An Abduction-based Method for Index Relaxation in Taxonomy-based Sources

Carlo Meghini[1], Yannis Tzitzikas[1]*, and Nicolas Spyratos[2]

[1] Consiglio Nazionale delle Ricerche
Istituto della Scienza e delle Tecnologie della Informazione, Pisa, Italy
[2] Laboratoire de Recherche en Informatique,
Universite de Paris-Sud, France

Abstract. The extraction of information from a source containing term-classified objects is plagued with uncertainty. In the present paper we deal with this uncertainty in a *qualitative* way. We view an information source as an agent, operating according to an open world philosophy. The agent knows some facts, but is aware that there could be other facts, compatible with the known ones, that might hold as well, although they are not captured for lack of knowledge. These facts are, indeed, *possibilities.* We view possibilities as explanations and resort to abduction in order to define precisely the possibilities that we want our system to be able to handle. We introduce an operation that extends a taxonomy-based source with possibilities, and then study the property of this operation from a mathematical point of view.

1 Introduction

Taxonomies are probably the oldest conceptual modeling tool. Nevertheless, they make a powerful tool still used for indexing by terms books in libraries, and very large collections of heterogeneous objects (e.g. see [8]) and the Web (e.g. Yahoo!, Open Directory). The extraction of information from an information source (hereafter, IS) containing term-classified objects is plagued with uncertainty. From the one hand, the indexing of objects, that is the assignment of a set of terms to each object, presents many difficulties, whether it is performed manually by some expert or automatically by a computer programme. In the former case, subjectivity may play a negative role (e.g. see [10]); in the latter case, automatic classification methods may at best produce approximations. On the other hand, the query formulation process, being linguistic in nature, would require perfect attuning of the system and the user language, an assumption that simply does not hold in open settings such as the Web.
A collection of textual documents accessed by users via natural language queries is clearly a kind of IS, where documents play the role of objects and words play the role of terms. In this context, the above mentioned uncertainty is typically

* This work has been carried out while Dr. Tzitzikas was a visiting researcher at CNR-ISTI as an ERCIM fellow. Our thanks to ERCIM.

dealt with in a quantitative way, *i.e.* by means of numerical methods: in a document index, each term is assigned a *weight,* expressing the extent to which the document is deemed to be about the term. The same treatment is applied to each user query, producing an index of the query which is a formal representation of the user information need of the same kind as that of each document. Document and query term indexes are then matched against each other in order to estimate the relevance of the document to a query (e.g. see [1]).

In the present study, we take a different approach, and deal with uncertainty in a *qualitative* way. We view an IS as an agent, operating according to an open world philosophy. The agent knows some facts, but it does not interpret these facts as the only ones that hold; the agent is somewhat aware that there could be other facts, compatible with the known ones, that might hold as well, although they are not captured for lack of knowledge. These facts are, indeed, *possibilities.* One way of defining precisely in logical terms the notion of possibility, is to equate it with the notion of *explanation.* That is, the set of terms associated to an object is viewed as a *manifestation* of a phenomenon, the indexing process, for which we wish to find an explanation, justifying why the index itself has come to be the way it is. In logic, the reasoning required to infer explanations from given theory and observations, is known as *abduction.* We will therefore resort to abduction in order to define precisely the possibilities that we want our system to be able to handle. In particular, we will define an operation that extends an IS by adding to it a set (term, object) pairs capturing the sought possibilities, and then study the property of this operation from a mathematical point of view. The introduced operation can be used also for ordering query answers using a *possibility*-based measure of relevance.

The paper is structured as follows. Sections 2 and 3 provide the basis of our framework, introducing ISs and querying. Section 4 introduces extended ISs and Section 5 discusses query answering in such sources. Subsequently, Section 6 generalizes extended ISs and introduces iterative extensions of ISs. Finally, Section 7 concludes the paper. For reasons of space, proofs are just sketched.

2 Information Sources

An IS consists of two elements. The first one is a taxonomy, introduced next.

Definition 1: A *taxonomy* is a pair $O = (T, K)$ where T is a finite set of symbols, called the *terms* of the taxonomy, and K is a finite set of conditionals on T, *i.e.* formulae of the form $p \rightarrow q$ where p and q are terms; K is called the *knowledge base* of the taxonomy. The *knowledge graph of* O is the directed graph $G_O = (T, L)$, such that $(t, t') \in L$ iff $t \rightarrow t'$ is in K. □

The second element of an IS is a structure, in the logical sense of the term.

Definition 2: Given a taxonomy $O = (T, K)$, a *structure on* O is a pair $U = (Obj, I)$ where: Obj is a countable set of objects, called the *domain* of the structure, and I is a finite relation from T to Obj, that is $I \subseteq T \times Obj$, called the *interpretation* of the structure. □

As customary, we will treat the relation I as a function from terms to sets of objects and, where t is a term in T, write $I(t)$ to denote the extension of t, *i.e.* $I(t) = \{o \in Obj \mid (t, o) \in I\}$.

Definition 3: An *information source* (IS) S is a pair $S = (O, U)$ where O is a taxonomy and U is a structure on O. □

It is not difficult to see the strict correspondence between the notion of IS and that of a restricted monadic predicate calculus: the taxonomy plays the role of the theory, by providing the predicate symbols (the terms) and a set of axioms (the knowledge base); the structure plays the basic semantical role, by providing a domain of interpretation and an extension for each term. These kinds of systems have also been studied in the context of description logics [3], where terms are called *concepts* and axioms are called *terminological axioms.* For the present study, we will mostly focus on the information relative to single objects, which takes the form of a propositional theory, introduced by the next Definition.

Definition 4: Given an IS S and an object $o \in Obj$, the *index of o in S, $ind_S(o)$*, is the set of terms in whose extension o belongs according to the structure S, formally: $ind_S(o) = \{t \in T \mid (t, o) \in I\}$. The *context of o in S, $C_S(o)$*, is defined as: $C_S(o) = ind_S(o) \cup K$. □

For any object o, $C_S(o)$ consists of terms and simple conditionals that collectively form all the knowledge about o that S has. Viewing the terms as propositional variables makes object contexts propositional theories. This is the view that will be adopted in this study.

Example 1: Throughout the paper, we will use as an example the IS graphically illustrated in Figure 1, given by (the abbreviations introduced in Figure 1 are used for reasons of space): $T = \{\top, \texttt{C}, \texttt{SC}, \texttt{MPC}, \texttt{UD}, \texttt{R}, \texttt{M}, \texttt{UMC}\}$, $K = \{\texttt{C} \rightarrow \top, \texttt{SC} \rightarrow \texttt{C}, \texttt{MPC} \rightarrow \texttt{C}, \texttt{UD} \rightarrow \top, \texttt{R} \rightarrow \texttt{SC}, \texttt{M} \rightarrow \texttt{SC}, \texttt{UMC} \rightarrow \texttt{MPC}, \texttt{UMC} \rightarrow \texttt{UD}\}$, and U is the structure given by: $Obj = \{1, 2\}$ and $I = \{(\texttt{SC}, 1), (\texttt{M}, 2), (\texttt{MPC}, 2)\}$. The index of object 2 in S, $ind_S(2)$ is $\{\texttt{M}, \texttt{MPC}\}$, while the context of 2 in S is $C_S(2) = ind_S(2) \cup K$. Notice that the taxonomy of the example has a maximal element, $\top$, whose existence is not required in every taxonomy. □

Given a set of propositional variables P, a *truth assignment for P* is a function mapping P to the set of standard truth values, denoted by **T** and **F**, respectively [5]. A truth assignment V *satisfies* a sentence σ, $V \models \sigma$, if σ is true in V, according to the truth valuation rules of predicate calculus (PC). A set of sentences Σ *logically implies* the sentence α, $\Sigma \models \alpha$, iff every truth assignment which satisfies every sentence in Σ also satisfies α.

In the following, we will be interested in deciding whether a certain conditional is logically implied by a knowledge base.

Proposition 1: Given a taxonomy $O = (T, K)$ and any two terms p, q in T, $K \models p \rightarrow q$ iff there is a path from p to q in G_O. □

From a complexity point of view, the last Proposition reduces logical implication of a conditional to the well-known problem on graphs REACHABILITY, which has been shown to have time complexity equal to $O(n^2)$, where n is the number of nodes of the graph [7]. Consequently, for any two terms p, q in T, $K \models p \to q$ can be decided in time $O(|T|^2)$.

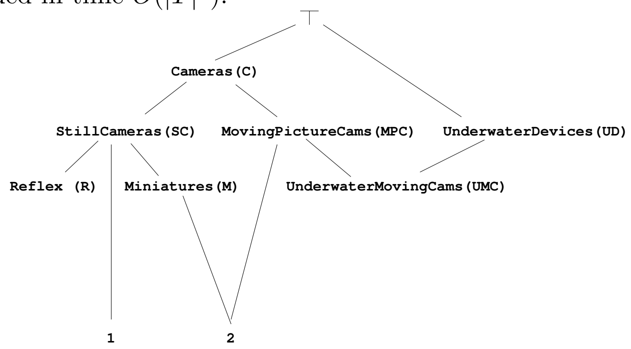

Fig. 1. A source

3 Querying Information Sources

We next introduce the query language for extracting information from an IS in the traditional question-answering way.

Definition 5: Given a taxonomy $O = (T, K)$, the *query language for* O, $\mathcal{L}_O$, is defined by the following grammar, where t is a term in T:

$$q ::= t \mid q \wedge q' \mid q \vee q' \mid \neg q \mid (q) \qquad \Box$$

The answer to queries is defined in logical terms by taking a model-theoretic approach, compliant with the fact that the semantical notion of structure is used to model the extensional data of an IS. To this end, we next select, amongst the models of object contexts, the one realizing a closed-world reading of an IS, whose existence and uniqueness trivially follow from the next Definition.

Definition 6: Given an IS S, for every object $o \in Obj$, the *truth model of* o *in* S, $V_{o,S}$, is the truth assignment for T defined as follows, for each term $t \in T$:

$$V_{o,S}(t) = \begin{cases} \mathbf{T} \text{ if } C_S(o) \models t \\ \mathbf{F} \text{ otherwise} \end{cases}$$

Given a query φ in $\mathcal{L}_O$, the *answer of* φ *in* S is the set of objects whose truth model satisfies the query: $ans(\varphi, S) = \{o \in Obj \mid V_{o,S} \models \varphi\}$. □

In the Boolean model of information retrieval, a document is returned in response to a query if the index of the document satisfies the query. Thus, the above definition extends Boolean retrieval by considering also the knowledge base in the retrieval process.

Example 2: The answer to the query $\mathtt{C}$ in the IS introduced in Example 1, $ans(\mathtt{C}, S)$, consists of both object 1 (since $\{\mathtt{SC},\ \mathtt{SC} \to \mathtt{C}\} \subseteq C_S(1)$ hence $V_{1,S}(\mathtt{C}) = \mathbf{T}$) and object 2 (since $\{\mathtt{MPC},\ \mathtt{MPC} \to \mathtt{C}\} \subseteq C_S(2)$ hence $V_{2,S}(\mathtt{C}) = \mathbf{T}$). □

The next definition introduces the function α_S, which, along with Proposition 1, provides a mechanism for the computation of answers.

Definition 7: Given an IS S, the *solver of* S, α_S, is the total function from queries to sets of objects, $\alpha_S : \mathcal{L}_O \rightarrow \mathcal{P}(Obj)$, defined as follows:

$$\alpha_S(t) = \bigcup \{I(u) \mid K \models u \rightarrow t\}$$

$\alpha_S(q \wedge q') = \alpha_S(q) \cap \alpha_S(q')$, $\alpha_S(q \vee q') = \alpha_S(q) \cup \alpha_S(q')$, and $\alpha_S(\neg q) = Obj \setminus \alpha_S(q)$. □

As intuition suggests, solvers capture sound and complete query answerers.

Proposition 2: For all ISs S and queries $\varphi \in \mathcal{L}_O$, $ans(\varphi, S) = \alpha_S(\varphi)$. □

We shall also use I^- to denote the restriction of α_S on T, i.e. $I^- = \alpha_{S|T}$.

Example 3: In the IS previously introduced, the term `C` can be reached in the knowledge graph by each of the following terms: `C`, `SC`, `MPC`, `R`, `M`, and `UMC`. Hence: $ans(\texttt{C}, S) = \alpha_S(\texttt{C}) = I(\texttt{C}) \cup I(\texttt{SC}) \cup I(\texttt{MPC}) \cup I(\texttt{R}) \cup I(\texttt{M}) \cup I(\texttt{UMC}) = \{1,\ 2\}$. Likewise, it can be verified that $ans(\texttt{M}, S) = \{2\}$ and $ans(\texttt{UMC}, S) = \emptyset$. □

In the worst case, answering a query requires (a) to visit the whole knowledge graph for each term of the query and (b) to combine the so obtained sets of objects via the union, intersection and difference set operators. Since the time complexity of each such operation is polynomial in the size of the input, the time complexity of query answering is polynomial.

4 Extended Information Sources

Let us suppose that a user has issued a query against an IS and that the answer does not contain objects that are relevant to the user information need. The user may not be willing to replace the current query with another one, for instance because of lack of knowledge on the available language or taxonomy. In this type of situation, both database and information retrieval (IR) systems offer practically no support. If the IS does indeed contain relevant objects, the reason of the user's disappointment is indexing *mismatch:* the objects have been indexed in a way that is different from the way the user would expect.

One way of handling the problem just described, would be to consider the index of an IS not as the ultimate truth about how the world is and is not, but as a flexible repository of information, which may be interpreted in a more liberal or more conservative way, depending on the context. For instance, the above examples suggest that a more liberal view of the IS, in which the camera in question is indexed under the term `M`, could help the user in getting out of the impasse. One way of defining precisely in logical terms the discussed extension, is to equate it with the notion of *explanation.* That is, we view the index of an object as a *manifestation,* or observation, of a phenomenon, the indexing process, for which we wish to find an explanation, justifying why the index itself has come to be as it is. In logic, the reasoning required to infer explanations from given theory and observations, is known as *abduction.*

The model of abduction that we adopt is the one presented in [4]. Let $\mathcal{L}_V$ be the language of propositional logic over an alphabet V of propositional variables, with syntactic operators $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\top$ (a constant for truth) and $\bot$ (falsity). A *propositional abduction problem* is a tuple $\mathcal{A} = \langle V, H, M, Th \rangle$, where V is a finite set of propositional variables, $H \subseteq V$ is the set of hypotheses, $M \subseteq V$ is the set of manifestations, and $Th \subseteq \mathcal{L}_V$ is a consistent theory. $S \subseteq H$ is a solution (or explanation) for $\mathcal{A}$ iff $Th \cup S$ is consistent and $Th \cup S \models M$. $Sol(\mathcal{A})$ denotes the set of the solutions to $\mathcal{A}$. In the context of an IS S, the terms in S taxonomy play both the role of the propositional variables V and of the hypotheses H, as there is no reason to exclude *apriori* any term from an explanation; the knowledge base in S taxonomy plays the role of the theory Th; the role of manifestation, for a fixed object, is played by the index of the object. Consequently, we have the following

Definition 8: Given an IS S and object $o \in Obj$, the *propositional abduction problem for o in S*, $\mathcal{A}_S(o)$, is the propositional abduction problem $\mathcal{A}_S(o) = \langle T, T, ind_S(o), K \rangle$. The solutions to $\mathcal{A}_S(o)$ are given by:

$$Sol(\mathcal{A}_S(o)) = \{A \subseteq T \mid K \cup A \models ind_S(o)\}$$

where the consistency requirement on $K \cup A$ has been omitted since for no knowledge base K and set of terms A, $K \cup A$ can be inconsistent. □

Usually, certain explanations are preferable to others, a fact that is formalized in [4] by defining a preference relation $\preceq$ over $Sol(\mathcal{A})$. Letting $a \prec b$ stand for $a \preceq b$ and $b \not\preceq a$, the set of preferred solutions is given by:

$$Sol_{\preceq}(\mathcal{A}) = \{S \in Sol(\mathcal{A}) \mid \nexists S' \in Sol(\mathcal{A}) : S' \prec S\}.$$

In the present context, we require the preference relation to satisfy the following criteria, reflecting the application priorities in order of decreasing priority: (1) explanations including only terms in the manifestation are less preferable than explanations including also terms not in the manifestation; (2) explanations altering the behaviour of the IS to a minimal extent, are to be preferred; (3) between two explanations that alter the behaviour of the IS equally, the simpler, that is the smaller, one is to be preferred. Without the first criterion, all minimal solutions would be found amongst the subsets of M, a clearly undesirable effect, at least as long as alternative explanations are possible. In order to formalize our intended preference relation, we start by defining perturbation.

Definition 9: Given an IS S, an object $o \in Obj$ and a set of terms $A \subseteq T$, the *perturbation of A on S with respect to o*, $p(S, o, A)$ is given by the number of additional terms in whose extension o belongs, once the index of o is extended with the terms in A. Formally:

$$p(S, o, A) = |\{t \in T \mid (C_S(o) \cup A) \models t \text{ and } C_S(o) \not\models t\}|. \qquad \square$$

As a consequence of the monotonicity of the PC, for all ISs S, objects $o \in Obj$ and sets of terms $A \subseteq T$, $p(S, o, A) \geq 0$. In particular, $p(S, o, A) = 0$ iff $A \subseteq ind_S(o)$.

We can now define the preference relation over solutions of the above stated abduction problem.

Definition 10: Given an IS S, an object $o \in Obj$ and two solutions A and A' to the problem $\mathcal{A}_S(o)$, $A \preceq A'$ if either of the following holds:

1. $p(S, o, A') = 0$
2. $0 < p(S, o, A) < p(S, o, A')$
3. $0 < p(S, o, A) = p(S, o, A')$, and $A \subseteq A'$. □

In order to derive the set $Sol_{\preceq}(\mathcal{A}_S(o))$, we introduce the following notions.

Definition 11: Given an IS S and an object $o \in Obj$, the *depth of* $Sol(\mathcal{A}_S(o))$, d_o, is the maximum perturbation of the solutions to $\mathcal{A}_S(o)$, that is:

$$d_o = max\{p(S, o, A) \mid A \in Sol(\mathcal{A}_S(o))\}$$

Moreover, two solutions A and A' are *equivalent,* $A \equiv A'$, iff they have the same perturbation, that is $p(S, o, A) = p(S, o, A')$. □

It can be readily verified that $\equiv$ is an equivalence relation over $Sol(\mathcal{A}_S(o))$, determining the partition $\pi_{\equiv}$ whose elements are the set of solutions having the same perturbation. Letting $\mathcal{P}_i$ stand for the solutions having perturbation i,

$$\mathcal{P}_i = \{A \in Sol(\mathcal{A}_S(o)) \mid p(S, o, A) = i\}$$

it turns out that $\pi_{\equiv}$ includes one element for each perburbation value in between 0 and d_o, as the following Proposition states.

Proposition 3: For all ISs IS S and objects $o \in Obj$, $\pi_{\equiv} = \{\mathcal{P}_i \mid 0 \leq i \leq d_o\}$. In order to prove the Proposition, it must be shown that $\{\mathcal{P}_i \mid 0 \leq i \leq d_o\}$ is indeed a partition, that is: (1) $\mathcal{P}_i \neq \emptyset$ for each $0 \leq i \leq d_o$; (2) $\mathcal{P}_i \cap \mathcal{P}_j = \emptyset$ for $0 \leq i, j \leq d_o$, $i \neq j$; (3) $\bigcup\{\mathcal{P}_i \mid 0 \leq i \leq d_o\} = Sol(\mathcal{A}_S(o))$. Items 2 and 3 above are easily established. Item 1 is trivial for $d_o = 0$. For $d_o > 0$, item 1 can be established by backward induction on i : the basis step, $\mathcal{P}_{d_o} \neq \emptyset$, is true by definition. The inductive step, $\mathcal{P}_k \neq \emptyset$ for $k > 0$ implies $\mathcal{P}_{k-1} \neq \emptyset$, can be proved by constructing a solution having perturbation $k-1$ from a solution with perturbation k. Finally, it trivially follows that this partition is the one induced by the $\equiv$ relation. □

We are now in the position of deriving $Sol_{\preceq}(\mathcal{A}_S(o))$.

Proposition 4: For all ISs S and objects $o \in Obj$,

$$Sol_{\preceq}(\mathcal{A}_S(o)) = \begin{cases} \mathcal{P}_0 & \text{if } d_o = 0 \\ \{A \in \mathcal{P}_1 \mid \text{for no } A' \in \mathcal{P}_1,\ A' \subset A\} & \text{if } d_o > 0 \end{cases}$$

This proposition is just a corollary of the previous one. Indeed, if d_o is 0, by Proposition 3, $Sol(\mathcal{A}_S(o)) = \mathcal{P}_0$ and by Definition 10, all elements in $Sol(\mathcal{A}_S(o))$ are minimal. If, on the other hand, d_o is positive, then by criterion (1) of Definition 10, all solutions with non-zero perturbation are preferable to those in $\mathcal{P}_0$, and not viceversa; and by criterion (2) of Definition 10, all solutions with perturbation equal to 1 are preferable to the remaining, and not viceversa. Hence, for a positive d_o, minimal solutions are to be found in $\mathcal{P}_1$. Finally, by considering the containment criterion set by item (3) of Definition 10, the Proposition results.

Example 4: Let us consider again the IS S introduced in Example 1, and the problem $\mathcal{A}_S(1)$. The manifestation is given by $\{\texttt{SC}\}$. Letting B stand for the set $\{\texttt{UMC}, \texttt{MPC}, \texttt{UD}, \top, \texttt{C}\}$, it can be verified that: $Sol(\mathcal{A}_S(1)) = \mathcal{P}(T) \setminus \mathcal{P}(B)$ as B includes all the terms in T not implying $\texttt{SC}$. Since $d_o = 5$, minimal solutions are to be found in the set $\mathcal{P}_1$. By considering all sets of terms in $Sol(\mathcal{A}_S(1))$, it

can be verified that: $\mathcal{P}_1 = \{\{\mathtt{M}\} \cup A \mid A \in \mathcal{P}(\{\mathtt{SC}, \mathtt{C}, \top\})\} \cup \{\{\mathtt{R}\} \cup A \mid A \in \mathcal{P}(\{\mathtt{SC}, \mathtt{C}, \top\})\} \cup \{\{\mathtt{SC}, \mathtt{UD}\} \cup A \mid A \in \mathcal{P}(\{\top, \mathtt{C}\})\} \cup \{\{\mathtt{SC}, \mathtt{MPC}\} \cup A \mid A \in \mathcal{P}(\{\top, \mathtt{C}\})\}$. By applying the set containment criterion, we have: $Sol_{\preceq}(\mathcal{A}_S(1)) = \{\{\mathtt{M}\}, \{\mathtt{R}\}, \{\mathtt{SC}, \mathtt{UD}\}, \{\mathtt{SC}, \mathtt{MPC}\}\}$. Analogously, it can be verified that:
$Sol_{\preceq}(\mathcal{A}_S(2)) = \{\{\mathtt{M}, \mathtt{MPC}, \mathtt{UD}\}, \{\mathtt{R}, \mathtt{M}, \mathtt{MPC}\}\}$. □

We now introduce the notion of *extension* of an IS. The idea is that an extended IS (EIS for short) adds to the original IS all and only the indexing information captured by the abduction process illustrated in the previous Section. In order to maxime the extension, all the minimal solutions are included in the EIS.

Definition 12: Given an IS S and an object $o \in Obj$, the *abduced index* of o, $abind_S(o)$, is given by:

$$abind_S(o) = \bigcup Sol_{\preceq}(\mathcal{A}_S(o)).$$

The *abduced interpretation of* S, I^+, is given by

$$I^+ = I \cup \{\langle t, o\rangle \in (T \times Obj) \mid t \in abind_S(o)\}.$$

Finally, the *extended* IS, S^e, is given by $S^e = (O, U^e)$ where $U^e = (Obj, I^+)$. □

Example 5: From the last Example, it follows that the extended S is given by $S^e = (O, U^e), U^e = (Obj, I^+)$ where: $abind_S(1) = \{\mathtt{SC}, \mathtt{M}, \mathtt{R}, \mathtt{UD}, \mathtt{MPC}\}$, $abind_S(2) = \{\mathtt{M}, \mathtt{MPC}, \mathtt{UD}, \mathtt{R}\}$ and
$I^+ = \{(\mathtt{SC}, 1),\ (\mathtt{M}, 1),\ (\mathtt{R}, 1),\ (\mathtt{UD}, 1),\ (\mathtt{MPC}, 1),\ (\mathtt{M}, 2),\ (\mathtt{MPC}, 2),\ (\mathtt{UD}, 2),\ (\mathtt{R}, 2)\}$ □

5 Querying Extended Information Sources

As anticipated in Section 4, EISs are meant to be used in order to obtain more results about an already stated query, without posing a new query to the underlying information system. The following Example illustrates the case in point.

Example 6: The answer to the query $\mathtt{M}$ in the extended IS derived in the last Example, $ans(\mathtt{M}, S^e)$, consists of both object 1 (since $\mathtt{M} \in abind_S(1)$ hence $\mathtt{M} \in C_{S^e}(1)$) and object 2 (since $(\mathtt{M}, 2) \in I$ hence $(\mathtt{M}, 2) \in I^+$). Notice that 1 is not returned when $\mathtt{M}$ is stated against S, *i.e.* $ans(\mathtt{M}, S) \subset ans(\mathtt{M}, S^e)$. Instead, $ans(\mathtt{UMC}, S) = ans(\mathtt{UMC}, S^e) = \emptyset$. □

It turns that queries stated against an EIS can be answered without actually computing the whole EIS. In order to derive an answering procedure for queries posed against an EIS, we introduce a recursive function on the IS query language $\mathcal{L}_O$, in the same style as the algorithm for querying IS presented in Section 3.

Definition 13: Given an IS S, the *extended solver of* S, α^e_S, is the total function from queries to sets of objects, $\alpha^e_S : \mathcal{L}_O \to \mathcal{P}(Obj)$, defined as follows:

$$\begin{aligned}
\alpha^e_S(t) &= \bigcap\{\alpha_S(u) \mid t \to u \in K \text{ and } K \not\models u \to t\} \\
\alpha^e_S(q \wedge q') &= \alpha^e_S(q) \cap \alpha^e_S(q') \\
\alpha^e_S(q \vee q') &= \alpha^e_S(q) \cup \alpha^e_S(q') \\
\alpha^e_S(\neg q) &= Obj \setminus \alpha^e_S(q)
\end{aligned}$$

where α_S is the solver of S. □

Note that since $\top$ is the maximal element the set $\{\alpha_S(u) \mid \top \rightarrow u \in K$ and $K \not\models u \rightarrow \top\}$ is empty. This means that $\alpha^e_S(\top)$, i.e. $\bigcap\{\alpha_S(u) \mid \top \rightarrow u \in K$ and $K \not\models u \rightarrow \top\}$ is actually the intersection of an empty family of subsets of Obj. However, according to the Zermelo axioms of set theory (see [2] for an overview), the intersection of an empty family of subsets of a universe equals to the universe. In our case, the universe is the set of all objects known to the source, i.e. the set Obj, thus we conclude that $\alpha^e_S(\top) = Obj$. The same holds for each maximal element (if the taxonomy has more than one maximal elements).

Proposition 5: For all ISs S and queries $\varphi \in \mathcal{L}_O$, $ans(\varphi, S^e) = \alpha^e_S(\varphi)$. □

Example 7: By applying the last Proposition, we have:
$ans(\mathtt{M}, S^e) = \alpha^e_S(\mathtt{M}) = \alpha_S(\mathtt{SC}) = I(\mathtt{SC}) \cup I(\mathtt{R}) \cup I(\mathtt{M}) = \{1,\ 2\}$. □

6 Iterative Extension of Information Sources

Intuitively, we would expect that $\cdot^+$ be a function which, applied to an IS interpretation, produces a new interpretation that is equal to or larger than the original extension, the former case corresponding to the situation in which the knowledge base of the IS does not enable to find any explanations for each object index. Technically, this amounts to say that $\cdot^+$ is a monotonic function, which is in fact the case. Then, by iterating the $\cdot^+$ operator, we expect to move from an interpretation to a larger one, until an interpretation is reached which cannot be extended any more. Also this turns out to be true, and in order to show it, we will model the domain of the $\cdot^+$ operator as a complete partial order, and use the notion of fixed point in order to capture interpretations that are no longer extensible.

Proposition 6: Given an IS S, the *domain* of S is the set $\mathcal{D}$ given by $\mathcal{D} = \{I \cup A \mid A \in \mathcal{P}(T \times Obj)\}$. Then, $\cdot^+$ is a continuous function on the complete partial order $(\mathcal{D}, \subseteq)$.

The proof that $(\mathcal{D}, \subseteq)$ is a complete partial order is trivial. The continuity of $\cdot^+$ follows from its monotonicity (also, a simple fact to show) and the fact that in the considered complete partial order all chains are finite, hence the class of monotonic functions coincides with the class of continuous functions [6]. □

As a corollary of the previous Proposition and of the Knaster-Tarski fixed point theorem, we have:

Proposition 7: The function $\cdot^+$ has a least fixed point that is the least upper bound of the chain $\{I, I^+, (I^+)^+, \ldots\}$. □

Example 8: Let R be the EIS derived in the last Example, *i.e.* $R = S^e$, and let us consider the problem $\mathcal{A}_R(1)$, for which the manifestation is given by the set $abind_S(1)$ above. It can be verified that $Sol(\mathcal{A}_R(1)) = \mathcal{P}_0 \cup \mathcal{P}_1$, where:

$$\mathcal{P}_0 = \{\{\mathtt{R}, \mathtt{M}, \mathtt{MPC}, \mathtt{UD}\} \cup A \mid A \in \mathcal{P}(\{\mathtt{SC}, \mathtt{C}, \top\})\}$$
$$\mathcal{P}_1 = \{\{\mathtt{R}, \mathtt{M}, \mathtt{UMC}\} \cup A \mid A \in \mathcal{P}(\{\mathtt{SC}, \mathtt{C}, \top, \mathtt{MPC}, \mathtt{UD}\})\}$$

Therefore: $Sol_{\preceq}(\mathcal{A}_R(1)) = \{\{\mathtt{R}, \mathtt{M}, \mathtt{UMC}\}\}$ from which we obtain: $abind_R(1) = \{\mathtt{R}, \mathtt{M}, \mathtt{UMC}\}$ which means that the index of object 1 in R has been extended with the term UMC. If we now set $P = R^e$, and consider the problem $\mathcal{A}_P(1)$, we find

$$Sol(\mathcal{A}_P(1)) = \mathcal{P}_0 = \{\{\mathtt{R}, \mathtt{M}, \mathtt{UMC}\} \cup A \mid A \in \mathcal{P}(\{\mathtt{SC}, \mathtt{MPC}, \mathtt{UD}, \mathtt{C}, \top\})\}$$

Consequently, $Sol_{\preceq}(\mathcal{A}_P(1)) = \{\{\mathtt{R}, \mathtt{M}, \mathtt{UMC}\}\}$ and $abind_P(1) \subseteq ind_P(1)$. Analogously, we have $abind_R(2) = ind_R(2) \cup \{\mathtt{UMC}\}$ and $abind_P(2) \subseteq ind_P(2)$. Thus, since $((I^+)^+)^+ = (I^+)^+$, $(I^+)^+$ is a fixed point, which means that P is no longer extensible. Notice that $\emptyset = ans(\mathtt{UMC}, S) = ans(\mathtt{UMC}, R) \subset ans(\mathtt{UMC}, P) = \{1,\ 2\}$. □

7 Conclusion and Future Work

To alleviate the problem of indexing uncertainty we have proposed a mechanism which allows liberating the index of a source in a gradual manner. This mechanism is governed by the notion of explanation, logically captured by abduction. The proposed method can be implemented as an answer enlargement [3] process where the user is not required to give additional input, but from expressing his/her desire for more objects. Another interesting remark is that the abduced extension operation can be applied not only to manually constructed taxonomies but also to taxonomies derived automatically on the basis of an inference service. For instance, it can be applied on sources indexed using taxonomies of *compound terms* which are defined algebraically [9]. The introduced framework can be also applied for ranking the objects of an answer according to an explanation-based measure of relevance. In particular, we can define the *rank* of an object o as follows: $rank(o) = min\{\ k \mid o \in \alpha_S^{(k)e}(\varphi)\}$.

References

1. R. Baeza-Yates and B. Ribeiro-Neto. *"Modern Information Retrieval"*. ACM Press, Addison-Wesley, 1999.
2. George Boolos. *"Logic, Logic and Logic"*. Harvard University Press, 1998.
3. F.M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in description logics. In G. Brewka, editor, *Principles of Knowledge Representation*, Studies in Logic, Language and Information, pages 193–238. CSLI Publications, 1996.
4. T. Eiter and G. Gottlob. The complexity of logic-based abduction. *Journal of the ACM*, 42(1):3–42, January 1995.
5. H.B. Enderton. *A mathematical introduction to logic.* Academic Press, N. Y., 1972.
6. P.A. Fejer and D.A. Simovici. *Mathematical Foundations of Computer Science. Volume 1: Sets, Relations, and Induction.* Springer-Verlag, 1991.
7. C.H. Papadimitriou. *Computational complexity.* Addison-Wesley, 1994.
8. Giovanni M. Sacco. "Dynamic Taxonomies: A Model for Large Information Bases". *IEEE Transactions on Knowledge and Data Engineering*, 12(3), May 2000.
9. Y. Tzitzikas, A. Analyti, N. Spyratos, and P. Constantopoulos. "An Algebra for Specifying Compound Terms for Faceted Taxonomies". In *13th European-Japanese Conf. on Information Modelling and Knowledge Bases*, Kitakyushu, J, June 2003.
10. P. Zunde and M.E. Dexter. "Indexing Consistency and Quality". *American Documentation*, 20(3):259–267, July 1969.

[3] If the query contains negation then the answer can be reduced.