap D$ iff it is known to be an instance of both concepts C and D, and $(C \sqcap D)^{\mathcal{I}}_- = C^{\mathcal{I}}_- \cup D^{\mathcal{I}}_-$, *i.e.* an object is known not to be an instance of $C \sqcap D$ iff it is known not to be an instance of one of the concepts C and D. Similarly, $(C \sqcup D)^{\mathcal{I}}_+ = C^{\mathcal{I}}_+ \cup D^{\mathcal{I}}_+$ and $(C \sqcup D)^{\mathcal{I}}_- = C^{\mathcal{I}}_- \cap D^{\mathcal{I}}_-$. Moreover, note that in accordance with our intuition $(\exists R.C)^{\mathcal{I}}_+ = (\neg\forall R.\neg C)^{\mathcal{I}}_+$ and $(\exists R.C)^{\mathcal{I}}_- = (\neg\forall R.\neg C)^{\mathcal{I}}_-$.

Given two $\mathcal{ALC}$MIR concepts C and D, C is *equivalent* to D, written $C \equiv D$, iff $C^{\mathcal{I}}_+ = D^{\mathcal{I}}_+$, for every interpretation $\mathcal{I}$. $\equiv$ is extended to roles in a straightforward way.

An interpretation $\mathcal{I}$ *satisfies an assertion* α iff $t \in C^{\mathcal{I}}(o^{\mathcal{I}})$ if $\alpha = C(o)$, $t \in R^{\mathcal{I}}(o^{\mathcal{I}}, o'^{\mathcal{I}})$ if $\alpha = R(o, o')$, $T^{\mathcal{I}}_+ \subseteq T'^{\mathcal{I}}_+$ if $\alpha = T \sqsubseteq T'$. Finally, $\mathcal{I}$ *satisfies* (is a *model* of) a DB Σ iff $\mathcal{I}$ satisfies all assertions in Σ.

Definition 3 *A document base* Σ entails *a query* Q*, written* $\Sigma \models_4 Q$*, if and only if all models of* Σ *satisfy* Q*.* ∎

3.3 Properties of the semantics

In this section we will discuss the most relevant features of the just introduced semantics, to the end of characterizing the notion of relevance captured by $\models_4$. As it will be clear, $\models_4$ is a special case of the classical implication relation (denoted as "$\models_2$") licensing only a mild form of *modus ponens*

while avoiding the paradoxes of logical implication and reasoning by case. A more detailed account of what follows can be found in [Meghini, 1996]. All the examples refer to the DB Σ_1 defined in Section 2.

First of all, because by definition a two-valued interpretation is a four-valued interpretation, $\Sigma \models_4 Q$ implies $\Sigma \models_2 Q$. This guarantees the soundness of the entailment relation, an important requirement if the semantics is to capture some of the intuitive ideas underlying TLs.

Modus ponens on roles

It can be easily verified that:

$$\Sigma_1 \models_4 \texttt{Token}(\texttt{f40})$$

i.e. `f40` is a `Token`. In fact, `l1` is an `IndexList`, all `IndexLists` are such that all their `IndTerms` are `Tokens`, `f40` is an indexed term wrt `l1` and therefore `f40` is a `Token`. This inference is an instance of the following schema, which we call *modus ponens on roles* (MPR, for short): for all concepts C and D, for any role R, and for all individuals a, b :

$$\{(\forall R.C)(a), R(a,b)\} \models_4 C(b) \qquad \text{and}$$
$$\{(\forall R.C)(a), (\exists R.D)(a)\} \models_4 (\exists R.C \sqcap D)(a).$$

MPR is not allowed by other four-valued TLs, as, for example, in [Patel-Schneider, 1987a]). We claim that MPR is very useful for MIR and, in general, for real problems, and therefore we provided it in our framework.

Paradoxes of logical implication

It is well known that the paradoxes of logical implication, *i.e.* a contradictory DB implies everything and a tautology is implied by every DB, do not hold in propositional tautological entailment [Anderson and Belnap, 1975; Levesque, 1984]. This property is shared by our semantics, too. In fact, consider the DB Σ_1' given by:

$$\Sigma_1 \cup \{\texttt{PrivateVendor}(\texttt{v1}), \texttt{Order}(\texttt{o3})\}.$$

Σ_1' has an inconsistency (in classical terms) about `v1`, yet it is satisfiable and we have both:

$$\Sigma_1' \models_4 (\texttt{Order} \sqcap \exists \texttt{Sender}.\texttt{Reseller})(\texttt{o1})$$
$$\Sigma_1' \models_4 (\texttt{Order} \sqcap \exists \texttt{Sender}.\neg\texttt{Reseller})(\texttt{o1}).$$

The entailment holds in both cases since in Σ_1' there is *evidence* to the fact that `v1` is an instance of both `Reseller` and $\neg$`Reseller`. On the other hand, Σ_1' knows nothing about `o3` sender, therefore, as expected,

$$\Sigma_1' \not\models_4 \exists \texttt{Sender}.\texttt{CarVendor}(\texttt{o3}).$$

In two-valued semantics, since Σ_1' is inconsistent:

$$\Sigma_1' \models_2 \exists \texttt{Sender}.\texttt{CarVendor}(\texttt{o3}).$$

This example shows a pleasant side-effect of our four-valued semantics: classically inconsistent DBs do not entail everything. This property is of paramount importance to IR, because it is unrealistic to assume consistency among content representations of different documents.

Dually, concepts whose extensions are always the entire domain of interpretation, are not necessarily entailed by every DB. For instance:

$$\Sigma_1' \not\models_4 (\forall \texttt{Sender}.(\texttt{CarVendor} \sqcup \neg\texttt{CarVendor}))(\texttt{o3}),$$

since there is an interpretation $\mathcal{I}$ satisfying Σ_1', such that $e \in \Delta^{\mathcal{I}}$ and $t \in \texttt{Sender}^{\mathcal{I}}(\texttt{o3}, e)$ and $\texttt{CarVendor}^{\mathcal{I}}(e) = \emptyset$. This is a state of affairs which models the fact that in Σ_1' there is no evidence about `o3` sender's, *whatever* they could be. On the contrary,

$$\Sigma_1' \models_2 (\forall \texttt{Sender}.(\texttt{CarVendor} \sqcup \neg\texttt{CarVendor}))(\texttt{o3}).$$

To our opinion, missing this last kind of inference is important for MIR purposes, since we want relevance of the premises to the conclusion.

Reasoning by cases

Another inference schema that is not licensed by our semantics is *reasoning by cases.* Consider the assertion :

$$\alpha = (\exists \texttt{Comp}.(\texttt{ItalianText} \sqcap \exists \texttt{Transl}.\texttt{EnglishText}))(\texttt{o1}).$$

Let $\mathcal{I}$ be a model of Σ_1 such that $\texttt{ItalianText}^{\mathcal{I}}(\texttt{t2}^{\mathcal{I}}) = \emptyset$. It is easy to see that such an interpretation exists. Now, it follows that

$$t \notin (\exists \texttt{Comp}.(\texttt{ItalianText} \sqcap \exists \texttt{Transl}.\texttt{EnglishText}))^{\mathcal{I}}(\texttt{o1}^{\mathcal{I}})$$

and thus $\Sigma_1 \not\models_4 \alpha$. However, reasoning by case analysis, one realizes that $\Sigma_1 \models_2 \alpha$. Consider any two-valued interpretation $\mathcal{I}$ which satisfies Σ_1. In this interpretation either `ItalianText(t2)` is true or `ItalianText(t2)` is false. In the former case `o1` has `t2` as a component which is an Italian text and whose English text translation is `t3`. In the latter case `o1` has `t1` as a component which is an Italian text and whose English text translation is `t2`. Therefore, in both cases α is true in $\mathcal{I}$ and, thus, $\Sigma_1 \models_2 \alpha$.

We feel that dispensing the above inference is a way of capturing relevance, since the DB has no information about `t2`'s text language.

4 Individual and predicate closures in $\mathcal{AL}$MIR

The main idea behind individual and primitive closures can be explained by considering the document base Σ_1. Suppose that all the positive information about the order `o1` is that contained in Σ_1, *i.e.* `o1` has an unique `Sender` `v1`, there is exactly one relative `IndTermList` (nothing else), the only components are `t1` and `t2`, etc. Contrary to expectations, this information is not sufficient to infer, for example, that all `o1`'s senders are resellers, or that `o1` is not an invoice. These inferences, as well as all the others that our intuition would judge as correct, can be obtained only when `o1` description is completed by entering all the negative facts involving it into the DB. For example, to this end, at least the indexer should tell the DB an assertion $\neg A(\texttt{o1})$ for all primitive concepts `o1` is not an instance of. It is not difficult to see that this amounts to an enormous number of negative individual assertions.

In order to overcome this problem, we would like to have the linguistic machinery to declare that the knowledge on a certain individual, such as $\texttt{o}_1$, is complete, and an inference relation able to correctly interpret such a declaration by applying a closed-world reasoning on the involved individual. Formally, here we envisage an entailment relation, be it $\models_{mir}$, that, considering a DB such as Σ_1 and an assertion such as `CL(o1)`, would grant the following inferences:

$$(\Sigma 1, \{\texttt{Cl}(\texttt{o1})\}) \models_{mir} (\texttt{Doc} \sqcap \forall \texttt{Sender}.\texttt{Reseller})(\texttt{o1})$$
$$(\Sigma 1, \{\texttt{Cl}(\texttt{o1})\}) \models_{mir} \neg\texttt{Invoice}(\texttt{o1}).$$

In a somewhat dual way, one would like to tell a DB that, for instance, all the index lists are only those *known* to be `IndexList`s, without bothering to specify $\neg$`IndexList`(a) for all the very many individuals that are not index lists. This would correspond to tell the DB another kind of assertion, involving the primitive concept `From`, thereby having the inference:

$$(\Sigma 1, \{\texttt{Cl(From)}\}) \models_{mir} (\texttt{Invoice} \sqcap \forall \texttt{From.CarVendor})(\texttt{i}).$$

The newly introduced assertions give the indexer the possibility of specifying meta-information, regarding the way in which the information on certain individuals and primitives is to be considered. More precisely, while the lack of information on normal individual/primitives is to be interpreted, in the usual way, as evidence of the incompleteness of the representation, the lack of information on individuals/primitives subject to closures is to be interpreted as evidence to the contrary.

4.1 Syntax

Let a be an individual and let T be a primitive concept or role. An *individual closure* is an expression of type `Cl`(a). The individual a is said to be *closed*. A *primitive closure* is an expression of type `Cl`(T). The term T is said to be *closed*. A *CBox* is a finite set of closures. An $\mathcal{ALC}$`MIR` document base is extended to be a pair (Σ, Ω), where Ω is a CBox.

4.2 Semantics

Let Δ be the *domain,* a countably infinite set of symbols, called *parameters* (denoted by p and p') and γ a fixed injective function from $\mathcal{O}$ to Δ. Let $\mathcal{I}$ be a four-valued interpretation. A *c-interpretation* $\mathcal{I}$ is a four-valued interpretation such that:

1. $\Delta^{\mathcal{I}} = \Delta$;
2. for all individuals a, $a^{\mathcal{I}} = \gamma(a)$.

The notion of satisfaction of assertions is extended to c-interpretations in the obvious way. Unless otherwise specified, in the following "interpretation" means "c-interpretation". $\mathcal{M}(\Sigma)$ indicates the set of all models of Σ.

Satisfaction of closures is defined on the basis of a notion of minimal knowledge, modelled by epistemic interpretations. An *epistemic interpretation* is a pair $(\mathcal{I}, \mathcal{W})$ where $\mathcal{I}$ is an interpretation and $\mathcal{W}$ is a set of interpretations.

Definition 4 *An epistemic interpretation* satisfies *a closure* `Cl`(a) *if and only if the following conditions hold:*

1. *for every primitive concept symbol A, $t \in A^{\mathcal{I}}(\gamma(a))$ iff $t \in A^{\mathcal{J}}(\gamma(a))$ for all $\mathcal{J} \in \mathcal{W}$;*
2. *for every primitive concept symbol A, $f \in A^{\mathcal{I}}(\gamma(a))$ iff $t \notin A^{\mathcal{J}}(\gamma(a))$ for some $\mathcal{J} \in \mathcal{W}$;*
3. *for every primitive role symbol P and parameter $p \in \Delta$, $t \in P^{\mathcal{I}}(\gamma(a), p)$ iff $t \in P^{\mathcal{J}}(\gamma(a), p)$ for all $\mathcal{J} \in \mathcal{W}$;*
4. *for every primitive role symbol P and parameter $p \in \Delta$, $f \in P^{\mathcal{I}}(\gamma(a), p)$ iff $t \notin P^{\mathcal{J}}(\gamma(a), p)$ for some $\mathcal{J} \in \mathcal{W}$.* ∎

In words, for any model of a document base (Σ, Ω) and closed individual a, $a^{\mathcal{I}}$ is allowed in the positive extension of a primitive concept A just in case $A(a)$ is entailed by Σ, in symbols $\Sigma \models_4 A(a)$. As a consequence, the lack of positive information will allow us, as will be soon shown, to infer the corresponding negative information. Similarly for roles. The semantics of primitive assertions is perfectly dual; it constrains the extensions of closed primitive concepts and roles with respect to parameters.

Definition 5 *An epistemic interpretation* satisfies *a closure* `Cl`(A)*, where A is a primitive concept, if and only if the following conditions hold:*

1. *for every parameter $p \in \Delta$, $t \in A^{\mathcal{I}}(p)$ iff $t \in A^{\mathcal{J}}(p)$ for all $\mathcal{J} \in \mathcal{W}$;*
2. *for every parameter $p \in \Delta$, $f \in A^{\mathcal{I}}(p)$ iff $t \notin A^{\mathcal{J}}(p)$ for some $\mathcal{J} \in \mathcal{W}$;*

An epistemic interpretation satisfies *a closure* `Cl`(P)*, where P is a primitive role, if and only if the following conditions hold:*

1. *for all parameters $p, p' \in \Delta$, $t \in P^{\mathcal{I}}(p, p')$ iff $t \in P^{\mathcal{J}}(p, p')$ for all $\mathcal{J} \in \mathcal{W}$;*
2. *for all parameters $p, p' \in \Delta$, $f \in P^{\mathcal{I}}(p, p')$ iff $t \notin P^{\mathcal{J}}(p, p')$ for some $\mathcal{J} \in \mathcal{W}$.* ∎

An epistemic interpretation *satisfies* (is a *model* of) a set of closures if and only if it satisfies each closure in the set.

Definition 6 *Let (Σ, Ω) be a document base. An interpretation $\mathcal{I}$* satisfies *(is a* model *of) (Σ, Ω) if and only if $\mathcal{I}$ is a model of Σ and $(\mathcal{I}, \mathcal{M}(\Sigma))$ is a model of Ω.* ∎

Essentially, in order to be a model of a DB, an interpretation has to satisfy the "normal" assertions in Σ and the requirements imposed by closures, given in the previous two definitions. Finally,

Definition 7 *A document base (Σ, Ω)* c-entails *a query Q, written $(\Sigma, \Omega) \models_{mir} Q$, if and only if all models of (Σ, Ω) satisfy Q.* ∎

4.3 Properties of closures

Let us consider the document base (Σ_1, Ω_1), where Ω_1 is the set of closures

$$\{\texttt{Cl(o1)}, \texttt{Cl(IndexList)}, \texttt{Cl(IndTermList)}, \texttt{Cl(RelatedTo)}, \texttt{Cl(From)}\},$$

and the following queries:

1. Q_1 := `Doc` $\sqcap \forall$`Sender.Reseller`, *i.e.* the documents whose senders are all resellers;
2. Q_2 := `Invoice` $\sqcap \forall$`From.CarVendor`, *i.e.* the invoices which are all originated by a car vendor;
3. Q_3 := `Doc` $\sqcap \exists$`About.`(`SportsCar` $\sqcap \exists$`Color.Green`), *i.e.* the documents which are about green sports cars;
4. Q_4 := `Doc` $\sqcap \forall$`About.Car`, *i.e.* the documents which are only about cars.

Thanks to the closures of o1, in all the models of (Σ_1, Ω_1), $\mathtt{o1}^{\mathcal{I}}$ only belongs to the positive extension of Order and, as first member of a pair, to that of Sender, with $\mathtt{v1}^{\mathcal{I}}$ as a second member. This means that in all the models of (Σ_1, Ω_1), $\mathtt{v1}^{\mathcal{I}}$ is the only parameter to be in the positive extension of Sender as a second member of a pair whose first element is $\mathtt{o1}^{\mathcal{I}}$; moreover, in all the models of (Σ_1, Ω_1), $\mathtt{v1}^{\mathcal{I}}$ is in the positive extension of Reseller, due to the assertions CarVendor(v1) and CarVendor$\sqsubseteq$Reseller in Σ_1; it follows that all the models of (Σ_1, Ω_1) satisfy $\forall$Sender.Reseller(o1), therefore, as desired (see previous Section):

$$(\Sigma_1, \Omega_1) \models_{mir} Q_1(\mathtt{o1}).$$

Similarly, the effect of closing the primitive From is that, in any model $\mathcal{I}$ of (Σ_1, Ω_1), the positive extension of From, $\mathtt{From}^{\mathcal{I}}_{+}$, is given by those pairs which are *known to be* instances of From, *i.e.* $\mathtt{From}^{\mathcal{I}}_{+}$ is the set:

$$\{(\mathtt{o1}^{\mathcal{I}}, \mathtt{v1}^{\mathcal{I}}), (\mathtt{o2}^{\mathcal{I}}, \mathtt{v2}^{\mathcal{I}}), (\mathtt{i}^{\mathcal{I}}, \mathtt{v1}^{\mathcal{I}}), (\mathtt{i}^{\mathcal{I}}, \mathtt{v2}^{\mathcal{I}})\}$$

and therefore, as desired,

$$(\Sigma_1, \Omega_1) \models_{mir} Q_2(\mathtt{i}).$$

The following c-entailment shows how information on a image can be used.

$$(\Sigma_1, \Omega_1) \models_{mir} Q_3(\mathtt{i}).$$

Note that, since the primitive About is *not* closed, *i.e.* About is interpreted in a open world view:

$$(\Sigma_1, \Omega_1) \not\models_{mir} Q_4(\mathtt{o2}).$$

On the other hand if we consider also Cl(About), then

$$(\Sigma_1, \Omega_1) \models_{mir} Q_4(\mathtt{o2}).$$

A formal investigation of the features of closures follows. A concept C is said to be *quantifier free* if no quantifier occurs in it. Moreover, a document base is called *completely closed* if all individuals appearing in it are closed.

In classical logic, a theory is said to be complete if, for any sentence α, either α or its negation follows from the theory. The next two theorems show that closing an individual or a primitive amounts to make the knowledge about it complete in the classical sense. Since an assertion containing a quantifier involves also other individuals, a proviso is required in the first part of the next theorem. The second part shows that, when all the individuals are closed, the proviso is no longer needed.

Theorem 1 *Let (Σ, Ω) be a document base, $C(a)$ a concept assertion and $\mathtt{Cl}(a) \in \Omega$. Then:*

1. *for any quantifier free concept C, either $(\Sigma, \Omega) \models_{mir} C(a)$ or $(\Sigma, \Omega) \models_{mir} \neg C(a)$;*
2. *if (Σ, Ω) is completely closed, then for any C, either $(\Sigma, \Omega) \models_{mir} C(a)$ or $(\Sigma, \Omega) \models_{mir} \neg C(a)$.* ∎

For closed terms we have:

Theorem 2 *Let (Σ, Ω) be a document base. Then if $\mathtt{Cl}(A) \in \Omega$ then, for all individuals a, either $(\Sigma, \Omega) \models_{mir} A(a)$ or $(\Sigma, \Omega) \models_{mir} \neg A(a)$.* ∎

It is natural to ask how c-entailment relates to entailment. The answer to this question comes in two steps. First, c-entailment extends entailment, that is $\models_4 \subset \models_{mir}$.

Theorem 3 *Let (Σ, Ω) be a document base and $C(a)$ an assertion. Then $\Sigma \models_4 C(a)$ implies $(\Sigma, \Omega) \models_{mir} C(a)$.* ∎

In order to show that $\models_4 \neq \models_{mir}$, it suffices to consider the DB (Σ_1, Ω_1) defined at the beginning of this Section. As we have seen, $\Sigma_1 \not\models_4 Q_1(o1)$, whereas $(\Sigma_1, \Omega_1) \models_{mir} Q_1(o1)$.

Second, c-entailment captures a form of Closed-World Assumption (CWA): a positive assertion is c-entailed if it is entailed, while a negative assertion is c-entailed if the corresponding positive assertion is not entailed. Also the converse holds, provided that the DB is satisfiable, because, as it follows from the semantics of closures, a closed individual can only be associated with the classical truth values ($\{t\}$ and $\{f\}$), hence on closed terms the DB behaves as a classical theory. The next theorem formalizes this fact, showing exactly what is the inferential gain of c-entailment over classical entailment.

Theorem 4 *Let (Σ, Ω) be a document base. Then*

1. *if $\mathtt{Cl}(a) \in \Omega$ then for each primitive concept A,*
   - *(a) $\Sigma \models_4 A(a)$ implies $(\Sigma, \Omega) \models_{mir} A(a)$;*
   - *(b) $\Sigma \not\models_4 A(a)$ implies $(\Sigma, \Omega) \models_{mir} \neg A(a)$.*

   Conversely, if (Σ, Ω) is satisfiable, then for each primitive concept A,
   - *(c) $(\Sigma, \Omega) \models_{mir} A(a)$ implies $\Sigma \models_4 A(a)$;*
   - *(d) $(\Sigma, \Omega) \models_{mir} \neg A(a)$ implies $\Sigma \not\models_4 A(a)$.*
2. *if $\mathtt{Cl}(A) \in \Omega$ then for all individuals a,*
   - *(a) $\Sigma \models_4 A(a)$ implies $(\Sigma, \Omega) \models_{mir} A(a)$;*
   - *(b) $\Sigma \not\models_4 A(a)$ implies $(\Sigma, \Omega) \models_{mir} \neg A(a)$.*

   Conversely, if (Σ, Ω) is satisfiable, then for each primitive concept A,
   - *(c) $(\Sigma, \Omega) \models_{mir} A(a)$ implies $\Sigma \models_4 A(a)$;*
   - *(d) $(\Sigma, \Omega) \models_{mir} \neg A(a)$ implies $\Sigma \not\models_4 A(a)$.*

Similarly for closed roles. ∎

In fact, part 1a of the last Theorem is a special case of Theorem 3 and it has been stated in this form only for symmetry.

Theorem 4 gives us the possibility of comparing our model with Naive CWA, historically the first notion of CWA to be proposed. Naive CWA is defined for finite sets of first-order sentences without equality and whose prenex normal forms contain no existential quantifiers. If T is one such sets, then the naive closure of T, *NCWA(T),* is given by [Lukaszewicz, 1990]:

$$NCWA(T) = T \cup \{\neg A : T \not\models A \text{ and } A \in HB(T)\},$$

where $HB(T)$ is the Herbrand Base of T. Now, the first-order translation of a set of $\mathcal{ALC}$MIR assertions yields a set of sentences with the existential quantificator. If we apply the NCWA operator to this kind of theories, the last Theorem tells us that c-entailment on completely closed DBs is equivalent to Naive CWA for the corresponding first-order theories. This is because we are considering theories with no function symbols, hence the terminological correspondent of $HB(T)$ is the set of primitive assertions.

It is worth noting that there is a big methodological difference between our approach and NCWA, or, for that matter, all other approaches with the same goal, as for example in Datalog[Fuhr, 1995]: in $\mathcal{AL}$MIR, CWA is not something happening *behind the scene*, but is explicitly called upon, via closures, by the document indexer, who has therefore full control of the situation, and is free to apply CWA only on specified terms.

5 Retrieval in $\mathcal{AL}$MIR

Retrieval of documents in $\mathcal{AL}$MIR DBs is performed via a new, Gentzen-like sequent calculus. There are several reasons why we developed our own retrieval engine. In the first place, none of the existing engines is able to deal with our semantics and with closures. These engines can be divided into two categories according to the kind of semantics they support: those for four-valued logics, being non-modular, are not easily extensible to deal with $\mathcal{AL}$MIR; those for two-valued logics, being based on refutation, cannot be applied to languages which are not closed under negation. So we decided to develop an engine that: (1) preserves the modularity of two-valued engines, (2) is not based on refutation and (3) is able to deal with closures and the four-valued semantics. The calculus invented by Gentzen for first-order logic satisfies the first two requirements, and we have modified it to meet also the third one. A full account of the calculus is beyond the space limit of the present paper. In what follows, we will confine ourselves to a brief overview.

A *sequent* is an expression of the form $\gamma_1, \ldots, \gamma_n \rightarrow \delta_1, \ldots, \delta_m$ where the γ_i's (the antecedent) and the δ_j's (the succedent) are $\mathcal{AL}$MIR assertions, in which variables can occur wherever individuals can. An interpretation $\mathcal{I}$ satisfies $\gamma_1, \ldots, \gamma_n \rightarrow \delta_1, \ldots, \delta_m$ iff it satisfies some $\delta_1, \ldots, \delta_m$ if it satisfies all $\gamma_1, \ldots, \gamma_n$.

In order to establish that $(\Sigma, \Omega) \models_{mir} Q(a)$ we try to prove the validity of the sequent having Σ as antecedent and $Q(a)$ as succedent. The proof proceeds by successively transforming, by means of the *unfolding rules,* the original sequent into simpler, equivalent sequents until obvious truths, the *axioms*, are obtained. Since rules transform one sequent into one or two equivalent sequents, the proof is best understood as the construction of a binary tree, whose root is given by the sequent to be proved, and whose descendant relationship is the product of rule application.

Similarly to a natural deduction system, in our calculus there is a couple of rules for each construct of the language: a rule that simplifies the antecedent by removing the construct, and a rule that dually operates on the succedent. For example, the rule of the first kind for the $\sqsubseteq$ construct is the following:

$$(\sqsubseteq\rightarrow) \quad \frac{C \sqsubseteq D, \Gamma \rightarrow C(o), \Delta \quad C \sqsubseteq D, \Gamma, D(o) \rightarrow \Delta}{C \sqsubseteq D, \Gamma \rightarrow \Delta}$$

and the rule of the second kind for the $\sqcap$ operator is:

$$(\rightarrow \sqcap) \quad \frac{\Gamma \rightarrow \Delta, C(o) \quad \Gamma \rightarrow \Delta, D(o)}{\Gamma \rightarrow \Delta, (C \sqcap D)(o)}$$

As an example of rule application, let us consider the following, involving one of the examples of the previous Section[3]:

$$\dfrac{\dfrac{\vdots}{\Sigma_1 \rightarrow (\forall \mathtt{S.R})(\mathtt{o1})} \qquad \dfrac{\Sigma_1 \rightarrow \mathtt{O(o1)} \quad \Sigma_1, \mathtt{D(o1)} \rightarrow \mathtt{D(o1)}}{\Sigma_1 \rightarrow \mathtt{D(o1)}}}{\Sigma_1 \rightarrow Q_1(\mathtt{o1})}$$

The tree is shown upside down. In the first step, the $(\rightarrow \sqcap)$ is applied in order to remove the $\sqcap$ operator from the query; in the second one, the $(\sqsubseteq\rightarrow)$ is applied to the assertion $\mathtt{Order} \sqsubseteq \mathtt{Doc} \in \Sigma_1$, yielding two sequents whose succedents are also antecedents. Such sequents are obviously true in any interpretation (as a matter of fact, they are axioms) hence that branch of the tree is no longer expanded.

The axiom schemas of the calculus are:

1. $\alpha, \Gamma \rightarrow \alpha, \Delta$;
2. $\Gamma \rightarrow \top(o), \Delta$;
3. $\bot(o), \Gamma \rightarrow \Delta$;
4. $\Gamma \rightarrow \neg A(a), \Delta$ if $\mathtt{Cl}(a) \in \Omega$ or $\mathtt{Cl}(A) \in \Omega$, and $\Sigma \not\models_4 A(a)$.

The calculus is shown to be sound and complete in [Meghini, 1996]. Since it is guaranteed to terminate (what may not be evident from what we said so far), it proves the decidability of the c-entailment problem. The NP-hardness of the same problem can be shown by reducing to it an NP-complete problem, namely propositional entailment.

Theorem 5 *Let (Σ, Ω) be a document base and Q a query. Then the $\Sigma \rightarrow Q$ is provable if and only if $(\Sigma, \Omega) \models_{mir} Q$.* ∎

6 Conclusions

We have presented a model for multimedia information retrieval based on a terminological logic extended with closed-world assertions. The logic has a four-valued semantics that gives to its inference relation a flavor of relevance. The fundamental features of this model, which have been treated at the formal level and also informally illustrated via numerous examples, can be characterized by comparing its inference relation, c-entailment, with classical logical implication. The comparison is depicted in Figure 1. c-entailment ($\models_{mir}$) can be considered as the union of two sets of inferences: those licensed by entailment ($\models_4$), which are a strict subset of classical implication ($\models_2$), plus those allowed by closures ($\models_{cwa}$). In summary, we have given up part of *modus ponens,* thereby happily loosing the paradoxes of logical implication, while gaining closed-world reasoning *on specified individuals and primitive concepts or roles.*

Instance checking in $\mathcal{AL}$MIR can be proven to be decidable, more specifically, EXPTIME-hard [Meghini, 1996].

In order to perform document retrieval in the model, we have developed a sound and complete calculus, based on Gentzen calculus for first-order logic. For space reason, the calculus has been only surveyed, nevertheless it is important to observe that we are able to concretely use our logic in real applications.

[3] For reasons of space, S, R, O and D stand for Sender, Reseller, Order and Doc, respectively.

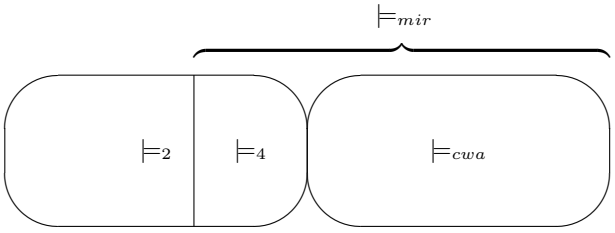

Figure 1: Pictorial characterization of c-implication

As a matter of fact, a prototypical implementation of the calculus is being developed in order to let us empirically verify the adequacy of the model. We plan to experiment $\mathcal{ALC}$MIR on a text collection, by having documents represented as individuals, related to a number of keywords (the documents contents) via a role. The power of the logic will be exploited by setting up a knowledge base containing domain knowledge on the concepts involved in queries, which will also be modelled as $\mathcal{ALC}$MIR concepts. In addition, we plan to test the model on an image collection, thus being able to exploit the expressive power of the logic also in the representation of image contents.

In parallel, we intend to extend the model with uncertainty management. The starting point to this end will be the logic $\mathcal{P}$-MIRTL [Sebastiani, 1994], a probabilistic TL.

Both the experimentations and the extension are on the agenda of the FERMI Project.

References

[Anderson and Belnap, 1975] A.R. Anderson and N.D. Belnap. *Entailment - the logic of relevance and necessity*. Princeton University Press, Princeton, NJ, 1975.

[Belnap, 1975] N.D. Belnap. How a computer should think. In *Contemporary Aspects of Philosophy: Proceedings of the Oxford International Symposium*, pages 30–56, Oxford, GB, 1975.

[Buchheit *et al.*, 1993] Martin Buchheit, Francesco M. Donini, and Andrea Schaerf. Decidable reasoning in terminological knowledge representation systems. *Journal of Artificial Intelligence Research*, 1:109–138, 1993.

[Buongarzoni *et al.*, 1995] Paolo Buongarzoni, Carlo Meghini, Rossella Salis, Fabrizio Sebastiani, and Umberto Straccia. Logical and computational properties of the description logic MIRTL. In Borgida A., Lenzerini M., Nardi D., and Nebel B., editors, *International Workshop on Description Logics*, pages 80–84, Rome, Italy, 1995.

[Dunn, 1976] J.M. Dunn. Intuitive semantics for first-degree entailment and coupled trees. *Philosophical studies*, 29:149–168, 1976.

[Fuhr, 1995] Norbert Fuhr. Probabilistic datalog - a logic for powerful retrieval methods. In *Proceedings of SIGIR-95, 18th International Conference on Research and Development in Information Retrieval*, pages 282–290, Seattle, WA, 1995.

[Levesque, 1984] Hector J. Levesque. A logic of implicit and explicit belief. In *Proceedings of AAAI-84, 4th Conference of the American Association for Artificial Intelligence*, pages 198–202, Austin, TX, 1984.

[Lukaszewicz, 1990] W. Lukaszewicz. *Non-Monotonic reasoning*, chapter 7: Approaches to Closed World Assumption, pages 281–308. Ellis Horwood series in Artificial Intelligence. Ellis Horwood, New York, 1990.

[Meghini *et al.*, 1993] C. Meghini, F. Sebastiani, U. Straccia, and C. Thanos. A model of information retrieval based on a terminological logic. In *Proceedings of SIGIR-93, 16th ACM Conference on Research and Development in Information Retrieval*, pages 298–307, Pittsburgh, PA, July 1993.

[Meghini, 1996] C. Meghini, editor. *A Logic for Information Retrieval*, Deliverable D2 of *ESPRIT Basic Research Action FERMI, n. 8134*. FERMI, Feb. 1996.

[Nelson, 1933] E.J. Nelson. On three logical principles in intension. *The Monist*, 43, 1933.

[Patel-Schneider, 1987a] Peter F. Patel-Schneider. A hybrid, decidable, logic-based knowledge representation system. *Computational Intelligence*, 3:64–77, 1987.

[Patel-Schneider, 1987b] P.F. Patel-Schneider. Decidable, logic-based knowledge representation. Technical Report 201/87, Department of Computer Science, University of Toronto, Toronto, Ontario, 1987.

[Patel-Schneider, 1989] Peter F. Patel-Schneider. A four-valued semantics for terminological logics. *Artificial Intelligence*, 38:319–351, 1989.

[Sebastiani, 1994] F. Sebastiani. A probabilistic terminological logic for modelling information retrieval. In *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 122–130, Dublin, IRL, 1994. Springer Verlag, Heidelberg.

[van Rijsbergen, 1989] C. J. van Rijsbergen. Towards an information logic. In *Proceedings of SIGIR-89, 12th Conference of the ACM Special Interest Group on Information Retrieval*, pages 77–86, Cambridge, MA, 1989.