Negation, Opposition, and Possibility in Logical Concept Analysis

Sébastien Ferré

Irisa/Université de Rennes 1, Campus de Beaulieu,
35042 Rennes cedex, France
ferre@irisa.fr

Abstract. We introduce the *epistemic extension*, a logic transformation based on the modal logic AIK (All I Know) for use in the framework of Logical Concept Analysis (LCA). The aim is to allow for the distinction between negation, opposition, and possibility in a unique formalism. The difference between negation and opposition is examplified by the difference between "young/not young" and "young/old". The difference between negation and possibility is examplified by the difference between "(certainly) not young" and "possibly not young". Furthermore this epistemic extension entails no loss of genericity in LCA.

1 Introduction

Many have felt the need to extend Formal Concept Analysis in various ways: valued attributes, partially ordered attributes, graph patterns, 3-valued contexts, distinction between negation and opposition, etc. Often a new and specific solution was proposed: conceptual and logical scales [Pre97], 3-valued contexts [Obi02], two different negation operators [Wil00, Kan05], etc.

We have introduced *Logical Concept Analysis* [FR00] in order to have a framework that could cover as many extensions as possible by simply changing the logic used to describe objects. Other authors have proposed similar frameworks where they talk about partial orderings of patterns instead of logics [CM00, GK01]. Indeed our logics can be seen as partial orderings, but we emphasize the use of the term logic as this brings useful notions like the distinction between syntax and semantics, reasoning through entailment or subsumption, and important properties like consistency and completeness. Moreover this makes available the numerous existing logics, and the theory and practice that comes with them.

We have already applied LCA to various logics for querying and navigating in logical information systems [FR04] (e.g., combinations of string patterns, intervals over numbers and dates, custom taxonomies, functional programming types). We further support its capabilities by showing how an existing modal logic, AIK, can be used to represent at the same time *complete* and *incomplete* knowledge (Closed World Assumption and distinction beteeen *certain* and *possible* facts), and the distinction between *negation* and *opposition*. The result is something more expressive than existing extensions of FCA, because for each object one can express certain and possible facts, these facts being expressed in an almost arbitrary logic whose negation plays the role of opposition;

B. Ganter and L. Kwuida (Eds.): ICFCA 2006, LNAI 3874, pp. 130–145, 2006.

and three levels of negation allow for distinguishing certainly/possibly true/false normal/opposite properties of objects. For example it becomes possible to distinguish between "possibly not young" (possibility), "certainly not young" (usual negation), and "old", i.e., "opposite of young" (opposition). All these distinctions are quite important when querying a context.

Section 2 presents useful preliminaries about Logical Concept Analysis (LCA), and the modal logic All I Know (AIK). Section 3 explains the ambiguity that lies in the interpretation of negation. Section 4 then gives a solution based on the logic AIK for distinguishing (usual) negation and opposition. Section 5 goes further in the use of the logic AIK in order to distinguish certainty and possibility. In Section 6 the logic AIK is replaced by a logic transformation in order to retain the genericity of LCA, and to simplify the use of AIK. Finally we compare our solution to related works in Section 7, before concluding.

2 Preliminaries

This paper is based on Logical Concept Analysis (LCA) where the logic is a modal logic called AIK (All I Know). As preliminaries, we first recall some definitions of LCA, and differences to FCA. Then we introduce the logic AIK with an emphasis on its semantics as it is crucial in the rest of the paper.

2.1 Logical Concept Analysis

Logical Concept Analysis (LCA) [FR00] has been introduced in order to allow for richer object descriptions, and to bring in automated reasoning when deciding whether an object belongs to a concept or not. The principle is to replace sets of attributes (object descriptions and intents) by the formulas of an "almost arbitrary logic". The kind of logics we consider is defined as follows.

Definition 1 (logic). *A* logic *(in LCA) is made of*

- *a* syntax *or* language*, i.e., a set L of formulas,*
- *a set of* operations *like conjunction ($\sqcap$, binary), disjunction ($\sqcup$, binary), negation ($\neg$, unary), tautology ($\top$, nullary), and contradiction ($\bot$, nullary),*
- *a* semantics*, i.e., a set of interpretations I, and a binary relation $\models$ ("is a model of") between interpretations and formulas,*
- *a* subsumption *relation $\sqsubseteq$, which decides whether a formula is* subsumed *("is more specific/less general") than another formula. Its intended meaning is that $f \sqsubseteq g$ iff every model of f is also a model of g.*

We also define $M(f)$ as a shorthand for $\{i \in I \mid i \models f\}$, i.e., the set of models of the formula f.

This definition is quite large, and covers most existing languages and semantics in logic. There are two differences with the usual presentation of logics. Firstly, logical operations (e.g., conjunction, negation) are not necessarily connectors in the language, which makes them more general. For example, if formulas are intervals, then the conjunction is the intersection on intervals. Secondly, the left argument of the entailment relation (subsumption) is restricted to a single formula instead of a set of formulas. This makes subsumption a generalization ordering, hence its name, which is crucial for use in concept analysis.

In LCA it is possible to look at logics only as partial orderings, and to forget about semantics. However semantics plays an important role when defining and reasoning about logics, like in this paper. This is why we introduce it above, and in the following we define the relation we usually expect between logical operations, subsumption, and semantics, i.e. consistency and completeness.

Definition 2 (consistency and completeness). *An operation of some logic $\mathcal{L}$ is* consistent and complete *if its expected meaning agrees with semantics, i.e., for all formulas $f, g \in \mathcal{L}$:*
1. $M(f \sqcap g) = M(f) \cap M(g)$
2. $M(f \sqcup g) = M(f) \cup M(g)$
3. $M(\neg f) = I \setminus M(f)$
4. $M(\top) = I$
5. $M(\bot) = \emptyset$
6. $f \sqsubseteq g \iff M(f) \subseteq M(g)$

From these definitions a *logical context* can now be defined, where a logic replaces a set of attributes. However remind that logical formulas replace sets of attributes, and not single attributes, as is demonstrated by the mapping from objects to formulas that replaces the binary relation.

Definition 3 (logical context). *A* logical context *is a triple $K = (\mathcal{O}, \mathcal{L}, d)$, where:*

- *$\mathcal{O}$ is a finite set of objects,*
- *$\mathcal{L}$ is a logic equiped with at least disjunction $\sqcup$, contradiction $\bot$, and subsumption $\sqsubseteq$ as consistent and complete operations. Because these 3 operations are consistent and complete, this logic can equivalently be seen as a 0-sup-semilattice, whose ordering is $\sqsubseteq$, bottom is $\bot$, and join operation is $\sqcup$.*
- *$d \in \mathcal{O} \rightarrow L$ maps each object $o \in \mathcal{O}$ to a single formula $d(o) \in \mathcal{L}$, its logical description.*

This definition is a slight weakening of the original definition [FR00] (conjunction and tautology operations are no more required as they have no consequence on the concept lattice); and it is similar to the definition of a *pattern structure* [GK01] (except disjunction is denoted by $\sqcap$ instead of $\sqcup$, and no logical interpretation is given[1]). The disjunction $\sqcup$ plays the role of set intersection $\cap$ in FCA, which is normal as a consistent and complete disjunction returns the most specific common subsumer of two formulas. So, in the case formulas are sets of attributes, disjunction is defined as $\cap$; and in the case formulas are Prolog terms, disjunction is defined as anti-unification [Plo70].

[1] In logic, disjunction is usually denoted by $\vee$ or $\sqcup$.

Galois connections can then be defined on logical contexts, enabling us to go from formulas to sets of objects (their *extent*), and from sets of objects to formulas (their *intent*).

Lemma 1. *Let $K = (\mathcal{O}, \mathcal{L}, d)$ be a logical context. The pair (ext, int), defined by*

$$\begin{array}{ll} ext(f) = \{o \in \mathcal{O} \mid d(o) \sqsubseteq f\} & \text{for every } f \in L \\ int(O) = \bigsqcup\{d(o) \mid o \in O\} & \text{for every } O \subseteq \mathcal{O} \end{array}$$

is a Galois connection between $\mathcal{P}(\mathcal{O})$ and L: $O \subseteq ext(f) \iff int(O) \sqsubseteq f$.

From there, concepts, the concept lattice, and its labelling are defined as usual, only replacing sets of attributes by formulas where necessary. However, we do not detail them in this paper as we here focus on expressing and distinguishing negation, opposition, and possibility in formulas and in the subsumption relation, which determines extents, which determine in turn the concept lattice.

2.2 The Logic All I Know

In computer science, epistemic aspects in logic have often been discussed under the term *Closed World Assumption* (CWA). CWA says that every fact that cannot be deduced from a knowledge base can be considered as false, contrary to the open world assumption, which says that such a fact is neither true, nor false. This often leads to the notion of *non-monotonous reasoning*, i.e., the addition of knowledge can make false a fact that was previously true by CWA.

There exists several formalisms for handling CWA, but all of them except the logic All I Know (AIK) are non-monotonous [Lif91, DNR97, Moo85, McC86, Lev90]. However logics to be used in the framework of Logical Concept Analysis (LCA) must form at least a partial ordering, and so, must have a monotonous subsumption relation. In fact, CWA should be applied locally on the formulas used to describe objects rather than globally on the knowledge base. The logic AIK precisely defines such an operation, a modal operator O. Moreover it has been established that this logic covers all non-monotonous formalisms cited above (see [Che94] for correspondences between these formalisms), and a proof method exists for its propositional version [Ros00].

The logic AIK [Lev90] is essentially a modal logic [Bow79], to which the two modal operators N and O have been added[2]. The formula language of AIK is defined as the propositional language *Prop*, whose atomic propositions belong to an infinite set *Atom*, and connectors are 1 (tautology), 0 (contradiction), $\wedge$ (conjunction), $\vee$ (disjunction), and $\neg$ (negation). It is extended by the modal operators K, N and O. The five logical operations ($\sqcap$, $\sqcup$, $\neg$, $\top$, $\bot$) are simply realized by the connectors of the language (resp. $\wedge$, $\vee$, $\neg$, 1, 0).

A Kripke semantics is given to AIK. Given a set of *worlds* W, interpretations are couples (w, R), where $w \in W$ is a world, and $R \subseteq W \times W$ is an *accessibility*

[2] The basic modal operator is K. Sometimes modal logics are presented as having the two modal operators $\Box$ and $\Diamond$, which are in fact equivalent to respectively K and $\neg K \neg$.

relation between worlds. More precisely, each world defines a valuation of atoms in *Atom* by boolean values $\{TRUE, FALSE\}$. The "is a model of" relation between interpretations and formulas is then defined as follows.

Definition 4 (semantics). *Let w be a world, and let R be an accessibility relation that is both* transitive[3] *and* euclidean[4]. *A Kripke structure (w, R) is a* model *of a formula $\phi \in AIK$, which is denoted by $(w, R) \models \phi$, iff the following conditions are satisfied ($R(w)$ denotes the set of successor worlds of w through R):*

1. $(w, R) \models a$	*iff*	$w(a) = TRUE$, *where* $a \in Atom$;
2. $(w, R) \models 1$	*iff*	*true, i.e., for every* (w, R) ;
3. $(w, R) \models 0$	*iff*	*false, i.e., for no* (w, R) ;
4. $(w, R) \models \neg\phi_1$	*iff*	$(w, R) \not\models \phi_1$;
5. $(w, R) \models \phi_1 \wedge \phi_2$	*iff*	$(w, R) \models \phi_1$ *and* $(w, R) \models \phi_2$;
6. $(w, R) \models \phi_1 \vee \phi_2$	*iff*	$(w, R) \models \phi_1$ *or* $(w, R) \models \phi_2$;
7. $(w, R) \models K\phi_1$	*iff*	*for every* $w' \in R(w)$, $(w', R) \models \phi_1$;
8. $(w, R) \models N\phi_1$	*iff*	*for every* $w' \notin R(w)$, $(w', R) \models \phi_1$;
9. $(w, R) \models O\phi_1$	*iff*	*for every* w', $w' \in R(w)$ *iff* $(w', R) \models \phi_1$.

The logic AIK can be equiped with a subsumption relation, which enables us to compare object descriptions and queries, for instance in the definition of extents. An axiomatization of AIK [Lev90] provides a consistent and complete algorithm for computing the subsumption relation $\sqsubseteq_{AIK}$. The logic $\mathcal{L}_{AIK} = (AIK, \wedge, \vee, \neg, 1, 0, \sqsubseteq_{AIK})$ can then be defined according to Definition 1. Given its semantics defined above, this logic also forms a complete lattice, whose ordering is $\sqsubseteq_{AIK}$, meet and join operations are respectively $\wedge$ and $\vee$, and top and bottom are respectively 1 and 0. This makes it applicable to LCA. In order to ease the understanding of modal operators, we provide the following lemma (a proof is available in [Fer02], p. 86).

Lemma 2. *If $\phi \in AIK$ and $W_R(\phi) = \{w | (w, R) \models \phi\}$ is the set of worlds where ϕ is true, then for every structure (w, R):*
1. $(w, R) \models K\phi$ *iff* $R(w) \subseteq W_R(\phi)$;
2. $(w, R) \models N\neg\phi$ *iff* $R(w) \supseteq W_R(\phi)$;
3. $(w, R) \models O\phi$ *iff* $R(w) = W_R(\phi)$.

This lemma shows that what counts in a model (w, R) of a modal formula is neither the initial world w, nor the accessibility relation itself, but the set of successor worlds $R(w)$. So modal formulas $M\phi$ ($M \in \{K, N\neg, O\}$) can be interpreted as sets of models of ϕ, rather than individual models of ϕ. For instance, the modal formula $K\phi$ represents some subsets of $W_R(\phi)$, in which at least ϕ, but not only ϕ is always true: $K\phi$ can be read "at least ϕ". Dually, the modal formula $N\neg\phi$ can be read "at most ϕ", and the modal formula $O\phi$, which is semantically equivalent to $K\phi \wedge N\neg\phi$ from definition 4, can be read "exactly ϕ" or "all I know is ϕ" (hence the name AIK [Lev90]).

[3] A relation R is transitive iff $\forall w, w', w'' : wRw'$ and $w'Rw''$ implies wRw''.

[4] A relation R is euclidean iff $\forall w, w', w'' : wRw'$ and wRw'' implies $w'Rw''$.

3 Ambiguity in the Interpretation of Negation

In this section we exhibit an ambiguity that lies in the interpretation of negation, when it is available in the logic. Practically, in information retrieval, the problem is that we get unsatisfactory answers (i.e., extents) to queries with negation. In order to show the problem more concretely, let us consider a logical context $K = (\mathcal{O}, \mathcal{L}_{Prop}, d)$, where $\mathcal{L}_{Prop}$ is the propositional logic (the same as AIK but with no modal operators). Let say objects are persons, and atoms are properties of these persons (e.g., young, rich). Describing objects then consists in expressing the knowledge one has about person properties. Let say one person of the context, Alice, is young, unhappy, and rich or smart. This can be represented by giving to object Alice the description $young \land \neg happy \land (rich \lor smart)$. Such a context can exhibit two "anomalies" w.r.t. our intuition, for every queries $q, q' \in Prop$:

1. $ext(q) \cup ext(\neg q) \subsetneqq \mathcal{O}$: an object can satisfy neither a query, nor its negation. Example: Alice is not an answer of any of the queries *tall* and $\neg tall$: so one cannot retrieve persons who are not known as tall.
2. $ext(q \lor q') \supsetneqq ext(q) \cup ext(q')$: an object can satisfy $q \lor q'$ while satisfying neither q, nor q'.
   Example: Alice is an answer to the query $rich \lor smart$, but neither to the query *rich*, nor *smart*: so one cannot retrieve persons who are either known as rich or known as smart.

These anomalies come from the fact that formulas, rather than interpretations, are used to describe objects. In relational databases and object-oriented databases, these anomalies do not exist because objects are described by sets of valued attributes letting no ambiguity in their interpretation. On the contrary, formulas generally do not have a unique interpretation but a set of models. When an object description has several models, there are always at least 2 of them that disagree w.r.t. a query q: one satisfies q while the other satisfies $\neg q$, which causes the anomaly 1. Similarly, even if all models satisfy $q \lor q'$, some may satisfy $\neg q$, and others $\neg q'$, which causes the anomaly 2. But interpretations cannot be used in practice because they are in general infinite, and we do not want to use them because they are not flexible enough. For example, in databases, they tend to force users to give a value to every attribute. With formulas, a finite set of relevant descriptors can be chosen among a large or infinite set (open language).

In fact, the expected interpretation of boolean operators in queries is generally *extensional*, i.e., logical operations on formulas (conjunction, disjunction, and negation) are expected to match set operations on extents (intersection, union, and complement); while they are fundamentally *intensional* in descriptions. In the intensional interpretation, negation can be understood as *opposition*, like the opposition that exists between "male" and "female": some things are neither male nor female. Now disjunction should be understood as *undetermination*, like knowing that something has a sex; one can know that something has a sex without knowing whether it is male or female.

Both kinds of negation are present in natural languages. In English, the grammatical word "not" is the extensional negation. Indeed everything is either happy or not happy, hot or not hot. The intensional negation is not so obvious in English as it can be realized either by various prefixes (as in happy/unhappy, legal/illegal), or by a totally different word (hot/cold, tall/small). However there are languages, like Esperanto [JH93], where a unique prefix (in Esperanto, mal-) is used to build all opposites, and thus becomes a grammatical element similar to a logical connector (e.g., varma/malvarma, alta/malalta). In the following, when translating formulas in English, we use this prefix "mal-" to build the opposites instead of the normal english word in order to make opposition more visible (as it is in the logical language with negation): e.g., tall/mal-tall.

Our objective is not to choose between extensional and intensional interpretations, but to combine both in a same formalism. This requires distinguishing occurences of negation as extensional or intensional. Considering a single property "young" and only negation, we obtain 4 different queries: "young", "not young", "mal-young" (i.e., "old"), and "not mal-young" (i.e., "not old"). The logic should also recognize the two subsumption relations that exist between these formulas: "young" entails "not mal-young", and "mal-young" entails "not young" (non-contradiction law). Finally, only opposition is relevant in descriptions because it should be enough not to say a person is young so as to retrieve this person from the query "not young". This is known as the *Closed World Assumption* (CWA).

4 Distinguishing Negation and Opposition

As said in the introduction, the logic AIK (Section 2.2) enables us to apply the Closed World Assumption (CWA) on individual object descriptions through the modality O ("exactly"). The principle is to apply this modality on each object description of a context $K = (\mathcal{O}, \mathcal{L}_{Prop}, d)$, so as to obtain the context

$$K^1 =_{def} (\mathcal{O}, \mathcal{L}_{AIK}, d^1), \text{such that } d^1(o) =_{def} O(d(o)), \text{ for all } o \in \mathcal{O}.$$

For example the description of Alice in previous section becomes $O(young \land \neg happy \land (rich \lor smart))$.

In queries we propose to use the modality K ("at least"), and we claim that negation has a different interpretation whether it is inside or outside the scope of this modality: $\neg K(young)$ means "is not young", while $K(\neg young)$ means "is mal-young". So boolean operations are extensional outside modalities, and intensional inside modalities. We do not consider the use of the modality O in queries as it is unlikely that a user would ask for an object being "exactly q".

We now show that the anomalies exhibited in Section 3 are solved, i.e., extensional operations match set operations on extents (see proof in [Fer02], p. 88).

Theorem 1. *Let $q, q' \in Prop$ be propositional formulas.*

1. $ext_{K^1}(\neg K(q)) = \mathcal{O} \setminus ext_{K^1}(K(q))$;
2. $ext_{K^1}(K(q) \lor K(q')) = ext_{K^1}(K(q)) \cup ext_{K^1}(K(q'))$;
3. $ext_{K^1}(K(q) \land K(q')) = ext_{K^1}(K(q)) \cap ext_{K^1}(K(q'))$;

These equalities are not verified when negation and disjunction appear inside modalities. For example Alice is neither in the extent of $K(tall)$, nor in the extent of $K(\neg tall)$, but she is in the extent of both $\neg K(tall)$ and $\neg K(\neg tall)$ ("Alice is neither tall nor small"). Furthermore, Alice is in the extent of $K(rich \vee smart)$ but neither in the extent of $K(rich)$ nor in the extent of $K(smart)$ ("Alice is rich or smart but we do not know which"). This confirms that operations inside modalities are intensional, whereas operations outside are extensional. Finally it can be proved in AIK that for every propositional formula q:

- $K(q) \sqsubseteq_{AIK} \neg K(\neg q)$ ("is q" entails "is not mal-q"),
- $K(\neg q) \sqsubseteq_{AIK} \neg K(q)$ ("is mal-q" entails "is not q").

This shows there is a hierarchy between both negations; opposition is more specific than extensional negation.

5 Distinguishing Certainty and Possibility

In previous section we applied the Closed World Assumption (CWA) on object descriptions so that everything not true in a description is considered as false. For example, Alice is now considered as "not tall" ($d^1(Alice) \sqsubseteq_{AIK} \neg K(tall)$) because her description is not subsumed by "tall" ($d(Alice) \not\sqsubseteq tall$). This assumes we have a *complete knowledge* about objects. However we sometimes have only *incomplete knowledge* about objects. In this case, some property that is not true in a description may still be *possible*. In the example, as the description of Alice is subsumed neither by "tall" nor by "small" ("mal-tall"), Alice may be everything among "tall", "mal-tall", and "neither tall nor mal-tall".

Between these two extreme positions, a range of 5 intermediate positions can be imagined, from the most incomplete to the most complete:

1. Alice is young and unhappy, and may have any other property;
2. Bob is young and unhappy, and may be rich and smart;
3. Charlie is young and unhappy, and may be either rich or smart but not both;
4. David is young and unhappy, and may be rich;
5. Edward is young and unhappy (and has no other property).

Edward corresponds to previous section, where a property is either true or false. In other cases, we want to distinguish certain properties and possible properties. For example David is certainly young, possibly rich, and certainly not tall; whereas Edward is certainly not rich. In the rest of this section, we adapt the logic AIK and its use in order to distinguish and represent "(certainly) true", "possibly true", "possibly false", and "(certainly) false".

5.1 First Solution with AIK

In a combination of certain and possible facts certain facts represent some kind of *minimum* of what is true, while possible facts represent a kind of *maximum*. Now Lemma 2 shows that the modalities K and $N\neg$ in AIK can be read respectively "at least" and "at most".

So we propose the following representation of above descriptions in AIK, before showing they are not fully satisfying:

1. $d(Alice) = K(young \wedge \neg happy) \wedge N\neg(0)$: at least young and unhappy, and at most everything;
2. $d(Bob) = K(young \wedge \neg happy) \wedge N\neg(young \wedge \neg happy \wedge rich \wedge smart)$: at least young and unhappy, and at most young, unhappy, rich, and smart;
3. $d(Charlie) = K(young \wedge \neg happy) \wedge (N\neg(young \wedge \neg happy \wedge rich) \vee N\neg(young \wedge \neg happy \wedge smart))$: at least young and unhappy, and at most young, unhappy, and rich, or young, unhappy, and smart;
4. $d(David) = K(young \wedge \neg happy) \wedge N\neg(young \wedge \neg happy \wedge rich)$: at least young and unhappy, and at most young, unhappy, and rich;
5. $d(Edward) = K(young \wedge \neg happy) \wedge N\neg(young \wedge \neg happy)$: at least and at most young and unhappy, i.e., exactly young and unhappy (what can be represented equivalently by $O(young \wedge \neg happy)$).

In order to verify whether these descriptions correspond to what we want, we draw the following table where the lines are the 5 descriptions, the columns are different queries q, and a cell is marked + if the description is subsumed by Kq, by − if it is subsumed by $\neg Kq$, and empty otherwise.

$\sqsubseteq_{AIK}$	$young$	$\neg young$	$\neg happy$	$rich$	$smart$	$rich \wedge smart$	$tall$
$d(Alice)$	+		+				
$d(Bob)$	+	−	+				−
$d(Charlie)$	+	−	+			−	−
$d(David)$	+	−	+		−	−	−
$d(Edward)$	+	−	+	−	−	−	−

If we read a query $K(X)$ as "X is (certainly) true", and a query $\neg K(X)$ as "X is (certainly) false", then the above table matches our expectations. The cells where neither $K(X)$, nor $\neg K(X)$ are satisfied can be read as "X is possibly true/false, but is not certainly true/false".

However descriptions $d(Alice)$ to $d(David)$ exhibit two problems. Firstly, objects *Alice* to *David* satisfy neither $K(rich)$, nor $\neg K(rich)$, what triggers the anomaly 1 (see Section 3) in spite of the presence of the modality K. This implies that possible facts cannot be represented, and so we cannot retrieve persons who "may be rich". Secondly, the query $\neg young$ is possible for *Alice* whereas *young* is true: i.e., "Alice is young, but may be old". This is obviously a contradiction. We handle these two problems by a generalization of the logic AIK [Fer01].

5.2 Generalization of AIK

As the first problem is similar to the problem in Section 3, it is tempting to apply the solution in Section 4, i.e., to encapsulate descriptions in the modality O, and queries in the modalities K and $\neg K$. Descriptions are then in the form

$$O(K(d_0) \wedge (N\neg(d_1) \vee \ldots \vee N\neg(d_n))), \text{ with } n \in \mathbb{N},\ \forall i \in 0..n : d_i \in Prop.$$

Unfortunately all formulas in this form have no model, and so are contradictions (see Example 2 of Section 2 in [Ros00]). So, AIK is not a direct answer to our problem, and we propose in the following two successive adaptations of AIK in order to solve it.

Similarly to the formula $O(young \land rich)$ that enables us to reason about the set of models of $young \land rich$ as if it would be a single model, we would like that the formula $O(K(young) \land N\neg(young \land rich))$ enables us to reason on the set of models of $K(young) \land N\neg(young \land rich)$, i.e., on the set of structures (w, R) such that $W_R(young \land rich) \subseteq R(w) \subseteq W_R(young)$ (see Lemma 2). To this end, it is necessary that the set $R(w)$ depends on the world w, so as to keep the multiplicity of interpretations of incomplete descriptions. This is why we propose to adapt the logic AIK by removing any condition on the accessiblity relation in Definition 4. The logic AIK can then be seen as an ordinary modal logic, where the modality K is defined on accessible worlds $R(w)$, while the modality N is defined on unaccessible worlds $W \setminus R(w)$, and the modality O is simply a combination of both ($O\phi =_{def} K\phi \land N\neg\phi$). Now a family of logics can be derived by applying various conditions on the accessibility relation, as this is already done for modal logics [Bow79]. For example, the usual logic AIK has a transitive and euclidean accessibility relation, and so could be renamed K45-AIK; while our adaptation has an arbitrary relation, and so could be nammed K-AIK.

Definition 5 (logic K-AIK). *The semantics of the logic K-AIK is defined as in Definitions 4, except there is no condition on the accessibility relation.*

In the logic K-AIK, knowledge is stratified because the accessibility relation is neither transitive nor euclidean. The object description $O(d(David)) = O(K(young \land \neg happy) \land N\neg(young \land \neg happy \land rich))$ is no more contradictory, and it can be read at three levels of knowledge: a model of

1. $young \land \neg happy$ is a world w'' satisfying the proposition $young \land \neg happy$;
2. $K(young \land \neg happy) \land N\neg(young \land \neg happy \land rich)$ is a world w' such that $R(w')$ is included in the set of models of $young \land \neg happy$, and contains all models of $young \land \neg happy \land rich$;
3. $O(K(young \land \neg happy) \land N\neg(young \land \neg happy \land rich))$ is a world w such that $R(w)$ is the set of all models of $K(young \land \neg happy) \land N\neg(young \land \neg happy \land rich)$.

So this description expresses the complete knowledge about the incomplete knowledge about the object *David*: "all I know about this person is that he is young and unhappy, and he may be rich as well". The logic K-AIK allows for the following subsumption relation.

Example. $O(d(David)) \sqsubseteq_{K-AIK}$
$$K(K(young)) \land K(\neg K(\neg young)) \land \neg K(K(rich)) \land \neg K(\neg K(rich)). \qquad \square$$

If the outermost K is read "I know", then we can translate the above entailment as "I know that David is young, and that he is not mal-young, but I do not know whether he is rich or not". So $K(K(q))$ represents (certainly) true facts, and $K(\neg K(q))$ represent (certainly) false facts. Formulas $\neg K(\neg K(q))$, which can be read as "I do not know that not q", represents possibly true facts, and similarly formulas $\neg K(K(q))$ represent possibly false facts. This solves the first problem about the representation of possible facts.

The first problem is now solved, but the second is not because $\neg$*young* ("mal-young") is judged as a possible fact for the object *Alice* whereas *young* is judged as true. This means there is a contradiction somewhere. In order to remove this contradiction, we try to exclude models (w', R) at the 2nd level of knowledge such that $R(w') = \emptyset$. This implies that in the description $d(Alice)$, the part $N\neg(0)$ does not mean "at most everything", but rather "at most everything provided it is not contradictory". Technically this is obtained by requiring the accessibility relation to be *serial*[5], which enforces every world to have at least one successor world: the result is the logic KD-AIK.

Definition 6 (logic KD-AIK). *The semantics of the logic KD-AIK is defined like in Definition 4, except the accessibility relation must be serial.*

This time we obtain the expected subsumption for *Alice*, with $\neg$*young* being judged as (certainly) false, whereas *rich* remains possibly true/false.

Example. $O(d(Alice)) \sqsubseteq_{KD-AIK}$
$$K(K(young)) \wedge K(\neg K(\neg young)) \wedge \neg K(K(rich)) \wedge \neg K(\neg K(rich)). \qquad \Box$$

In summary we use a version of the logic AIK (KD-AIK) with two levels of modalities in descriptions and queries. In descriptions the modality O is used at the outermost level, while a combination of modalities K and N is used in order to express certain and possible facts about object. In queries the modality K is used at both levels, which implies that negations can occur at three levels. From innermost to outermost levels the logical negation represents respectively *opposition*, *falsity*, and *possibility*. It is noticeable that in order to distinguish these three kinds of negations, there is need to introduce neither new connectors, nor special semantics. A small variation of the logic AIK, which is itself a small variation of well known modal logics, and appropriate combinations of modal operators are sufficient. In the following section we generalize the application of modal operators to other logics than propositional logic, and we introduce a compact and more intuitive syntax in place of combinations of modal operators and negations.

6 The Epistemic Extension of an Arbitrary Logic

Though the logic KD-AIK is suitable to the expression of opposition, negation, and possibility, it has the drawback of being applicable only to contexts whose facts are expressed in the propositional logic. So, we now define the epistemic

[5] A relation R is serial iff $\forall w : \exists w' : wRw'$.

extension $\mathcal{L}^2$ (2 levels of modalities) of an arbitrary logic $\mathcal{L} = (L, \sqcap, \sqcup, \neg, \top, \bot, \sqsubseteq)$ (among those applicable in LCA) so as to enable us to distinguish between opposition, falsity, and possibility.

The language $\mathcal{L}^2$ is defined by the following grammar, where L is the language of formulas of $\mathcal{L}$:

$$\begin{array}{rcl} L^2 & \longrightarrow & [L,\ L \shortmid \ldots \shortmid L] \\ & | & !L \\ & | & ?L \\ & | & L\ and\ L \mid L\ or\ L \mid not\ L \mid true \mid false. \end{array}$$

The correspondance of the first 3 derivations with AIK modalities is as follows:

- $[d_0,\ d_1 \shortmid \ldots \shortmid d_n] \longrightarrow O(K(d_0) \wedge (N\neg(d_0 \sqcap d_1) \vee \ldots \vee N\neg(d_0 \sqcap d_n)))$: d_0 represents all certain facts, while each $d_0 \sqcap d_i$ represents a maximum set of possible facts;
- $!q \longrightarrow K(K(q))$: q is *certainly* true;
- $?q \longrightarrow \neg K(\neg K(q))$: q is *possibly* true.

A complete knowledge where every possible fact is also certain has the form $[d_0, \top]$, which can be shortened as $[d_0]$. At the opposite, an incomplete knowledge where every non-certain fact is possible has the form $[d_0, \bot]$. Boolean operations have an extensional meaning, and enables us to express a *certainly false* fact q as a not possibly true fact, i.e., by the formula *not* $?q$, and a *strictly possibly true* fact q (i.e., possible but not certain) by the formula $?q$ *and not* $!q$. So *not* $!q$ must be read "not certainly q", i.e., "possibly not q". Similarly, the formula *not* $?q$ must be read "not possibly q", and is equivalent to "certainly not q" (i.e., "q is false"). These two equivalences are kinds of Morgan laws like those that exist between conjunction and disjunction, between universal and existential quantifiers, and between modal operators $\Box$ and $\Diamond$. Finally, opposition does not appear in the above grammar as it is played by the negation of $\mathcal{L}$ (when defined).

We now give the semantics of the language L^2 as a function of the semantics of L by using the Lemma 2. Moreover, as we use exactly 2 levels of modalities in object descriptions and queries, only the set $\{R(w') \mid w' \in R(w)\}$ semantically matters in a structure (w, R). So interpretations in L^2 are simply sets of sets of interpretations of L.

Definition 7 (semantics). *Let $S_L = (I, \models)$ be the semantics of a logic $\mathcal{L} = (L, \sqcap, \sqcup, \neg, \top, \bot, \sqsubseteq)$. The semantics $S_{L^2} = (I^2, \models^2)$ of the language L^2 is defined by (given that for all $f \in L$, $M_L(f) = \{i \in I \mid i \models f\}$):*

- $I^2 = 2^{2^I}$;
- *and for all* $i^2 \in I^2$,

$$i^2 \models^2 \begin{cases} [d_0, d_1 \mid \ldots \mid d_n] & \textit{iff} \quad i^2 = \{M \subseteq I \mid M \subseteq M_L(d_0), \\ & \qquad \exists k \in 1..n : M \supseteq M_L(d_0 \sqcap d_k)\} \\ !q & \textit{iff} \quad \forall M \in i^2 : M \subseteq M_L(q) \\ ?q & \textit{iff} \quad \exists M \in i^2 : M \subseteq M_L(q) \\ Q \textit{ and } Q' & \textit{iff} \quad i^2 \models^2 Q \textit{ and } i^2 \models^2 Q' \\ Q \textit{ or } Q' & \textit{iff} \quad i^2 \models^2 Q \textit{ or } i^2 \models^2 Q' \\ \textit{not } Q & \textit{iff} \quad i^2 \not\models^2 Q \\ \textit{true} & \textit{iff} \quad \textit{true, i.e., for all } i^2 \\ \textit{false} & \textit{iff} \quad \textit{false, i.e., for no } i^2 \end{cases}$$

We now have to verify that the anomalies in Section 3 are still solved, and that the semantics of possible facts is like expected. This is demonstrated by the following theorem.

Theorem 2. *Let $K = (\mathcal{O}, \mathcal{L}^2, d)$ be a context, and $d_i, d'_i, q \in \mathcal{L}$ and $Q, Q' \in \mathcal{L}^2$ be formulas. If the subsumption $\sqsubseteq$ of $\mathcal{L}$ is consistent and complete, the following equations are verified:*

1. $ext(!q) = \{o \in \mathcal{O} \mid d(o) = [d_0, d_1 \mid \ldots \mid d_n], d_0 \sqsubseteq q\}$
2. $ext(?q) = \{o \in \mathcal{O} \mid d(o) = [d_0, d_1 \mid \ldots \mid d_n], \exists k \in 1..n : (d_0 \sqcap d_k) \sqsubseteq q\}$
3. $ext(Q \textit{ and } Q') = ext(Q) \cap ext(Q')$
4. $ext(Q \textit{ or } Q') = ext(Q) \cup ext(Q')$
5. $ext(\textit{not } Q) = \mathcal{O} \setminus ext(Q)$
6. $ext(all) = \mathcal{O}$
7. $ext(none) = \emptyset$

Proof: The following proofs use Definition 7 of the semantics of $\mathcal{L}^2$, and use a lot the result that formulas in the form $d = [d_0, d_1 \mid \ldots \mid d_n]$ have a single model

$$m(d) =_{def} \{M \subseteq I \mid M \subseteq M_L(d_0), \exists k \in 1..n : M_L(d_0 \sqcap d_k) \subseteq M\}. \quad (1)$$

For conciseness, we omit to note the context K as indices of *ext*.

1. $o \in ext(q) \iff d(o) \sqsubseteq^2 q$
   $\iff \{m(d(o))\} \subseteq \{i^2 \in I^2 \mid \forall M \in i^2 : M \subseteq M_L(q)\}$
   $\iff \forall M \subseteq I : M \subseteq M_L(d_0)$
   and $(\exists k \in 1..n : M_L(d_0 \sqcap d_k) \subseteq M)$ implies $M \subseteq M_L(q)$ (1)
   $\iff d_0 \sqsubseteq q$: by contradiction, it is enough to consider the case where $M = M_L(d_0)$ knowing that $d_0 \sqcap d_k \sqsubseteq d_0$ for all k since $\mathcal{L}$ forms a lattice, and its subsumption is consistent and complete
2. $o \in ext(?q) \iff d(o) \sqsubseteq^2 ?q$
   $\iff \{m(d(o))\} \subseteq \{i^2 \in I^2 \mid \exists M \in i^2 : M \subseteq M_L(q)\}$
   $\iff \exists M \subseteq I : M \subseteq M_L(d_0)$
   and $(\exists k \in 1..n : M_L(d_0 \sqcap d_k) \subseteq M)$ et $M \subseteq M_L(q)$ (1)
   $\iff \exists k \in 1..n : M_L(d_0 \sqcap d_k) \subseteq M_L(d_0)$
   $\iff \exists k \in 1..n : d_0 \sqcap d_k \sqsubseteq d_0$. ($\sqsubseteq$ is consistent and complete)
3. immediate from the Galois connection.
4. $o \in ext(Q \textit{ or } Q') \iff d(o) \sqsubseteq^2 Q \textit{ or } Q'$
   $\iff \{m(d(o))\} \subseteq M_{L^2}(Q \textit{ or } Q') \iff \{m(d(o))\} \subseteq M_{L^2}(Q) \cup M_{L^2}(Q')$
   $\iff \{m(d(o))\} \subseteq M_{L^2}(Q)$ or $\{m(d(o))\} \subseteq M_{L^2}(Q')$
   $\iff d(o) \sqsubseteq^2 Q$ or $d(o) \sqsubseteq^2 Q' \iff o \in ext(Q) \cup ext(Q')$.

5. $o \in ext(not\ Q) \iff d(o) \sqsubseteq^2 not\ Q$
$\iff \{m(d(o))\} \subseteq M_{L^2}(not\ Q) \iff \{m(d(o))\} \not\subseteq M_{L^2}(Q)$
$\iff d(o) \not\sqsubseteq^2 Q \iff o \notin ext(Q)$.
6. immediate from the Galois connection.
7. $o \in ext(none) \Longrightarrow d(o) \sqsubseteq^2 none$
$\iff \{m(d(o))\} \subseteq \emptyset$: contradiction, hence $ext(none) = \emptyset$. ■

It can be verified that Theorem 2.(4) solves the anomaly 2, and that Theorem 2.(5) solves the anomaly 1. In fact operations in $\mathcal{L}^2$ (*and*, *or*, *not*, *true*, *false*) have a fully extensional interpretation as they match exactly set operations on context extents (intersection, union, complement, the set of all objects, the empty set). This theorem also generalizes Theorem 1 in two ways. Firstly, the logic used for describing objects and elementary queries is arbitrary (among those applicable to LCA), and not necessarily the propositional logic $\mathcal{L}_{Prop}$. Secondly, results apply to arbitrary queries in L^2: variables Q and Q' are not restricted to the language L.

Finally, this theorem gives us a way to compute the answers of queries in $\mathcal{L}^2$, provided we have a decision procedure for the subsumption $\sqsubseteq$ in $\mathcal{L}$. We have thus integrated epistemic knowledge in the querying process, and allowed distinction between opposition, negation, and possibility.

7 Related Work

The issue of representing incomplete knowledge, and distinguishing between certain and possible facts has already been studied in the scope of Formal Concept Analysis [Obi02, BH05]. They use 3-valued contexts which are similar to our distinction between true, possible, and false facts in object descriptions. They also use compound attributes, which are in fact formulas in the propositional logic. A first difference is that instead of having simple attributes we have formulas of an almost arbitrary logic in descriptions and queries, e.g., propositions over valued attributes. On the contrary they have only extensional operations, and no intensional operations like opposition. Another difference is that they have several derivation operators corresponding to the two modalities "certainly" and "possibly", which they respectively denote by $\Box$ and $\Diamond$, and which we denote in the epistemic extension (Section 6) by ! and ?. Instead we have only one derivation operator *ext* that applies to any combination of modalities, and both extensional and intensional operations. A more general difference is that they use contexts as the semantics of Contextual Logic whereas we use logics and their semantics right inside logical contexts.

The *certainly and possibly valid* implications defined by S. Obiedkov can be represented as follows in our framework:

– an implication $\phi \rightarrow \psi$ is *certainly valid* in a context K if every object that possibly satisfies ϕ certainly satisfies ψ, i.e., $ext_K(?\phi) \subseteq ext_K(!\psi)$,
– an implication $\phi \rightarrow \psi$ is *possibly valid* in a context K if every object that certainly satisfies ϕ possibly satisfies ψ, i.e., $ext_K(!\phi) \subseteq ext_K(?\psi)$.

Opposition has already been introduced as an alternative negation that respects the law of contradiction ($q \wedge \neg q$), but not the law of excluded middle ($q \vee \neg q$). Indeed, nobody can be young and old at the same time, but somebody can be neither young, nor old. In the scope of FCA, opposition appears first in the double Boolean algebra [Wil00]. It is also introduced in Description Logics through a new connector $\sim$, and an extended semantics [Kan05]. In both cases a new connector is introduced, whereas we simply use the classical connector in combination with modalities so as to obtain the logical properties of opposition.

8 Conclusion

To the best of our knowledge, our solution is the first that enables us to represent incomplete knowledge, and distinguish negation and opposition in a unique formalism. The logic used to describe objects is let free, and its negation (if defined) plays the role of opposition. Object descriptions contain certainly true facts (opposites or not), and a disjunction of possibly true facts. If a fact does not appear in a description it is considered as certainly false by CWA, which allows for concise descriptions (remind that the language of facts is often infinite). This is more expressive than 3-valued contexts as properties like "possibly either rich or smart" can be represented. Moreover the subsumption of the logic AIK recognizes the hierarchy that exists between opposition, negation, and possibility (from the most specific to the most general).

Surprisingly this result is achieved with no extension of the theory of LCA, and a small variation of the logic AIK that is common place in modal logics. The result comes from the right combination and interpretation of modal operators. The danger of losing the genericity of LCA is escaped by the *epistemic extension* of a logic that enables us to retain, and even extend its genericity. Indeed a detailed study about the combination of logics [FR02] shows that, w.r.t. their application in LCA, less properties are required on logics when this epistemic extension is applied on them. In other words, the epistemic extension ensures that a weak logic (having only consistent and partially complete subsumption) gains all desired properties for LCA.

The epistemic extension is implemented as part of a toolbox of logic components, LOGFUN (see `www.irisa.fr/lande/ferre/logfun`). It is systematically used in applications of CAMELIS, an implementation of logical information systems [FR04], which can handle efficiently up to several 10,000 objects (see `www.irisa.fr/lande/ferre/camelis`).

References

[BH05] P. Burmeister and R. Holzer. Treating incomplete knowledge in formal concept analysis. In B. Ganter, G. Stumme, and R. Wille, editors, *Formal Concept Analysis*, LNCS 3626, pages 114–126. Springer, 2005.

[Bow79] K. A. Bowen. *Model Theory for Modal Logic.* D. Reidel, London, 1979.

[Che94] J. Chen. The logic of only knowing as a unified framework for non-monotonic reasoning. *Fundamenta Informatica*, 21, 1994.

[CM00] L. Chaudron and N. Maille. Generalized formal concept analysis. In G. Mineau and B. Ganter, editors, *Int. Conf. Conceptual Structures*, LNCS 1867. Springer, 2000.

[DNR97] F. M. Donini, D. Nardi, and R. Rosati. Autoepistemic description logics. In *IJCAI*, 1997.

[Fer01] S. Ferré. Complete and incomplete knowledge in logical information systems. In S. Benferhat and P. Besnard, editors, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, LNCS 2143, pages 782–791. Springer, 2001.

[Fer02] S. Ferré. *Systèmes d'information logiques : un paradigme logico-contextuel pour interroger, naviguer et apprendre*. Thèse d'université, Université de Rennes 1, October 2002. Accessible en ligne à l'adresse http://www.irisa.fr/bibli/publi/theses/theses02.html.

[FR00] S. Ferré and O. Ridoux. A logical generalization of formal concept analysis. In G. Mineau and B. Ganter, editors, *Int. Conf. Conceptual Structures*, LNCS 1867, pages 371–384. Springer, 2000.

[FR02] S. Ferré and O. Ridoux. A framework for developing embeddable customized logics. In A. Pettorossi, editor, *Int. Work. Logic-based Program Synthesis and Transformation*, LNCS 2372, pages 191–215. Springer, 2002.

[FR04] S. Ferré and O. Ridoux. An introduction to logical information systems. *Information Processing & Management*, 40(3):383–419, 2004.

[GK01] B. Ganter and S. Kuznetsov. Pattern structures and their projections. In H. S. Delugach and G. Stumme, editors, *Int. Conf. Conceptual Structures*, LNCS 2120, pages 129–142. Springer, 2001.

[JH93] P. Janton and H.Tonkin. *Esperanto: Language, Literature, and Community.* State University of New York Press, 1993.

[Kan05] K. Kaneiwa. Negations in description logic - contraries, contradictories, and subcontraries. In F. Dau, M.-L. Mugnier, and G. Stumme, editors, *Contributions to ICCS 2005*, pages 66–79. Kassel University Press GmbH, 2005.

[Lev90] H. Levesque. All I know: a study in autoepistemic logic. *Artificial Intelligence*, 42(2), March 1990.

[Lif91] V. Lifschitz. Nonmonotonic databases and epistemic queries. In *12th International Joint Conference on Artificial Intelligence*, pages 381–386, 1991.

[McC86] J. McCarthy. Applications of circumscription to formalizing common sense knowledge. *Artificial Intelligence*, 28(1), 1986.

[Moo85] R. C. Moore. Semantical considerations on nonmonotonic logic. *Artificial Intelligence*, 25(1):75–94, 1985.

[Obi02] S. A. Obiedkov. Modal logic for evaluating formulas in incomplete contexts. In *ICCS*, LNCS 2393, pages 314–325. Springer, 2002.

[Plo70] G. D. Plotkin. A note on inductive generalization. *Machine Intelligence, Edinburgh Univ. Press*, 5:153–163, 1970.

[Pre97] S. Prediger. Logical scaling in formal concept analysis. *LNCS 1257*, pages 332–341, 1997.

[Ros00] R. Rosati. Tableau calculus for only knowing and knowing at most. In Roy Dickhoff, editor, *TABLEAUX*, LNCS 1847. Springer, 2000.

[Wil00] R. Wille. Boolean concept logic. In G. Mineau and B. Ganter, editors, *Int. Conf. Conceptual Structures*, LNCS 1867, pages 317–331. Springer, 2000.