Abduction and Learning

Yannis Dimopoulos and Antonis Kakas
Department of Computer Science, University of Cyprus
75 Kallipoleos Str., P.O. Box 537, CY-1678, Nicosia, Cyprus
{yannis, antonis}@turing.cs.ucy.ac.cy

Abstract. In this paper we study the problem of of integrating abduction and learning as they appear in Artificial Intelligence. A general comparison of abduction and induction as separate inferences and fields in AI is given. Based on this analysis we study their possible interaction and integration. We introduce the notions of of abductive concept learning as a framework for learning with incomplete background theories. Together with this we propose a general methodology for incorporating abduction in inductive concept learning that allows us to exploit more fully in the learning process the knowledge contained in the background theory. This basic methodology extends in a natural way to the more general context of theory revision.

1 Introduction

Over the last few years many authors have studied, within the wider area of Artificial Intelligence, the two fields of abduction and learning separately and have shown their importance in providing systems with various advanced capabilities such as the ability to acquire and assimilate new information intelligently, explain and diagnose situations, plan actions and communicate and cooperate with other systems. More recently, there has been some work ([1], [2], [4], [5], [6], [10], [14], [16], [19], [39], [47], [50]) that incorporates abduction and learning mainly in the context of particular learning systems in order to enchance the system's competence. It is therefore important to understand more generally the relation of the two fields and study their effective incorporation and integration.

This paper is a study of the relationship between abduction and learning as they appear in Artificial Intelligence and particular within its subarea of Logic Programming. As such, it is an attempt to bring the two fields of Abductive Logic Programming and Inductive Logic Programming closer together. The integration of these two fields has recently been studied in [1]. We will be interested in examining, at different levels, the two way interaction between abductive reasoning and various forms of inductive generalization and learning. Our work aims to provide a simple understanding of this interaction that can unify the different existing works on this subject. It also aims to suggest new possible areas of cross-fertilization between abduction and Machine Learning, identify specific problems of interest and point out possible directions for their solution thus strengthening the links between the two fields. The paper does not contain an exhaustive survey of all existing works on the subject. It does though try to cover the main approaches identifying their common elements and their possible conceptual unification.

A brief review of abduction is given followed by a general comparison of abduction and induction as separate inferences in AI. Based on this analysis we then study their possible integration and interaction. We will examine how abduction can affect learning suggesting a general semantic framework and methodology for their integration. In doing this, we will investigate the relationship of abduction with the basic problem of *inductive concept learning* in Machine Learning and more particularly as is addressed in the subfield of Inductive Logic Programming (ILP) ([29], [37]). Within this study we also examine the relevance and use of abduction in Explanation-based Learning ([7], [33]).

Purely from the point of view of Machine Learning, our work can be seen, just like ILP, as part of the general study to integrate empirical and analytical based learning and to make learning more knowledge intensive. This is done by using abduction as a method to either provide suitable knowledge to the learner or to help in learning with incomplete information. In fact, some of the existing learning systems that combine empirical and analytical methods use abduction in their learning process (e.g. [4], [5], [6], [16], [19], [39], [47], [50]).

Similarly, our work can be seen within the area of multistrategy learning ([32]) where we are concerned with the problem of how abduction can be one of the different inferences and computational mechanisms in a learning system. In particular, we will examine the role of abduction in the area of *theory revision* (for a recent review see [53]) as one of the revision operators ([2], [10], [14]) and as a useful mechanism for identifying places where the theory can be revised ([39], [52]).

In an extended version of this paper we include a proposal of a new framework for learning where the notion of coverage of the training examples is given by abduction, with the abductive hypotheses in terms of the concept(s) to be learned.

2 A Preliminary Review of Abduction

In this paper we assume that the reader is familiar with the basic techniques and ideas of inductive learning and theory revision in Machine Learning, as they are reviewed for example in [29], [37], [53]. In this section we review briefly the field of abduction as this is studied in the area of Artificial Intelligence. The aim of this review is to present the main ideas and problems from this field that will be needed in the rest of the paper. As such, this review is presented from the perspective of the problem of relating abduction to learning. In many cases the review is (deliberately) simplified to a level sufficient for the purposes of this paper. For more extensive reviews the reader can refer to various existing surveys, e.g. [25], [40]. As in the rest of this paper we will confine ourselves within the logic-based approach to abduction and the role of abduction in logic programming[1].

Abduction is reasoning to an explanation according to a given known theory. We explain the observation that the "grass is wet" by "rain" according to the known theory that "rain results in wet grass". In general, the basic notion of abduction can be formalized as follows. Given a set of sentences T (a theory presentation), and a sentence O (observation), the abductive task is the problem of finding a set of sentences H (abductive explanation for O) such that:

(1) $T \cup H \models O$, and

[1]Other approaches to abduction include the set covering approach see e.g. [44] or case-based explanation see e.g. [30].

(2) $T \cup H$ is consistent.

This basic formalization though, does not capture fully the explanatory nature of the abductive explanation H in the sense that it necessarily conveys some reason why the observations hold. It would for example allow an observation O to be explained by itself or in terms of some other observations rather than in terms of some "deeper" reason for which the observation must hold according to the theory T. Also as the above specification stands the observation can be abductively explained by generating in H some new (general) theory completely unrelated to the given theory T. In this case H does not account for the observations O according to the given theory T and in this sense may not be considered as an explanation for O relative to T. For these reasons, in order to specify a "level" at which the explanations are required and to understand these relative to the given general theory about the domain of interest, the members of an explanation are restricted to belong to a special pre-assigned, domain-specific class of sentences called *abducible*. An abductive explanation of O is then seen as a reason for O, extracted out of the given general theory T by examining an appropriate part (or instance) of this theory.

The abducible sentences, also called assumptions or hypotheses, are usually build from a set of designated abducible predicates, A, which are not defined in the theory T. Usually, when representing our knowledge for a problem in the theory T, the following two implicit assumptions on T and its abducible predicates are made. First, the abducible predicates carry all the incompleteness of the theory T (in modelling the external problem domain) in the sense that if we (could) complete the abducible predicates then the theory would describe completely the problem. Second, the abducible predicates are "basic" relations for the particular way in which the theory T models the problem, and so they are not to be defined by other relations. These assumptions though can be relaxed especially when abduction is used more as a technique within some other framework[2] rather than as a form of reasoning for knowledge representation. Abduction thus allows us to represent and reason with problems where we have incomplete information about the related domain when this incompleteness can be captured (or isolated) in some of the relations of the domain.

The assumptions on the abducible predicates used for building up the explanations may be subject to restrictions that are expressed through *integrity constraints*. Hence, although in general the abducible predicates have no definition in the theory T, some information about them may be known implicitly through integrity constraints, IC, that must always be satisfied by the theory together with any explanation that we want to adopt. Therefore, in a knowledge representation framework where abduction is one of the forms of reasoning, a theory or an abductive theory is a triple, denoted by $< T, A, IC >$, where T is the background theory, A is a set of abducible predicates and IC is a set of integrity constraints. Then, one more requirement is added to the definition of the abductive problem given above, namely that

(3) $T \cup H$ satisfies IC.

There are several different notions of satisfaction of integrity constraints (see [25] and references therein). In this paper, we will mainly use integrity constraints in the form of a denial, i.e. $\forall x \neg(A_1(X), \ldots, A_n(X))$, where A_i is an atom (also written in logic programming style as, $\leftarrow A_1(X), \ldots, A_n(X)$. Such an integrity constraint is satisfied

[2]E.g. as a technique within learning as we will see in section 4.3 of this paper.

by a theory T if there does not exist an instance "d" of "X" such that $T \models A_i(d)$, for each $i = 1, 2, \ldots, n$.

Note that the integrity constraints reduce the number of explanations for a set of observations. However, there are cases where there may be several explanations that satisfy (do not violate) the integrity constraints. This problem is known as the multiple explanations problem. In fact, abduction is often described as *inference to the best explanation*. To this end, additional criteria can help to discriminate between different explanations, by rendering some of them more appropriate or plausible or preferred over others. A typical criterion is the requirement of minimality (wrt set inclusion) of the explanations. Also various (domain specific) criteria of preference can be specified. Moreover, we may be able to acquire new additional information that refutes or corroborates an explanation (see e.g. [18], [49]).

An abductive theory $< T, A, IC >$ can be viewed as representing a collection of different possible states of affairs. The possible states are characterized by the subsets of abducible instances, but only those that satisfy the integrity constraints IC are allowed. With this view an answer "yes" to a query possed to the theory, is correct if there exists an allowed state where the query is true, but such an answer will be conditional on the particular subset of abducibles that define the state. Moreover, this view allows us to consider an abductive theory as a dynamic object that evolves as new information is provided in the form of observations O. This new knowledge can be assimilated by changing the collection of allowed abducible assumptions and corresponding states that the abductive theory represents allowing only those in which O is true.

Formally, the semantics of an abductive theory can be defined as follows.

Definition 2.1 *Let $AT =< T, A, IC >$ be an abductive theory, O a set of observations, and H a set of (ground) atomic abducible formulae. Then, $T \cup H$ is an* abductive extension *of (AT, O) iff $T \cup H$ satisfies the integrity constraints IC and the observations O are entailed by $T \cup H$.*
A formula ϕ is a credulous *abductive consequence of AT and O if ϕ is true in at least one abductive extension of (AT, O). Moreover, a formula ϕ is a* skeptical *consequence of AT and O iff ϕ is true in all abductive extensions of (AT, O).*

Example 2.2 *Consider the following abductive framework $< T, A, IC >$ with T a theory on family relations:*
$father(X, Y) \leftarrow parent(X, Y), male(X)$
$mother(X, Y) \leftarrow parent(X, Y), female(X)$
$son(X, Y) \leftarrow parent(Y, X), male(X)$
$daughter(X, Y) \leftarrow parent(Y, X), female(X)$
$child(X, Y) \leftarrow son(X, Y)$
$child(X, Y) \leftarrow daughter(X, Y)$
$loves(X, Y) \leftarrow parent(X, Y)$
the integrity constraints $IC = \{\leftarrow male(X), female(X)\}$, and abducible predicates $A = \{parent, male, female\}$.
Consider now the observation $O_1 = father(Bob, Jane)$. An abductive explanation for O_1 is the set $H_1 = \{parent(Bob, Jane), male(Bob)\}$. This is the unique minimal explanation. Let now $O_2 = child(John, Mary)$ be another observation. This has two possible explanations $H_2 = \{parent(John, Mary), male(John)\}$ and $H_2' = \{parent(John, Mary), female(John)\}$. If we also knew that $male(John)$ holds then

H_2' would be rejected due to the violation of the integrity constraint. In fact, these two explanations are incompatible with each other. But, note that since $parent(John, Mary)$ belongs to both explanations, then $parent(John, Mary)$ and $loves(John, Mary)$ are skeptical consequences of this abductive framework and the observation O_2.

Abduction has many applications in Artificial Intelligence and other fields. These include diagnosis, high-level vision, natural language understanding, database view updates and others. In general, abduction can be used to assimilate new information or find ways in which a new goal can be satisfied. The general way in which abduction is exploited in these applications is that it provides an automatic translation from high level specification goals to low level computational goals. The abducible assumptions take the role of an intermediate high-level answer that can be further analyzed according to the particular task. For example, in diagnosis abduction translates the observed symptoms to causes (e.g. diseases) that would account for these. In database updates abduction can be used to translate an update request on an intensionally defined relation (or a view) to a set of updates or transactions on the extensional relations of the database. As a result of this the use of abduction allows high-level representation of problems close to their natural logical specification. We will see that this translation nature of abduction will also be relevant in the context of learning.

In order to illustrate the above ideas let us describe here briefly how we can use abduction in temporal reasoning to assimilate new knowledge or generate a plan to reach a desired state. Using the event calculus [28], [27] as a basis for a theory of actions we can write the following simplified abductive theory $< T, A, IC >$.

Example 2.3 *The general theory T contains the axiom*

$$\begin{aligned} holds(P, T_2) \quad \leftarrow \quad & happens(E, T_1), \\ & T_1 < T_2, \\ & initiates(E, P) \end{aligned}$$

together with other (domain specific) rules that define the "initiates" predicate. This general axiom expresses the persistence of a property P from the time T_1 that it is initiated by an event E to a later time T_2.
The predicate "happens" is not defined (fully) by the theory and is designated as an abducible predicate. Hence $A = \{happens\}$. The incompleteness of the theory can be seen to reside in this predicate in the sense that if we know everything that has happened then the theory has complete information about its problem domain. Assumptions or hypotheses on the abducible predicate happens must satisfy the following general integrity constraint in IC:

$$\leftarrow (happens(E, T), precondition(E, P), \neg holds(P, T))$$

that expresses the fact that we can only assume that an event E has happened at time T if all its preconditions P hold true at this time. The predicate precondition is defined within T.

This theory is not intended to be used by first completing fully or learning the definition of the incomplete predicate "happens", a task which is general impossible or inappropriate, and then use deduction to infer what properties hold at different times. On the contrary the natural way to use this theory is to first assimilate any new information (observation) that a property holds at a particular time point through the

explanation of this in terms of the happening of some event that initiates this property at an earlier point of time and use the abductive entailment notions defined above to infer what other properties must hold. Similarly, a plan of events (or actions) that can bring about a desired property (or state) can be understood as an explanation, in terms of assumptions on happens, of the goal that this property holds at some future time.

We have seen above how abduction can be used to reason with incomplete information in the representation of a problem domain. Another important form of reasoning with incomplete information is that of default reasoning e.g. [45]. In fact, abduction can be used to capture default reasoning [17], [25], [42]. To illustrate this consider again the temporal reasoning example given above. This was oversimplified since its axiom allows that once a property is initiated it continues to hold for all later times. This axiom in T should not be a strict rule but rather it should be a default rule of persistence that allows a property to persist until some contradictory information holds. We can effect such a default nature of this axiom by writing it as

$$\begin{aligned} holds(P,\, T_2) \quad \leftarrow \quad & happens(E, T_1), \\ & T_1 < T_2, \\ & initiates(E,\, P), \\ & persists(T_1,\, P,\, T_2) \end{aligned}$$

where the predicate *persists* is a new (default) abducible to be added in A which, like the abducible *happens*, is not defined in the theory T but it is constrained by the following new integrity constraint in IC:

$$\leftarrow (persists(T_1,\, P,\, T_2), happens(E,\, T), terminates(E,\, P), T_1 < T < T_2).$$

Here the predicate *terminates* is defined like the predicate *initiates* in the theory T. Thus as long as we do not know or do not assume any terminating events for some property P the assumption $persists(T_1, P, T_2)$ will be possible and so the property P will continue to hold by default (via the axiom) to later times T_2.

In a similar way, it is possible to understand *negation as failure* (NAF) through abduction [17], [25]. This is achieved by interperting negative literals as abductive hypotheses that can be assumed to hold provided that, together with the program, they satisfy a canonical set of integrity constraints. A general logic program P is thereby transformed into an abductive theory $< P^*, A^*, I^* >$ in the following way.

- A new predicate symbol p^* (the opposite of p) is introduced for each p in P, and A^* is the set of all these predicates.
- P^* is P where each negative literal *not* $p(t)$ has been replaced by $p^*(t)$.
- I^* is a set of all integrity constraints of the form

$$\begin{aligned} & \leftarrow \; p(X) \wedge p^*(X) \text{ and} \\ & \forall X \, [p(X) \vee p^*(X)]. \end{aligned}$$

The integrity constraints in I^* play a crucial role in capturing the meaning of NAF. The denials express that the newly introduced symbols p^* are the negations of the corresponding p. They prevent an assumption $p^*(t)$ if $p(t)$ holds. On the other hand the disjunctive integrity constraint force a hypothesis $p^*(t)$ whenever $p(t)$ does not hold. Within this abductive view of NAF a goal G succeeds iff there is a set of NAF

hypotheses that explain G, i.e. whose addition to the definite Horn program P^* derives G, provided also that this does not violate the integrity constraints in I^*. Note that this abductive view introduces a symmetry over the positive and negative goals.

This understanding of NAF and other forms of default reasoning through abduction provides a framework where both abduction and default reasoning are unified thus simplifying their treatment. Such an integration of these forms of reasoning is needed in many cases e.g. in planning as we have seen above. We thus get a powerful and general non- monotonic reasoning framework with a simple computational model based on abduction.

In general, the computation of abduction is goal-driven by the observation that it is trying to explain. The top-level goal of the given observation is transformed to more basic goals that can be satisfied directly by including them as hypotheses in the explanation set. Often, when the abductive theory T is a logic program, this transformation is done via some form of resolution. Together with this phase of reduction or translation of goals to abducible assumptions the computational model of abduction interleaves another phase for checking that these assumptions are consistent and that they satisfy the integrity constraints of the abductive theory. The abductive assumptions generated in existing systems of abduction usually consist of ground abducible facts [3] or existentially quantified abducible sentences together with a set of domain-constraints on the possible values of these existential variables. In this last case this is called *constructive abduction* [8], [23]. For example, in planning assumptions about events that happen must also include assumptions on the relative time order in which these events must happen for the goal to be satisfied.

An important feature of the integrity checking phase is that it may generate new goals to be satisfied which can result in the generation of further abductive hypotheses in the explanation. We thus have a full interleaving between the two types of computation with the integrity constraints playing an active role in the construction of the explanation. For example, in the planning example if the top-level goal requires the assumption of some event whose precondition is not true in the theory T then the integrity checking phase of this assumption will, through the integrity constraint above, generate a new subsidiary abductive goal to render this precondition true which in turn will produce new abductive assumptions in the explanation. This basic interleaving of the two phases of computation is also seen in the abductive treatment of NAF where every time in the computation we meet a new NAF hypothesis the state of the computation changes into the other phase (for more details see [17], [25]).

During the integrity checking phase it is also possible (see [22], [24]) to generate in the explanation "negative" assumptions namely, statements on abducibles indicating the needed absence or falsity of abducibles. In the family example the assumption $male(John)$ would also generate the assumption $not\ female(John)$ in the explanation. This extra explicit addition of the needed absence of $female(John)$ can be used to signal directly the violation of the integrity constraint should the further assumption $female(John)$ is generated by some other goal. Another case in which such "negative" assumptions can arise is when we want to explain negative observations. We will call such explanations *naf-explanations* and we will see that these can be useful in learning in several ways e.g. in relation to the consistency requirement that no negative example of the concept must be covered.

[3] Abducible predicates may be used as names of rules e.g. default rules [42]. In this case a ground abducible fact corresponds to the assumption of the corresponding instance of the rule named by this abducible predicate.

We conclude this section with the observation that abduction as it has already been developed in the wider A.I. community can provide a general semantical framework for *non-monotonic learning* with the capabilities of (i) learning rules and exceptions ([3], [15]), (ii) three-valuedness in the learning framework ([9]) and (iii) learning in the face of incomplete background predicates. Furthermore, existing abductive techniques can be directly transferred and exploited in machine learning systems to enhance their capabilities of handling NAF and other forms of negative information, integrity constraints and incomplete predicates.

3 Abduction and Induction

In this section, we discuss generally the two separate forms of reasoning, induction and abduction, pointing out their similarities and differences. This comparison will help us later in the study of the integration of the two mechanisms.

The two inferences, abduction and induction, as they are commonly used in Artificial Intelligence, share the same basic formal specification: In a logic-based approach, given a background knowledge T (often called domain theory) that describes our problem domain, and an observation (or a set of observations) O, find a hypothesis H such that

$$T \cup H \models O \qquad (1)$$

However the two inferences differ significantly in the way they satisfy this specification as they typically use and extend this in different ways. Peirce [43] distinguishes the two forms of reasoning as follows:

(a) Abduction derives the case from the rule and the result.
(b) Induction derives the rule from the case and the result.

According to this view of Peirce, "result" denotes the observation O as new information, and in the case of abduction the given knowledge, namely the theory T, is a general rule whereas in the case of induction the given knowledge consists of specific "case" information.

This simple characterization suggests in fact that it may be possible to see abduction and induction as *dual* to each other. Consider the common specification (1) that these are required to satisfy and suppose first that we are given an empirical inductive learning framework. The background theory T_L, contains facts for various objects, and the set O of observations consists of training examples on a concept, c, that we want to learn. The hypothesis H_L constructed by our inductive learning is a set of rules for c whose conditions refer to the predicates (relations) in T_L. We have the learning equation

$$T_L \cup H_L \models O.$$

Consider now exchanging the role of T_L and H_L and regard H_L as a given background theory in an abductive framework and the facts in T_L as the possible abductive hypothesis. So we let $T_A = H_L$. Then we can construct abductive explanations, H_A, of O where H_A is drawn from T_L. Hence the abductive equation $T_A \cup H_A \models O$ can be written as

$$H_L \cup T'_L \models O$$

where T'_L is a subset of T_L. Comparing this equation with the original equation of learning above we see that one is dual to the other.

Conversely, consider now that we start from an abductive framework with background theory T_A, observations O on a single concept c defined in T_A and an explanation H_A of O. Can we use H_A to learn back the definition of c in T_A ? Let us illustrate the problem with an example.

Example 3.1 *Let T_A be the theory,*
$tall(X) \leftarrow adult(X), European(X)$
$European(X) \leftarrow French(X) \qquad European(X) \leftarrow Italian(X)$
$adult(Allan) \qquad adult(Paolo)$
where the predicates "French" and "Italian" are abducible predicates and the other predicates are assumed to be completely defined by the theory.

Let also the set O of observations be $\{tall(Allan), tall(Paolo)\}$ and consider an abductive explanation H_A of O, say $H_A = \{French(Allan), Italian(Paolo)\}$. If we then take as background theory T_L the theory $T'_A \cup \{French(Allan), Italian(Paolo)\}$ where T'_A is the theory T_A without the rule for the predicate "tall" we can learn back this rule for "tall" from the same observations O as positive examples for this concept. Hence again we see we can effectively exchange the abductive hypothesis to become the background theory for learning and the abductive background theory to play the role of a hypothesis generated from learning. Starting from the abductive equation

$$T_A \cup H_A \models O$$

we have the learning equation

$$H'_A \cup T^c_A \models O$$

where the background theory H'_A is the closure of $H_A \cup (T_A - T^c_A)$ and T^c_A is the subset of T_A that defines the predicate "c" to be learned back.

This analysis suggests a methodology of how to integrate abduction in learning. Abduction can be used first to explain the training data of our learning problem to generate suitable or relevant background data on which to base the inductive generalization. In general, we may have many different abductive explanations for the same set of observations O. In the example above $H_A = \{French(Allan), French(Paolo)\}$ also forms an explanation. In such a case an appropriate background theory can again be obtained from all these possible explanations using the skeptical notion of abductive entailment (i.e. truth under all abductive explanations) with respect to the theory $T_A - T^c_A$. In the next section we will use this dual correspondence from abductive problems to learning problems to build a possible framework of integration of abduction and learning.

Abduction and induction both provide a method for handling incomplete information in the description T of a problem. The addition of the generated hypothesis H to T completes (partially) this description. The two forms of reasoning though are concerned with different types of incompleteness in the background theory T and have different aims in dealing with this. Informally, we can say that abduction **extracts** the hypothesis H from T, while induction **constructs** its hypothesis using information from T. In the case of abduction, while the theory T is in general incomplete it is typically assumed that it contains complete (with respect to the external world that T is trying to model) definitions for some of its relations and the observations O are on these

complete predicates. In this sense, the incompleteness of T is isolated in the abducible predicates on which we typically do not have direct information. The hypotheses H generated by abduction consists of formulae, usually facts, on these abducible predicates that aim to give an account (or an explanation) for the truth of the observations relative to the given theory T.
Abduction is thus viewed more as a way of reasoning with incomplete knowledge rather than as a mechanism for completing the knowledge in T. Its primary aim is to extract from the given theory T an explanation for the observation O thus providing a way to reason with T in the face of the new information O via the abductive entailment relation given in the previous section (Definition 2.1). Completing the theory T with the information in the explanation is often a secondary issue. For example, in the planning application of abduction described in the previous section, we are not primarily concerned with completing the theory by completing the definition of the abducible predicate *happens*. In fact, it is not appropriate (or feasible) to do so[4]. Furthermore, we could say that predicate *happens* is not to be learned i.e. it is not a concept that we apply inductive concept learning.

In induction the theory T may not in general have any complete predicates. The observations O are typically on predicates that are not defined by the theory and its primary task is to generate definitions for these in order to model the domain more accurately, or, in other words, complete the theory T. Their addition to T helps T to define more fully the interrelations between the different entities used to describe the domain. Hence the form of the generated hypothesis H can be as general as the language of T allows and it is typically a general rule (or rules). The addition of H to T aims in making T more complete in its general description of the problem domain.

Hence, the utility of abduction and induction is in general different. Abduction is used to generate a reason for the truth of the observation O in terms of hypotheses H, which are typically specific to the situation and individual objects at hand. Each separate observation will get its own explanation not necessarily related to the explanations of other observations, and not intended to account for other (latter) observations. On the other hand, induction is used when we want to synthesize the information conveyed by the observations into a hypothesis which can account for all the observations together in a common (uniform) way. Moreover this hypothesis will also be able to account for other new observations on different objects than those involved in the original observations. Thus, typically an abductive hypothesis consists of specific facts further describing the situation at hand relative to the given theory, while in induction a hypothesis consists of general rules pertaining to a whole class of situations.

It is important to note that we are arguing here for a restricted meaning of abduction where explanations are generated (extracted) directly from the given theory T *based only* on the general laws that can be found in T. This means that abductive explanations typically do not contain new general rules. In contrast to this restricted view of abduction the term abduction has been used in some cases (e.g. [31]) more liberally allowing the inclusion of new general laws in the explanation. We argue here that in these cases the reasoning involved has a *complex form* that can be broken down to a combination of more basic abduction steps, of the restricted nature described above, together with inductive generalization steps.
For example, we may say that, given the theory T = *"all bananas from Barbados are*

[4]Similarly, once we have or learned the general laws of motion in nature (e.g. Newton's laws) we are not concerned with learning the boundary conditions for each object (e.g. the initial position and velocity) in the world.

not sweet" we can explain the observation(s) O = *"all bananas in the shop are not sweet"* by the abductive explanation that "all bananas in the shop are from Barbados" (this example is taken from [31]). But we can also see this instead of a single process of abduction as comprising of two processes as follows. First we have a simple form of abduction to explain a typical case of "a banana being not sweet" and then we use (again a simple form of) induction to synthesize these different explanations, for each different banana in the shop together resulting in the final rule "all bananas in the shop are from Barbados". In the same spirit the term abduction has been used to name the process of generating new scientific theories, a process which can have a very complex form that can involve several separate more basic mechanisms.

These complex forms of reasoning named as abduction are thus viewed here as combinations of simpler and more basic computational mechanisms that include abductive and inductive inference as described above. In other words, the allowance in our explanations of general rules is not seen as an extension of the type of possible explanations that can be generated by abduction but rather as a form of synthesis (or generalization) together of several basic abductive explanations. This then helps us to separate and distinguish abductive and inductive processes.

It is clear from the above discussion that this issue of "general abduction" is already part of the central issue of this paper, namely the problem of integration of abduction and inductive generalization. In [1] an integrated abductive and inductive framework is proposed in which abductive explanations that may include general rules can be generated by incorporating an inductive learning method into abduction. In the next sections of this paper we will also see that it is possible to accommodate this "general form" of abduction within our integrated framework of abduction and induction.

As discussed above, abduction and induction are used for different purposes and can help complete our knowledge in T in different ways. But if we are just presented with a set of observations which one of the two (abduction or induction) should we use, when we are trying to assimilate this new information in our theory? It is not always clear which choice to take and how to make this decision. This problem is related to that of theory revision in machine learning. The relation of abduction and theory revision will be investigated in the next section.

In general, the question of how abduction and induction could be integrated and how they would cooperate, complement and affect each other is emerging as an important problem. For example, as learning becomes more knowledge intensive in the sense that it relies heavily on the background theory, abduction can help the learning task by translating the original training data into a more suitable form for generalization. On the other hand, advanced applications of abduction may need to include a form of generalization or theory revision, thus requiring the integration of a suitable learning method.

4 Integrating Abduction and Learning

In this section, we investigate how abduction can be integrated with the learning process. It is clear that this integration is only meaningful if the learning background theory, T_L, can be understood as an abductive theory, as described in the earlier sections, since otherwise abduction can not be applied. Hence the integration will be suitable in the context of analytic knowledge intensive learning (e.g. [41], [36]) where we have a "rich high-level" background theory.

There are two general dimensions along which the integration with abduction can help the learning process. In the first one abduction can help exploit more fully the high-level background theory available for learning. On the other hand, abduction can also help to handle possible incompleteness in the background theory thus providing greater flexibility in the learning process.

Let us initially discuss the second dimension of incorporation as this will also be needed in our study of the first one. We are thus concerned with the problem of how we can adapt the conventional learning framework so that we we can learn in cases where there is missing information from the background theory. More specifically, we are interested in learning (and later identifying new cases) under incomplete information on some of the background relations in the available learning background theory i.e. relations for which the absence of information does not necessarily mean falsity. In the context of abduction such relations will be designated as abducible predicates and although these are not defined in the theory they can be partially specified implicitly via a set on integrity constrains. The background theory for learning, T_L, is thus an abductive theory, as defined in the previous section 2, consisting of a tuple $< T, A, IC >$ where T is a set of rules (theory), A is a set of abducible predicates and IC a set of integrity constraints. Credulous entailment in this framework can be formulated as follows.

Definition 4.1 *(*abductive entailment*)*
A formula α is entailed by an abductive theory $AT =< T, A, IC >$, denoted by $AT \models_A \alpha$, if there is an abductive explanation of α from AT.

We can then define the *abductive learning* problem for a concept.

Definition 4.2 *(*abductive concept learning*)*
Given

- *A set of positive observations O^+ of a concept C*
- *A set of negative observations O^- of a concept C*
- *An abductive theory $AT =< T, A, IC >$ as background theory*

Find
A new abductive theory $AT' =< T', A, IC' >$ such that $AT' \models_A O^+$ and $AT' \not\models_A O^-$.

The new theory AT' will contain in T' rules for the concept C and possibly new integrity constraints in IC'. Notice that within the abductive learning framework, abducible predicates are allowed as conditions in the definition of the learned concepts. In abductive learning the requirement of the truth of the premises in the concept definition is replaced by the requirement of the consistency of the premises for the abducible ones. Since this is a weaker requirement, we need some control over the use of abducible predicates in the concept definitions. Without a control it is possible that "irrelevant" abducible predicates will be added in the concepts.

Learning systems should allow the use of abducibles only when it is really needed. This means preference should be given to predicates that are completely defined over the abducible predicates. An abducible can be used only if it is consistent with the

positive examples (however, there might be cases where this policy does not apply). Moreover, similarly to the case of classical inductive learning, as a general policy we may want to add an abducible predicate to the definition of a concept only if this predicate is inconsistent with some negative example. Consider the following example.

Example 4.3 *Let AT be an abductive theory $AT =< T, A, IC >$, where $A = \{male\}$, $T = \{parent(Tom, John), parent(Peter, Tony), parent(Jane, Jim), female(Jane)\}$, $IC = \{\leftarrow male(X), female(X)\}$. Let the set of example be $O^+ = \{father(Tom, John), father(Peter, Tony)\}$ and $O^- = \{father(Jane, Jim)\}$. Then the clause $father(X, Y) \leftarrow parent(X, Y)$ covers all positive observations but also covers the negative. The addition of the literal $male(X)$ in the clause has the effect of blocking the coverage of the negative observation but none of the positive observations. This is because the assumption $male(Jane)$ that is needed for the negative example to follow, violates the integrity constraint in the face of the information $female(Jane)$.*

The main advantage of extending the notion of coverage in learning with abduction is that this allows us (sometimes) to learn more accurate rules in the face of missing information and similarly classify later new examples that may be incompletely described.

The important issue of which abducibles should appear in the definition of a concept and how these can be computed will be studied quite generally in subsection 4.2 below. Before this though, in order to illustrate further the problem of abductive concept learning, we will discuss in subsection 4.1, the relation between EBL [7], [33] and abduction. In fact, we will see that EBL learning and its abductive extension is a special case of the general integration of abduction in the learning process. As such this discussion will also motivate partly our study of what we called at the beginning of this section as the first dimension of integrating abduction, in which abduction helps the learning process to exploit more fully the knowledge of the high-level background theory.

4.1 Explanation-based Learning and Abduction

A particular learning method that can be seen to fall within the abductive learning framework and for which the problem of which abducibles are relevant is easily resolved, is the explanation based learning (EBL, [7], [33]). We will indicate how EBL can be characterized in terms of abduction, and use it, as a special case of abductive concept learning, to illustrate and expand the basic ideas given in the previous section.

In particular, we are interested in the *theory specialization problem* ([20]), where we are given a domain theory T that contains the definition of a concept C, together with an example E of the target concept C' which we are told is more specific than C, i.e. E is also an instance of C. We also have the *operationality criterion*, which indicates which predicates from T can be used in the definition of C'. The new concept C' is defined by generalizing to all cases some of the specific explanations, in terms of the operational predicates, of the training example as an instance of the more general concept C defined in T.

Let us first see how abduction is related to EBL. Consider an EBL problem on a background theory T, a concept C' and an example E of this, and let P be a set of predicates expressing the operationality criterion (which we assume that are extensionally defined). Let P' be the set of facts on predicates from P contained in T. We

denote by T' the set $T' = T - P'$. Then the EBL problem can be captured trivially, as a problem of abductive concept learning described above, with background theory the abductive theory $< T', P', \emptyset >$. The additional information that the required concept C' is a specialization of C means that combinations of abducibles in P' that can be used in defining C' are restricted to only those that can form an explanation of E as an instance of the known concept C. If we want to, we can capture this by including in the abductive theory the integrity constraint $\leftarrow C'(X)$, *not* $C(X)$. Thus, the issue of relevant abducibles for the definition of C' is resolved by this extra knowledge that C' is more specific than C and the restriction that the definition of the new concept must use predicates from P'.

A new extension of the original EBL framework is the Abductive Explanation Based Learning ([5], [6], [47]) where assumptions about the truth of some abducible predicates, A, can be made. These assumptions are used in order to complete proofs (explanations) of the example that would otherwise fail. These extensions are captured similarly as abductive concept learning with background theory $< T', A \cup P', \emptyset >$.

This view of EBL as an abductive learning method can be beneficial in several ways. Let us see how this helps us with the treatment of negative information. Negative information, in the form of negative examples, can be used in EBL (e.g. see [5], [6]) in the same way it is used in empirical learning, namely for the specialization of the hypothesis generated by the positive examples. This use of negative information is reflected in definition 4.2 where we require that the negative examples must not be abductively covered by the concept definition. However due to the presence of abducible predicates in concept definitions this requirement may be too strong in some cases. Consider the following example.

Example 4.4 *Let T be the following theory containing the rules*
$relative(X,Y) \leftarrow relative(X,Z), relative(Z,Y)$
$relative(X,Y) \leftarrow father(X,Y)$
$relative(X,Y) \leftarrow mother(X,Y)$
$father(X,Y) \leftarrow parent(X,Y), male(X)$
$mother(X,Y) \leftarrow parent(X,Y), female(X)$
$parent(Tom,John)$, $parent(John,Alice)$, $parent(Jane,Mark)$,
$parent(Mark,George)$
Furthermore, parent, male and female are operational predicates, male and female are abducibles and the integrity constraint $\leftarrow male(X), female(X)$ is part of the theory.
Now assume that we are given the positive example $grandfather(Tom,Alice)$ of the concept grandfather assumed to be a more specific concept than relative. Since we have no information about whether the individuals involved in the example are male or female we can assume any of them in different explanations, and obtain, by explaining "relative(Tom, Alice)", the following definition for grandfather.
$grandfather(X,Y) \leftarrow parent(X,Z), parent(Z,Y), male(X), male(Y)$
$grandfather(X,Y) \leftarrow parent(X,Z), parent(Z,Y), male(X), female(Y)$
$grandfather(X,Y) \leftarrow parent(X,Z), parent(Z,Y), female(X), male(Y)$
$grandfather(X,Y) \leftarrow parent(X,Z), parent(Z,Y), female(X), female(Y)$
Suppose now that we are also given the negative example $grandfather(Jane,George)$. Note that all of the rules for grandfather can explain the negative example. Hence, according to definition 4.2 all of rules need to be specialized. The only possible specialization, given the information at hand, is the removal of the rules.

The previous example illustrates the difficulties in handling negative information in A-EBL (and the abductive learning framework, in general). Definition 4.2 requires that no negative example should have an explanation in the theory that is constructed. Since explanations can involve assumptions on the abducible predicates, they are more easily constructed and hence the impact of negative examples on the theory can be too strong. We could relax the second condition of abductive concept learning by replacing this with the condition that $AT' \models_A not\ (O^-)$, where $not\ (O^-)$ denotes the negation as failure literal corresponding to O^-, and using naf-explanations. In the example, "$not\ grandfather(Jane, George)$" is consistent with all the above rules since we can assume for example that "$not\ male(Jane)$" for the first rule and "$not\ female(Jane)$" for the fourth rule[5]. In either case, such negative examples should not have any significant effect as they do not carry any information different from that in the positive examples. Therefore, together with the negative examples other information about the abducible predicates involved should also be provided. For instance, in the previous example if the additional information $female(Jane)$ is known, then the concept can be (correctly) specialized by removing the last two rules in the definition of $grandfather$. This is in accord to our earlier discussion where it was preferable for an abducible condition to be consistent with all positive examples and inconsistent with at least one negative example.

So how do we specialize the rules of the theory? One possibility is to add literals to the rules. This introduces an empirical element in EBL, since the literals to be added do not belong to the explanation constructed by the EBL process. We prefer non-abducible over abducible predicates and the general policy, discussed above, for the addition of abducible in the rules applies.

Another possibility for specializing a theory is to remove the rules, as we did with the last two rules of the theory of the example 4.4. Moreover, we might want to record and generalize the information of the negative examples and use this in subsequent steps of the learning process. This can be done by constructing an integrity constraint[6] restricting the concept when the abducible information of the negative example holds. In effect we achieve the removal of the rule by blocking it with an integrity constraint. For instance, the integrity constraint that can be constructed by $O^- = \{grandfather(Jane, George)\}$ in the previous example, is $\leftarrow grandfather(X, Y), female(X)$. This, apart form having the effect of blocking the rules for $grandfather(X, Y)$ with $female(X)$ in their body, will also disable the assumption $female(X)$ for any individual that is known to be the grandfather of another individual. Hence, through the use of integrity constraints, negative examples may have non-local effects on the theories constructed.

Effecting the removal of a rule via an integrity constraint opens the new possibility of using integrity constraints to partly specialize rules, i.e. without having the effect of blocking all instances of the rule. These constraints aim to impose restrictions on the assumptions that can be made about the abducibles in the rules that need specialization. The following example illustrates these possibilities.

Example 4.5 *Assume that using EBL we have obtained the following definition of the good-manager predicate using the available positive examples*
$good\text{-}manager(X) \leftarrow works\text{-}hard(X), well\text{-}trained(X)$

[5]If we also have the integrity constraint that $\forall x(male(x) \vee female(x))$ then we need to drop one of these rules.

[6]For the treatment of integrity constraints within learning and theory revision systems see also [2],[12], [13], [38].

Also assume that the integrity constraint $\leftarrow organizer(X), late\text{-}meetings(X)$ *is part of our theory, where the predicate organizer is an abducible. Let* $good\text{-}manager(John)$ *be a negative example, and* $works\text{-}hard(John)$, $well\text{-}trained(John)$ *and* $late\text{-}meetings(John)$ *be known facts about John. Then we could specialize our theory by adding the abducible* $organizer(X)$ *to the definition of the concept good-manager, or add the integrity constraint* $\leftarrow good\text{-}manager(X), late\text{-}meetings(X)$.
On the other hand if $good\text{-}manager(Tom)$ *was our negative example and* $works\text{-}hard(Tom)$, $well\text{-}trained(Tom)$, $bad\text{-}leader(Tom)$ *where facts known about Tom, then the only way to specialize our theory, without removing the rule, would be to add the new constraint* $\leftarrow good\text{-}manager(X), bad\text{-}leader(X)$.

Let us elaborate a little further on the addition of integrity constraints to the theory when negative information is available. First, we note that we may want to add to the denial of the integrity constraint all the abducible predicates which are known to be true for the individuals involved in the negative example. This will lead to the least specialization of the learned theory.
Assume now that several negative examples for a concept are given. The first choice we have is to handle each negative example in isolation from the others and create an integrity constraint for each of them. The second choice is to synthesize and generalize the information in the negative examples like we do for positive examples. Consider the next example.

Example 4.6 *Let* $good\text{-}manager(X) \leftarrow organizer(X), well\text{-}trained(X)$ *be the rule learned from the positive examples,* $E^- = \{good\text{-}manager(Tom), good\text{-}manager(John)\}$ *be the negative examples and* $T = \{leaves\text{-}early(Tom), long\text{-}brakes(John), lazy(X) \leftarrow leaves\text{-}early(X), lazy(X) \leftarrow long\text{-}brakes(X), lazy(X) \leftarrow comes\text{-}late(X)\}$, *be part of the domain theory, where leaves-early, long-brakes comes-late are abducibles.*
The two choices we have are either to add two constraints, namely $\leftarrow good\text{-}manager(X), leaves\text{-}early(X)$ *and* $\leftarrow good\text{-}manager(X), long\text{-}brakes(X)$ *or to generalize and add the constraint* $\leftarrow good\text{-}manager(X), lazy(X)$.

This in effect is equivalent to learning part of the negative concept for *good-manager* ([11]) together with a priority of the negative rule over the positive ([15]). Notice that up to this point all the integrity constraints added to the theory are local to the concept to which the associated negative examples are related. In some cases we might want to go a step further in the generalization of the integrity constraints and induce from constraints specific to particular concepts constraints that are generally applicable. This can be done when there is evidence across the concepts that the abducible predicates are incompatible. However, since these generalizations have global effects on the learned theory, they should be generated with care.

The abductive view of EBL can also help us address the multiple EBL problem ([20]), the identification of the target concept from multiple examples. This will be covered in the next section in the more general context of abduction in learning.

4.2 Abduction in Inductive Learning

We have seen how abduction can help learning to overcome difficulties with missing information from the background theory by providing an alternative way of reasoning (coverage) appropriate for situations of incomplete background knowledge. Abductive

methods, such as the use of integrity constraints, can be employed to adapt the learning methods to this more flexible notion of coverage.

Let us now investigate what we called in the beginning of this section the first dimension of integration, where we want to see how abduction can be *incorporated* inside the learning process itself. The possibility of incorporating in a natural way abduction in the inductive learning process is evident in Peirce's informal characterization of the inferences of abduction and induction (see section 3). If induction requires a "case" as given knowledge (or input) and abduction produces "cases" then clearly abduction can play a preparatory role for the inductive generalization. Given a background theory for learning in the form of an abductive theory, abduction can be used to bring the given learning data into a more suitable form for inductive generalization. If we are given a set of observations O as training data then an abductive explanation that gives an account for these observations represents, informally, a better and deeper understanding of this data and hence one could expect that this would have positive effects on the overall learning.

The general pattern of this integration of abduction and induction is a two-level learning process where abduction is used first to prepare, via its abductive explanation of the various observations, suitable data for the inductive generalization at the second level of the process. The observations explained in this process involve individuals in the examples or individuals "related" to them, and are either present in the background theory or are given with the examples. Finding the relevant, wrt a particular learning problem, observations is an important problem but we will not address this further in this paper. For the purposes of this paper we assume that the input for the learning process are tuples $< E, O >$ where E is an example for the concept to be learned and O a set of relevant observations.

We have already seen in the previous subsection EBL as an example of this integration. The explanation step of EBL is an abductive task explaining the given example in terms of the operational predicates. This explanation is then generalized directly to all cases. In fact, this is a very special case of the possible integration of abduction in the learning process. More generally, in the first level, abduction can *extract*, via its abductive explanations of the given observations, a suitable part of the background (abductive) theory to be used as background knowledge for the second-level of generalization. We refer to this particular incorporation of abduction and learning as the **extraction-case**.

The extraction case applies when we are given a specific learning problem with a set of training examples for a concept to be learned, each example been described in an associated set of observations, O, in terms of several high-level characteristics or attributes. Then by first explaining these observations for each example we are in effect extracting from the general and high- level theory a suitable part to play the role of a background theory for the inductive generalization. This extraction of relevant background information can be quite useful and sometimes it is necessary for the generalization since only if we explain the observations of the training data do we begin to see the common properties amongst the different examples. In fact, the duality between abduction and induction as discussed in the previous section supports this extraction-case of incorporation of abduction in learning and suggests the following simple methodology:

a explain the observations for each example

b find the (skeptical) abductive conclusions under these explanations and

c apply an inductive generalization algorithm with background knowledge the one generated in step (b).

Before refining this further let us illustrate these ideas by an example.

Example 4.7 *Consider the following general theory T of family relations:*
$father(X,Y) \leftarrow parent(X,Y),\ male(X)$
$mother(X,Y) \leftarrow parent(X,Y),\ female(X)$
$son(X,Y) \leftarrow parent(Y,X),\ male(X)$
$daughter(X,Y) \leftarrow parent(Y,X),\ female(X)$
$child(X,Y) \leftarrow son(X,Y)$
$child(X,Y) \leftarrow daughter(X,Y)$
$sibling(X,Y) \leftarrow parent(Z,X),\ parent(Z,Y)$
$brother(X,Y) \leftarrow sibling(X,Y),\ male(X)$
$sister(X,Y) \leftarrow sibling(X,Y),\ female(X)$
$grandparent(X,Y) \leftarrow parent(X,Z),\ parent(Z,Y)$
with abducible predicates $A = \{parent, male, female\}$ *and integrity constraints* $IC = \{\leftarrow male(X), female(X)\}$.
Suppose now that we are given two positive examples $< e_1^+, O_1 >, < e_2^+, O_2 >$ *of the concept to be learned, "uncle(-, -)", as follows:*
$e_1^+ : uncle(Bob, Tom) : O_1 = \{sibling(Bob, Jack), father(Jack, Tom)\}$
$e_2^+ : uncle(John, Ann) : O_2 = \{son(John, Rob), father(Rob, Helen), mother(Helen, Ann)\}$

According to the methodology suggested above we first use abduction to explain these observations O_1, O_2 for the examples. We get
$\Delta_1 = \{parent(sk_1, Bob), parent(sk_1, Jack), parent(Jack, Tom), male(Tom)\}$
$\Delta_2 = \{parent(Rob, John), parent(Rob, Helen), parent(Helen, Ann), male(John),\ female(Helen)\}$
where sk_1 denotes an unknown constant[7]. The next step is to add these to the theory T to find the abductive consequences. We then see that we have, apart from these facts in the explanations and in the observations, other new consequences such as $sibling(John, Helen), brother(John,\ Helen)$. All these together then form a relevant background theory on to which we can base our generalization to induce a definition for the desired concept of "uncle". This could generate the rule

$$uncle(X,Y) \leftarrow parent(Z,X), parent(Z,W), parent(W,Y) \text{ or}$$
$$uncle(X,Y) \leftarrow sibling(X,W), parent(W,Y)$$

as the final result of the learning process. Each of these rules does indeed cover the given examples under our notion of abductive coverage with respect to the original given background abductive theory.

Note that the generalization was only possible because we first explained the given information in the observation as otherwise we are not able to see the "common pattern" in the different examples. To emphasize this point we could have been given a third positive example as $e_3^+ : uncle(Mike, Chris) : O_3 = \{grandparent(George, Chris),\ son(Mike,\ George)\}$. Again only if we explain these characteristics with
$\Delta_3 = \{parent(George, Mike), parent(George, sk_2), parent(sk_2, Chris), male(Mike)\}$
we could see (at the generalization step) that this example also falls within the same

[7]For the technical details of this form of *constructive* abduction see [8], [23].

definition as found above. Note also that some parts of the explanations e.g. the male or female assumptions do not generalize, and thus do not appear in the final rule, as they are not "common" to all the examples.

In the above example, the application of the proposed method was simple as we had a *unique* abductive explanation for each training example. This will not be true in general as a set of observations may have *multiple* explanations because of the incompleteness in the abductive theory. This problem remains even after integrity checking and the application of various preference criteria such as minimality, as discussed in section 2. Let us see how this affects the above proposed methodology for incorporating abduction into learning. Initially we note that the method continues to apply as we can still compute for each example, in step (2), the skeptical abductive consequences implied by the observations of the example, but we will see below ways in which this could be refined to address better this problem.

To illustrate the problem consider again the above example with the third positive example changed to:
$e_3^+ : uncle(Mike, Chris) : O_3 = \{grandparent(George, Chris), child(Mike, George)\}$
Now O_3 has two possible abductive explanations, namely
$\Delta_{3.1} = \{parent(George, Mike), parent(George, sk_2), parent(sk_2, Chris), male(Mike)\}$
$\Delta_{3.2} = \{parent(George, Mike), parent(George, sk_2), parent(sk_2, Chris), female(Mike)\}$

As mentioned above we can still apply the same method and indeed in this case the final result will be the same. The differentiating "male(Mike)" and "female(Mike)" assumptions do not have any effect. This occurs in the same way that different[8] assumptions across different examples do not have any direct effect on the final rules. But are there ways that can help us select "good" choices amongst the multiple explanations?

One possibility is as follows. Remembering that we are trying to synthesize the different examples we may prefer one explanation for a particular example over its others, if this, informally speaking, ties up closer with the explanations for the other examples. In other words, we could prefer to select explanations in such a way that these are "similar" (i.e., involve the same abducible predicates) across the different examples so as to maximize the degree of synthesis at the generalization step. To illustrate this consider again the above example of learning the concept "uncle" with the three given training examples as:
$e_1^+ : uncle(Bob, Tom) : O_1 = \{brother(Bob, Jack), father(Jack, Tom)\}$
$e_2^+ : uncle(John, Ann) : O_2 = \{son(John, Rob), father(Rob, Helen), mother\ (Helen, Ann)\}$
$e_3^+ : uncle(Mike, Chris) : O_3 = \{grandparent(George, Chris), child(Mike, George)\}$.
In this case, since both the unique explanations for e_1^+ and e_2^+ contain the assumption $male$ on the respective first object on which the concept is defined (i.e. Δ_1 contains $male(Bob)$ and Δ_2 contains $male(John)$) we could prefer amongst the two explanations for the third example the one which similarly contains the assumption $male(Mike)$. Then selecting this as the explanation for the third example and following the general method we learn the rule

$$uncle(X, Y) \leftarrow brother(X, W), parent(W, Y).$$

Note that in effect, the first two explanations are imposing a preference on selecting from the explanations of the third example. This discussion motivates the following

[8] By "different" here we meant different only at the level of the predicate symbol, not at the actual instance level of the assumption as these will trivialy be different.

definition that captures more formally the intuitive notion of selecting explanations that make the overall set of explanations more compact[9]. We first give the definitions of an *explanation set* and the *generic form* of an explanation set that will be used in the following.

Definition 4.8 *An* explanation set *S for a set of observations O is a set of sets of abducibles, such that every element of S is an abductive explanation of an observation in O.*

Definition 4.9 *Let S be an explanation set of a set of observations and I an identification function[10] of the constants appearing in the observations. The* generic form *of S under I, denoted by S^*, is the set of hypotheses obtained by taking the union of the explanations in S after applying I to the constants.*

In the extraction case where the observations are related to positive examples of a concept c to be learned, a natural identification function is given by identifying all constants that occur in the same position in the predicate c in the examples by a new constant, and identifying all other constants occuring in the explanation set by the same new constant. We call this identification function *extraction identification function.*

Definition 4.10 *Let E^+ be a set of positive examples for a concept c with a set of associated observations O^+. An explanation set S^c for O^+ is a* **compact explanation set** *for E^+ if for any other explanation set S for O^+, $S^{c^*} \subset S^*$ holds, where S^{c^*} and S^* are the generic forms of the explanation sets S^c and S respectively, under the extraction identification function.*

We note that this notion of compactness is similar to the common subtree method traditionally used in multiple explanation-based generalization (mEBG, [20]). This method uses the largest subtree that is shared by the explanation trees of all the examples and then applies the EBL generalization technique to obtain a general rule.
In the example above the explanation set, S, where we choose the first explanation for the third example is compact as S^* would contain "$male(*_1)$", where $*_1$ replaces the constants corresponding to the first argument of the concept "*uncle*", whereas the second choice for the third example leads to an explanation set, S', with S'^* containing "$male(*_1)$" and "$female(*_1)$" together with the same assumptions on the "*parent*" relation as in S^*.

Part of the problem of handling multiple explanations is that of learning with disjunctive background information. In general there may be many compact explanation sets. We then have two options: either to split the learning process and generate a different rule for each explanation set separately (and hence a disjunctive definition) or to consider them all together and use the second step of the method of abductive entailment to generalize.

We can now refine the methodology proposed above as follows:

a.1 explain the observations for each example

a.2 construct the compact explanation sets

[9]This definition is just one particular way that we can formalize this notion.

[10]An identification function for a set of constants C is a mapping of the constants in C to some new constants.

a.3 CHOOSE a subset of the compact explanations

b find the (skeptical) abductive conclusions under the chosen subset of explanation sets, and

c apply an inductive generalization algorithm with background knowledge the one generated in step (b).

Let us now study how negative examples might be used in this extraction-case of integration of abduction and learning and how they can affect the above methodology. Returning to our "uncle" example above consider a negative example as follows:
$e_4^- : uncle(Mary, Steve) : O_4 = \{sister(Mary, Bill), father(Bill, Steve)\}$.
We can again explain the associated observations O_4 to give
$\Delta_4 = \{parent(sk_3, Mary), parent(sk_3, Bill), parent(Bill, Steve), male(Bill), female(Mary)\}$.

Such explanations of negative examples can then be used to reduce the number of multiple explanations in other positive examples by rendering some of these inappropriate. In this case, the "$female(Mary)$" condition can be used to reject the second explanation in the third example as it contains the same assumptions on the first object of the concept to be learned and hence any explanation set S that is build using this would also explain, when we consider its generic form S^*, the negative example. So in general we require in step (a.2) of the methodology that an explanation set S must not explain with its generic form S^* the observations of any negative example. Alternatively, we could also apply similar techniques to those discussed in subsection 4.1 and use information from known integrity constraints[11] to render the explanations of the negative examples inconsistent e.g. in the example above we could notice that "$female(Mary)$" would be inconsistent if we adopt an explanation set S where $male(*_1)$ is in S^* and thus again the negative example will have the effect of preferring to resolve the multiple explanations in the third example in favor of the explanation that contains the "$male$" assumption for the first object.

In general, we need to be able to separate which combinations of explanations are consistent in order to construct explanation sets with consistent generic forms. This task can be facilitated by known techniques developed in Abductive Logic Programming and ATMS. The integrity constraints implicitly impose these restrictions on the combinations of explanations. This implicit information can be made explicit by adopting naf-explanations so that it can be used more effectively. The naf abductive explanations include information about what abducibles assumptions must be *absent* for the integrity constraints to be satisfied. In the *uncle* example the assumption "$male(Bob)$" in the first example would also come with the assumption "$not\ female(Bob)$", due to the known integrity constraint, and then this extra information can be used to prefer the first explanation in the third example when we are constructing explanation sets for all positive examples.

It is interesting to note that from the point of view of abduction the learning task feeds back into abduction, through notions like that of a compact explanation set, a preference criterion for multiple abductive explanations. An explanation for one observation, amongst a set of different observations that have been grouped together for some reason, can affect the explanation of other observations in this set even though these may refer to different objects.

[11] We could also generate new integrity constraints that would exclude abductive coverage of the negative examples.

4.3 Abduction in Theory Revision

Theory revision generalizes inductive rule learning by allowing the background theory itself to be modified. Theory revision systems (see [53] for a recent review) use, apart from induction as in concept learning, a variety of other operators like antecedent specialization and generalization, rule deletion, rule synthesis etc. In this section we show possible ways to carry over the integration of abduction and induction as discussed in earlier parts of this paper, to the case of theory revision, and discuss how abductive techniques can be used to help other theory revision operators. We will see that the basic methodology developed above extends naturally to the more general context of theory revision.

In fact, abduction has already been used in the context *Knowledge Assimilation* ([22], [26]) and theory revision ([2], [10], [14]) where a system (or an agent) is given different observations about the world without these necessarily constituting an explicit learning problem and is required to assimilate them into its current knowledge base. Trivially, an abductive explanation offers a limited form of revision by adding the previously unknown explanation into our theory. Hence abduction can be one of the operators of theory revision systems. In this case, abductive explanations of positive examples are added to the background theory and in this way coverage of these examples is achieved. Also abduction has been used in theory revision [39], [52] as a method to identify the specific parts of the theory that need revision.

Let us first study how the theory revision process can be affected by abduction by examining how some of the revision operators can be adapted when the background theory is an abductive theory. Apart from generating new rules for incomplete predicates in theory revision, we may want to (1) refine existing rules or (2) synthesize rules. So assume that we are at some stage where our abductive theory contains an incomplete definition for the predicate of a concept that we want to learn and these rules may be incorrect. Our task is to complete the definition of the concept by refining the existing rules and/or generating additional rules. As we get more information on this concept either directly as new examples of this concept or indirectly through abductive explanations of new observations we have the following possibilities for revising and refining the existing rules.

(1) A new positive example is already covered (abductively according to the definition in section 4) by the existing rules. Then we can just add the (best) explanation in the theory i.e. abduction becomes one of the theory revision operators as mentioned above. Of course, if this is repeated we may want, as these explanations build up and we "notice" that a suitable learning problem is developing, to synthesize these explanations and learn new rules using the extraction-case methodology on this "emerging" learning problem. We could also, if we want, apply directly the extraction-case method on such examples but it is most likely that the rule learned will be the same as the rule that covers these examples. Note though that if the different examples are covered by different rules then the extraction-case may produce rules which are not the same as the existing one, for example it could produce a single rule that subsumes the previous ones. This is the case of synthesis that we will discuss again below in case (4).

(2) A case where it is useful to apply the extraction-case method directly in theory revision, is when a new positive example is not covered (abductively) by the existing rules. In this case, we may be able to refine the existing rule (e.g. [1], [39]) so that this does not hold any more. To show how we would employ the extraction-case method for the refinement of rules in this case consider the following example.

Example 4.11 *Suppose that we have the "family" abductive theory as given above in example 4.7 together with the rule*
$uncle(X,Y) \leftarrow brother(X,W), father(W,Y)$
for the concept $uncle(-,-)$ *that we are learning and we are given the new positive example* $< e^+, O >$, *where*
$e^+ : uncle(Bob, Steve) : O = \{brother(Bob, Mary), mother(Mary, Steve)\}$.

The existing rule can not cover this example since it would require the abductive assumption $male(Mary)$ which is not consistent with the observation $mother(Mary, Steve)$ and $female(Mary)$ which is abductively entailed by this. This may be understood as an indication that the implicit condition $male(W)$ that the uncle rule contains should be dropped. Of course, this does not mean that we should drop the *male* condition for the rule for $farther$ which we assume is correct.

A general way to carry out such refinements of the rule is to call on to the extraction-case methodology as follows. Note that the existing rule carries in effect an (compact) explanation set for a set of previous examples of the concept that may have generated this rule. In our example the rule carries the explanation set Δ with generic form Δ^* as
$\Delta^* = \{parent(*_1, *_x), parent(*_1, *_2), parent(*_2, *_y), male(*_x), male(*_2)\}$.
We then take this together with the explanation(s) for the associated observations of the new positive example, in the example
$\Delta' = \{parent(sk_1, Bob), parent(sk_1, Mary), parent(Mary, Steve), male(Bob), female(Mary)\}$, and apply the extraction-case methodology as given in the previous section regarding these as the output of step (a.1). We can see that in the example the result of this is to produce the rule

$$uncle(X,Y) \leftarrow brother(X,W), parent(W,Y).$$

In effect this has replaced the $father$ condition in the initial rule by the more general one of *parent*, by first breaking down the rule and then rebuilding this using the extra information of the new example.

(3) A new negative example is covered by some existing rule. This rule needs to be specialized and there are several ways that this can be done as already discussed in section 4 under the general study of abductive coverage.

(4) Finally, let us discuss how synthesis of rules ([39], [52]) can be achieved using the integration methodologies developed above. Here we assume that the existing rules that we have for the concept to be learned are correct but they do not define completely the concept. We first translate each existing rule to a "generic" explanation set in the same way described above in case (2). Then again we have a set of explanation sets and we can apply the extraction-case methodology directly from step (a.3) by choosing non-singleton (otherwise this would just generate the same rules back) subsets of these explanation sets.

An example that illustrates these ideas of synthesis is as follows.

Example 4.12 *Let* T_A *be the abductive theory,*
$tall(X) \leftarrow French(X)$
$tall(X) \leftarrow Italian(X)$
$European(X) \leftarrow French(X)$
$European(X) \leftarrow Italian(X)$

$European(X) \leftarrow German(X)$
...
where the predicates "French", "Italian" and "German" are abducible, the predicate "tall" is not completely defined by these rules but "European" is fully defined in T. New positive examples for "tall" can generate via the rules explanations of the form $\Delta_1 = \{French(*)\}$ *or* $\Delta_2 = \{Italian(*)\}$ *and then since these give as an abductive consequence (see step b of the general methodology) "European(*)" we will get the new single rule* $tall(X) \leftarrow European(X)$ *generalizing (and subsuming) the two existing rules.*

Let us now turn to the question of how abduction can be used in theory revision as a useful technique irrespective of whether the background theory has a natural abductive form stemming from its representation of the problem domain. One important instance of this use, is when abduction helps in the identification of places where the revision of the theory can be performed. For example, in ([39], [52]) abduction can be used to find assumptions under which the positive examples are covered. Once these assumptions have been found, they are used by the system for the computation of the revised theory. Similarly, in [1], [2] abduction is used as a technique to help identify useful learning problems "hidden" in the new information (observations) given to the system. We will concentrate mainly on this particular use of abduction for the identification of learning problems and we will refer to this case of the integration of abduction and theory revision (or learning) with the name **identification case**.

In the identification case, similarly to the extraction case, we have a two level learning process where now in the first level abduction identifies the actual training data of some generalization problem. This is done by translating seemingly unrelated observations into information that has some pattern and can be identified as a generalization problem. Then the result of this generalization is effectively a way of synthesizing the understanding, by the current knowledge base, of some of the different observations. The role of abduction is again important as only if these observations are explained (or understood) at a "deeper level" we can notice the interrelations between the different observations and thus recognize that there is room for synthesis and generalization.

As mentioned above the identification case can be used irrespective of whether the background theory comes in an abductive form or not. This is turned into an abductive theory for technical purposes by designating for example the incompletely or incorrectly defined predicates of the theory as abducible predicates. Since we may already have some abducible predicates it is better to separate these two classes, as it is done in [1], by calling this new class of abducibles *inducibles* [12]. The appearance of inducible assumptions in the abductive explanations provides training examples for learning or refining the definition of inducible predicates. This view has been proposed and studied in some detail in [2]. To illustrate these ideas consider the family example again where now amongst the other rules and facts of the background abductive theory T we have the rules
$son(X,Y) \leftarrow child(X,Y), male(X)$
$daughter(X,Y) \leftarrow child(X,Y), female(X)$,
the abducible predicates are $A = \{parent, male, female\}$ and the predicate *child* is inducible with no rules for it in T. Suppose that we are presented with the new information given by the observations $O_1 = \{son(John, Bob)\}$ and $O_2 = \{daughter(Mary, Tim)\}$.

[12] As argued in section 3 in some cases it may not be appropriate to learn rules for the abducibles that come from the representation of the problem (e.g. the "happens" predicate in the planning example).

We can assimilate each one of these separately by the following abductive explanations
$\Delta_1 = \{child(John, Bob), male(John)\}$.
$\Delta_2 = \{child(Mary, Ann), female(Mary)\}$.
We can then notice that this new information has generated a possible learning problem for the concept $child(-,-)$ with positive examples
$e_1^+ : child(John, Bob)$, $O_1^+ \supseteq \{male(John)\}$
$e_2^+ : child(Mary, Ann)$, $O_2^+ \supseteq \{female(Mary)\}$.
If we now have other direct information in our background theory on these objects we may be able to use this to revise our theory by learning a definition for the concept $child(-,-)$. Alternatively, we may have other observations in the stream of new information acquired on these objects. We can then take these as associated observations with these positive examples and apply the extraction-case methodology of the earlier subsection. Hence if we were also given in O_1^+ that $father(Bob, John)$ and in O_2^+ $mother(Ann, Mary)$, then we would, after abductively explaining these and applying the method, get the rule $child(X, Y) \leftarrow parent(Y, X)$. Note that in each example e_1^+, e_2^+ the rest of the explanation Δ_1, Δ_2 above can be used as additional information on the objects of interested and thus are also added to the associated observations. Their explanation in step (a.1) of the method will be trivially themselves as these are abducibles.

Finally, we mention that the use of abduction as a technique in theory revision can be enhanced by the advance abductive techniques developed in the area of Abductive Logic Programming (see section 2). Adopting these techniques can help extend the type of theories considered in theory revision e.g. allow negation or negation as failure in the theory or in general a non-monotonic theory possibly with integrity constraints as the theory to be revised. Also they can help us analyze better negative information, using the notion of *naf-explanation* described in section 2, allowing us to generate also negative examples for the learning problem that we are identifying. For example, if we are given the observation $\neg parent(Jack, Rob)$, assuming that our background abductive theory contains the rule $parent(X, Y) \leftarrow child(Y, X)$, the (naf) abductive explanation for this would contain $not\ child(Rob, Jack)$ we so would generate the negative example $e^- : child(Rob, Jack)$.

5 Conclusions

We have studied the relationship between abduction and inductive learning. Although the two inferences of abduction and induction may share the same basic formal specification they also have important differences that distinguish the two. These differences originate mainly from the different aims and utility that each one of these have. Despite these differences we have seen that there is a rich ground of interaction between abduction and learning both at the semantic specification level and at the computational level. We have introduced a basic integrated framework of abductive concept learning that allows learning from incomplete background theories. Associated with this we have also studied a general computational methodology for integrating abduction in the inductive generalization or more generally in the theory revision process. Abduction provides a first-level in which it extracts and prepares, from the given abductive background theory a part of knowledge suitable for the generalization or revision step that follows. Thus the integration of abduction into learning enhances the knowledge intensive nature of learning, helping us to exploit more fully the rich information that

might be contained in the background theory. We have also that abduction can also provide the basis for non-monotonic learning frameworks providing a uniform technique to handle negation as failure, incomplete predicates and integrity constraints.

Many of the ideas presented here need further study. We need to examine further suitable notions of generic and compact explanation sets that can help us handle more effectively multiple abductive explanations. We also need to address the important problem of how we choose for each positive training example the associated relevant observations to be explained. In this paper we concentrated primarily on how abduction can affect learning. It is interesting to examine further, in the context of knowledge assimilation, how learning can affect back abduction in a framework (like [2]) where abduction and learning (or theory revision) are fully interleaved.

In general, more work is needed to understand better how various learning criteria, theoretical and practical, used in conventional learning systems are affected (and therefore must be adapted) in the case of abductive concept learning. We need to develop practical learning systems based on these ideas of integration that can be used to test and refine these and evaluate the extend to which this integration of abduction and learning is indeed useful.

References

[1] H. Ade and M. Denecker. Abductive Inductive Logic Programming. *IJCAI-95.*

[2] H. Ade, B. Malfait and L. De Raedt. RUTH: an ILP Theory Revision System. *ISMIS94*, 1994.

[3] M. Bain and S. Muggleton, Non-monotonic learning. In: J.E. Hayes-Michie and E. Tyugu, eds., *Machine Intelligence 12.* Oxford University Press, 1990 bitembain

[4] F. Bergadano and A. Giordana. Guiding Induction with Domain Theories. In *Machine Learning: An AI Approach*, Vol. III.

[5] W. W. Cohen. Abductive Explanation-Based Learning: A Solution to the Multiple Inconsistent Explanation Problem. *Machine Learning*, Vol. 8, pp. 167-219, 1992.

[6] W. W. Cohen. Incremental Abductive EBL. *Machine Learning*, Vol. 15, pp. 5-24, 1994.

[7] G. DeJong and R. Mooney. Explanation-based Learning: An Alternate View. *Machine Learning*, Vol. 1, pp. 145-176, 1986.

[8] M. Denecker and D. De Schreye. SLDNFA: an abductive procedure for normal abductive programs. *Proc. Intern. Conference and Symposium on Logic Programming*, pp. 686–700, 1992.

[9] L. De Raedt and M. Bruynooghe. On negation and three-valued logic in interactive concept learning. *Proc. of the 9th European Conference on AI, ECAI-90*, 207-212, 1990.

[10] L. De Raedt. *Interactive Theory Revision: An Inductive Logic Programming Approach.* Academic Press, 1992.

[11] L. De Raedt and M. Bruynooghe. On negation and three-valued logic in interactive concept learning. *Proc. of the 9th European Conference on AI, ECAI-90*, 207-212, 1990.

[12] L. De Raedt and M. Bruynooghe. Belief Updating from Integrity Constraints and Queries. *AI Journal*, 53, pp. 291-307, 1992.

[13] L. De Raedt and M. Bruynooghe. A Theory of Clausal Discovery. *IJCAI-93.*

[14] L. De Raedt and M. Bruynooghe. Interactive Theory Revision. In *Machine Learning: A multistrategy approach*, R. Michalski and G. Tecuci, 1994, Morgan Kaufmann.

[15] Y. Dimopoulos and A. Kakas. Learning Non-Monotonic Logic Programs: Learning Exceptions. *Machine Learning: ECML-95* , (Proc. European Conf. on Machine Learning, 1995), N. Lavrac and S. Wrobel (eds), LNCS 914, pp 107-121, Springer Verlag.

[16] B. Duval. Abduction for Explanation-Based Learning. *European Working Session on Learning, EWSL-91*, LNCS 482, 348-360, Porto, 1991.

[17] K. Eshghi and R.A. Kowalski. Abduction compared with negation by failure. *Proc. 6th Inter. Conference on Logic Programming*, 234–255, Lisbon, 1989.

[18] C.A. Evans and A.C. Kakas. Hypothetico-deductive reasoning. *Proc. Inter. Conference on Fifth Generation Computer Systems*, 546–554, Tokyo, 1992.

[19] P. Flach. *Conjectures: An Inquiry concerning the Logic of Induction*, ITK dissertation series, 1995.

[20] N. S. Flann and T.G. Dietterich. A Study of Explanation-Based Methods for Inductive Learning. *Machine Learning*, Vol. 4, pp. 187-266, 1989.

[21] A. C. Kakas and P. Mancarella. On the relation of truth maintenance and abduction. *Proc. of the 1st Pacific Rim International Conference on Artificial Intelligence, PRICAI'90*, Nagoya, Japan, 1990.

[22] A. C. Kakas and P. Mancarella. Knowledge assimilation and abduction. *ECAI-90, International Workshop on Truth Maintenance*, Stockholm, Springer Verlag Lecture notes in Computer Science, 1990.

[23] A. C. Kakas, P. Mancarella and G. Plagenza. Constructive Abduction in Logic Programming. Technical Report, University of Cyprus, 1995.

[24] A. C. Kakas and A. Michael. Integrating abductive and constraint logic programming. *Proc. Twelfth International Conference Logic Programming* , 399–413, Tokyo, Japan, 1995.

[25] A. C. Kakas, R. A. Kowlaski and F. Toni. Abductive Logic Programming. *Journal of Logic and Computation*, 2(6), 719-770, 1993.

[26] R. A. Kowalski. *Logic for problem solving.* Elsevier, New York, 1979.

[27] R.A. Kowalski and M. Sergot. A logic-based calculus of events. *New Generation Computing* 4, 67–95, 1986.

[28] R.A. Kowalski. Database updates in the event calculus. *Journal of Logic Programming*, 12, 121–146, 1992.

[29] N. Lavrac and Saso Dzeroski. *Inductive Logic Programming: Techniques and Applications*. Ellis Horwood, 1994.

[30] D. Leake. Abduction, Experience and Goals: A Model for Everyday Abductive Explanation. *Journal of Experimental and Theoretical Artificial Intelligence*, to appear.

[31] R. S. Michalski. Inferential Theory of Learning as a Conceptual Basis for Multistrategy Learning. *Machine Learning*, Vol. 11, pp 111-151, 1993.

[32] R. Michalski and G. Tecuci. *Machine Learning: A Multistrategy Approach*, Vol. IV, Morgan Kaufmann, 1994.

[33] T. Mitchell and S. Kedar-Cabelli. Explanation-based Learning. A Unifying View. *Machine Learning*, Vol. 1, pp 47-80, 1986.

[34] R. J. Mooney. Induction over the Unexplained: Using Oberly-General Domain Theories to Aid Learning. *Machine Learning*, Vol. 10, pp. 79-110, 1993.

[35] R. J. Mooney and J. M. Zelle. Integrating ILP and EBL. *ACM SIGART*, Vol. 5, No. 1, pp. 12-21, 1994.

[36] K. Morik, S. Wrobel, J. Kietz and W. Emde. *Knowledge Acquisition and Machine Learning: Theory, Methods and Applications*, Academic Press, 1993.

[37] S. Muggleton and L. De Raedt. Inductive logic programming : theory and methods. *Journal of Logic Programming*, Vol. 19-20, pp 629-679, 1994.

[38] S. Muggleton. Inverse entailment and Progol. *New Generation Computing*, 13, 1995.

[39] D. Ourston and R. J. Mooney. Theory Refinement Combining Analytical and Empirical Methods. *Artificial Intelligence*, 66, pp. 311–344, 1994.

[40] G. Paul. Approaches to abductive reasoning: an overview. *Artificial Intelligence Review*, 7, 109-152, 1993.

[41] M. Pazzani and D. Kibler. The Utility of Knowledge in Inductive Learning. *Machine Learning*, 9, pp. 57-94, 1992.

[42] D. Poole. A logical framework for default reasoning. *Journal of Artificial Intelligence*, 36, 27–47, 1988.

[43] C. S. Peirce. *Collected Papers of Charles Sanders Peirce*, Vol. 2, 1931-1958. Hartshorn at al. (eds), Harvard University Press.

[44] Reggia, J., Diagnostic experts systems based on a set-covering model. *International Journal of Man-Machine Studies*, 19(5), 437–460, 1983.

[45] R. Reiter. A Logic for default reasoning. *Journal of Artificial Intelligence*, 13, 81–132, 1980.

[46] B. L. Richards and R. J. Mooney. Automated refinement of forst-order horn-clause domain theories. *Machine Learning*, 19, 95-131, 1995.

[47] P. O'Rorke. Abduction and Explanation-Based Learning: Case Studies in Diverse Domains. *Computational Intelligence*, Vol. 10, pp. 295-330, 1994.

[48] L. Saitta, M. Botta and F. Neri. Multistrategy Learning and Theory Revision. *Machine Learning*, Vol. 11, pp. 153-172, 1993.

[49] A. Sattar and R. Goebel. Using crucial literals to select better theories. Tech. Report Dept. of Computer Science, University of Alberta, Canada, 1989.

[50] G. Tecuci. An Inference-based Framework for Multistrategy Learning. In *Machine Learning: A multistrategy approach*, R. Michalski and G. Tecuci, 1994, Morgan Kaufmann.

[51] C. Thompson and R. Mooney. Inductive Learning for Abductive Diagnosis. *Proc. of AAAI-94*.

[52] J. Wogulis. Revising Relational Domain Theories. *Proc. of the 8th Inter. Workshop on Machine Learning*, Evaston, IL, pp. 462-466, Morgan Kaufmann, 1991.

[53] S. Wrobel. First Order Theory Refinement. *ILP Project (ESPRIT III) Common Deliverable* for Workpackage 2, 1995.