Tecuci G., Plausible Justification Trees: a Framework for the Deep and Dynamic Integration of Learning Strategies, *Machine Learning Journal*, vol.11, pp. 237-261, 1993.

Plausible Justification Trees: A Framework for Deep and Dynamic Integration of Learning Strategies

GHEORGHE TECUCI
Center for Artificial Intelligence, Department of Computer Science,
George Mason University, 4400 University Drive, Fairfax, VA 22030-4444, USA
and Research Institute for Informatics, 71316, Bd. Averescu 8-10, Bucharest 1, Romania
email: tecuci@aic.gmu.edu
tel: (703) 993-1722

Abstract. This paper describes a framework for the deep and dynamic integration of learning strategies. The framework is based on the idea that each single-strategy learning method is ultimately the result of certain elementary inferences (like deduction, analogy, abduction, generalization, specialization, abstraction, concretion, etc.). Consequently, instead of integrating learning strategies at a macro level, we propose to integrate the different inference types that generate individual learning strategies. The paper presents a concept learning and theory revision method that was developed in this framework. It allows the system to learn from one or from several (positive and/or negative) examples, and to both generalize and specialize its knowledge base. The method integrates deeply and dynamically different learning strategies, depending of the relationship between the input information and the knowledge base. It also behaves as a single-strategy learning method whenever the applicability conditions of such a method are satisfied.

Key words: multistrategy task-adaptive learning, plausible justification trees, theory revision, concept learning

running head: A Framework for Multistrategy Learning

1. Introduction

Research in machine learning has elaborated and investigated in detail several single-strategy learning methods like, for instance, empirical induction, explanation-based learning, learning by abduction, learning by analogy, case-based learning, and others [Michalski, Carbonell & Mitchell, 1983, 1986; Kodratoff & Michalski, 1990; Shavlik & Dietterich, 1990]. However, as this field evolves and concentrates more and more on solving complex real-world learning problems, it becomes more and more clear that the single-strategy learning methods provide solutions to overly-simplified problems. One kind of over-simplification consists of specific requirements imposed to the input information and to the content of the KB. For instance, empirical induction requires many input examples and a small amount of background knowledge. Explanation-based learning requires one input example and a complete background knowledge. Learning by analogy and case-based learning require background knowledge analogous with the input. Learning by abduction requires causal background knowledge related to the input. Another kind of over-simplification consists of the limited result of the single-strategy learning process. This is a hypothetical generalization of several input examples (in the case of empirical induction), or an operational generalization of an input example (in the case of explanation-based learning), or new knowledge about the input (in the case of learning by analogy or case-based learning), or new background knowledge (in the case of learning by abduction).

From the above characterization, one may notice however the complementarity of the requirements and of the results of the single-strategy learning methods. This naturally suggests that by properly integrating them, one could obtain a synergistic effect in which different strategies mutually support each other, and compensate for each other's weaknesses. This hypothesis has been confirmed by the many multistrategy learning methods and systems that have been developed in the past several years [e.g., Bergadano & Giordana, 1990; Cox & Ram, 1991; Danyluk, 1987; DeRaedt & Bruynooghe, 1991; Flann & Dietterich, 1989; Genest, Matwin & Plante, 1990; Hirsh, 1989; Lebowitz, 1986; Minton & Carbonell, 1987; Mooney & Ourston, 1991; Morik, 1992; Pazzani, 1988; Reich, 1991; Saitta & Botta, 1992; Shavlik & Towell, 1990; Tecuci & Kodratoff, 1990; Whitehall, 1990; Widmer, 1991; Wilkins, 1990].

After the development of many methods and techniques for the integration of learning strategies, the research in multistrategy learning started to address the problem of defining general principles and frameworks for the design of advanced multistrategy learning systems [Michalski, 1992; Michalski & Tecuci, 1991]. One such framework for a multistrategy learning system consists of a cascade of single strategy learning modules, in which the output of one module is an input to the next module. Another framework consists of a global control module and a tool box of single strategy learning modules, all using the same knowledge base. The control module analyzes the relationship between the input and the knowledge base and decides which learning module to activate.

In this paper we propose another general framework for multistrategy learning. This framework is based on the idea that each single-strategy learning method is ultimately the result of certain elementary inferences (like deduction, analogy, abduction, generalization, specialization, abstraction, concretion, etc.). As a consequence, instead of integrating learning strategies at a macro level, we propose to integrate the different inference types that generate individual learning strategies. By this we achieve a deep integration of the learning strategies. The paper presents a concept learning and theory revision method that was developed in this framework. It allows the system to learn from one or from several (positive and/or negative) examples, and to both generalize and specialize its knowledge base. The method integrates deeply and dynamically different learning strategies, depending of the relationship between the input information and the knowledge base. It is therefore a multistrategy task-adaptive learning (MTL) method [Michalski, 1990, 1992; Tecuci & Michalski, 1991a,b]. An important feature of this MTL method is that it is also a generalization of the integrated single-strategy methods in that it behaves as any of these methods whenever their applicability conditions are satisfied.

This paper is organized as follows. Section 2 defines and illustrates the general learning task of the MTL method. The next section contains a general presentation of the proposed MTL method. Sections 4, 5, and 6 present in more detail and illustrate the main stages of the MTL method. Next, section 7 presents the cases in which the MTL method behaves as a single-strategy learning method. The last section of the paper analyzes the strength and the limitations of our approach to multistrategy learning, and indicates what we consider to be the most promising directions of the future research.

2. The Learning Task

The learning task of a system is defined by the input information, the background knowledge, and the learning goal. We are considering a general learning task for multistrategy learning, that subsumes the learning tasks of the integrated single-strategy methods. In particular, it is both a theory revision task and a concept learning task, as indicated in Table 1.

Table 1. The learning task.

Input: *one or several (positive and/or negative) examples of a concept.*
The examples are represented as conjunctions of first-order predicates, are considered noise-free, and are presented in sequence.

Background knowledge: *incomplete and partially incorrect knowledge base (KB).*
The KB may include a variety of knowledge types (facts, examples, implicative or causal relationships, determinations, etc.), represented with first-order predicates.

Goal: *learn different concept definitions from the input example(s) and improve the KB.*
The learned concept definitions may be operational or abstract, and the KB is improved by both generalizing and specializing it, so that to entail these definitions.

By generalization of the KB we mean any transformation that results in an increase of knowledge inferable from the KB. The KB may be generalized by generalizing knowledge pieces or by simply adding new knowledge pieces.
Similarly, by specialization of the KB we mean any transformation that results in a decrease of knowledge inferable from the KB. The KB may be specialized by specializing knowledge pieces or by simply removing knowledge pieces from the KB.
These operations are also associated with an increase in the plausibility of the knowledge pieces inferable from the KB.

As stated in the above formulation of the learning task, our approach is based on the following assumptions.
The input to the learning system consists of concept examples that are noise-free. However, the system may learn from a single positive example, or from a sequence of positive and negative examples.
The KB is considered to be both incomplete and partially incorrect. It may also contain different types of knowledge pieces expressed as first-order predicate formulas.
The goal of the learning system is to learn as much as possible from any input it receives. This is a general goal that consists in learning different types of concept definitions, and in performing different types of improvements of the KB. In a specific application of this learning method, this goal would need to be specialized. For instance, some of the learnable

concept definitions may not be useful and, consequently, will not be learned.

In order to illustrate this learning task and the corresponding learning method, we shall consider the case of a learning system in the area of geography. The purpose of the system is that of acquiring geographical data and rules in order to answer questions about geography. Throughout this paper we use : : > to denote concept assignment, □to denote certain (deductive) implication, ∅ to denote plausible implication, and --> to denote plausible determination (see section 4.3).

Let us consider, for instance, that the knowledge base is the one from Table 2. It contains several ground facts, two examples of fertile soil, a plausible determination rule and three deductive rules.

Table 2. A sample of an incomplete and partially incorrect KB.

Facts:

terrain(Philippine, flat), rainfall(Philippine, heavy), water-in-soil(Philippine, high)

Examples *(of fertile soil):*

soil(Greece, red-soil) : : > soil(Greece, fertile-soil)

terrain(Egypt, flat) & soil(Egypt, red-soil) : : > soil(Egypt, fertile-soil)

Plausible determination:

rainfall(x, y) --> water-in-soil(x, z)

Deductive rules:

∀ x, soil(x, loamy) ==> soil(x, fertile-soil)

∀ x, climate(x, subtropical) ==> temperature(x, warm)

∀ x, water-in-soil(x, high) & temperature(x, warm) & soil(x, fertile-soil) ==> grows(x, rice)

Let us also consider that the input consists of the sequence of examples from Table 3. The left hand side of each positive example (negative example) is the description of a country that grows rice (does not grow rice), and the right hand side is the statement that the respective country grows rice (does not grow rice).

Table 3. Positive and negative examples of "grows(x, rice)".

Positive Example 1:
rainfall(Vietnam, heavy) & climate(Vietnam, subtropical) & soil(Vietnam, red-soil) & terrain(Vietnam, flat) & location(Vietnam, SE-Asia) : : > grows(Vietnam, rice)

Positive Example 2:
rainfall(Madagascar, heavy) & climate(Madagascar, subtropical) & soil(Madagascar, loamy) & terrain(Madagascar, flat) & in(Madagascar, Pacific-Ocean) : : > grows(Madagascar, rice)

Negative Example 3:
rainfall(Nepal, heavy) & climate(Nepal, subtropical) & soil(Nepal, loamy) & terrain(Nepal, abrupt) & location(Nepal, Central-Asia) : : > ¬ grows(Nepal,rice)

The different types of knowledge pieces learned from the above KB and input examples are presented in Table 4.
One result of learning consists of several concept definitions [Michalski, 1990].
The first definition in Table 4 is an operational definition of "grows(x, rice)", expressed with the features present in the input examples.
The second definition is an abstract definition of "grows(x, rice)", expressed with more general features, derived from those present in the input examples (since this rule was already known, the new knowledge is just that it represents an abstract definition).
The third definition is an abstraction of Example 1 that was obtained by instantiating the previous abstract definition.

The other result of learning is the improvement of the KB so as to entail the learned concept definitions.
The KB was generalized by learning two new facts and a rule.
It was also specialized, by conjunctively adding a literal to the left hand side of the plausible determination.
As indicated in Table 4, the system also keeps all the examples of the learned knowledge pieces in order to update them when new knowledge becomes available. These instances have been generated through different forms of plausible reasoning and have been validated during the learning process. Therefore, they also constitute an improvement of the KB.

Table 4. The learned knowledge.

Concept definitions

Operational definition of "grows(x, rice)":

{ rainfall(x, heavy) & terrain(x, flat) & climate(x, subtropical) &
(soil(x, red-soil) ∆ soil(x, loamy)) } : : > grows(x, rice)

Abstract definition of "grows(x, rice)":

water-in-soil(x, high) & temperature(x, warm) & soil(x, fertile-soil) : : > grows(x, rice)

Abstraction of Example 1:

water-in-soil(Vietnam, high) & temperature(Vietnam, warm) &
soil(Vietnam, fertile-soil) : : > grows(Vietnam, rice)

Improved KB

New facts:

water-in-soil(Vietnam, high), water-in-soil(Madagascar, high)

New rule:

∀ x, soil(x, red-soil) ––> soil(x, fertile-soil)
with the positive examples: (x<-Greece), (x<-Egypt), (x<-Vietnam).

Improved (specialized) plausible determination:

rainfall(x, y) & terrain(x, flat) --> water-in-soil(x, z)
with the positive examples: (x<-Philippine, y<-heavy, z<-high),
(x<-Vietnam, y<-heavy, z<-high),
(x<-Madagascar, y<-heavy, z<-high).
with the negative example: (x<-Nepal, y<-heavy).

3. General Presentation of the Learning Method

The learning method consists in building, generalizing and/or specializing plausible justification trees of the input examples, and in generalizing and/or specializing the KB so as to entail these trees.

A plausible justification tree is a demonstration that the input is a plausible consequence of the KB. It is like a proof tree, except that the inference steps which compose it may be the result of different types of reasoning (not only deduction, but also analogy, inductive prediction, abduction, etc.). For instance, a plausible justification tree of Example 1 in Table 3 is the one from Figure 1. It shows that, in the context of the current KB, Example 1 is indeed a positive example of "grows(x, rice)".

In the following, we shall associate with each elementary inference step from a plausible justification tree (for instance, from A infer B), an implication (A ∅ B or A □B, depending of whether the inference is plausible or certain).

The main steps of the learning method are the following ones (more details are given in the next sections).

• ***For the first positive example I_1:***

1. Build a plausible justification tree T of I_1

The plausible justification tree T demonstrates that the input I_1 is a plausible consequence of the knowledge from the KB. The inference steps in the tree T may be the result of different types of inference: deduction, analogy, inductive prediction, abduction, etc.

2. Build the plausible generalization T_u of T

First build an explanation structure ES, by replacing each implication from T with a plausible generalization of it. The generalization of each implication will depend of the type of inference, and of the knowledge used to derive it. It will correspond to the least general generalization of all the similar implications that the system would consider plausible.

Then determine the most general unification of the general implications from ES. The obtained tree T_u is the most general plausible generalization of T.

3. Generalize the KB so that to entail T_u

Introduce into the KB the knowledge pieces hypothesized (through analogy, inductive generalization and prediction, abduction, etc.) during the building of the plausible justification tree T and the explanation structure ES.

• ***For each new positive example I_i:***

1. Generalize T_u so as to cover a plausible justification tree of I_i

Determine the instance T_i of T_u (which, in general, is an AND/OR tree) corresponding to the current input I_i. Analyze the leaf predicates and the implications from T_i and remove the false ones. If the resulting T_i is a plausible AND/OR tree (i.e. it contains a plausible justification of I_i) then T_u already covers a plausible justification tree of I_i. This ends the processing of the current example. Otherwise make minimum modifications to T_i, so as to contain a plausible justification of the current positive example, and generalize T_u, as little as possible, so as to cover the updated justification tree T_i.

2. Generalize the KB so that to entail the new T_u

Introduce into the KB the knowledge pieces hypothesized (through analogy, inductive generalization and prediction, abduction, etc.) during the updating of the plausible justification trees T_i and T_u.

• ***For each new negative example I_i:***

1. Specialize T_u so as not to cover any plausible justification tree of I_i

Determine the instance T_i of T_u corresponding to the current input I_i. Analyze the leaf predicates and the implications from T_i and remove the false ones. If the resulting T_i is not a plausible AND/OR tree (i.e., it does not contain any plausible justification of I_i), then T_u does not covers any plausible justification tree of I_i. This ends the processing of the current example. Otherwise, T_i is a wrong AND/OR justification tree that is covered by T_u. Indeed, T_i "proves" that the current input is a positive example, although it is known that it is a negative example. Hypothesize that the implications $R_{k1},...,R_{kn}$ from T_i are false. The selection of the implications $R_{k1},...,R_{kn}$ is based on the following criteria:

- if all the implications $R_{k1},...,R_{kn}$ are false, then T_i is not a plausible AND/OR tree (i.e., it does not contain any plausible justification of I_i);
- prefer the weakest implications (first abduction, then inductive prediction, then analogy, and then deduction);
- prefer the implications for which the specializations of the KB and of T_u cause the minimum loss of coverage of previous examples;
- prefer the implications for which the specializations of the KB and of T_u produce a minimum increase in the complexity of the modified knowledge pieces;

Specialize T_u, as little as possible, so as no longer to cover the implications $R_{k1},...,R_{kn}$.

2. Specialize the KB so that to entail the new T_u without entailing the previous T_u

Specialize, as little as possible, the knowledge pieces from the KB that generated the implications $R_{k1},...,R_{kn}$, so as no longer to entail them.

• ***Learn different concept definitions***

Extract several knowledge pieces from the tree T_u as, for instance, operational definitions of the concept illustrated by the input examples, abstract definitions of the concept illustrated by the input examples, or abstractions of different input examples.

4. Learning from the First Example

4.1 Building the plausible justification tree

In order to determine a plausible justification of the input, the system builds an AND/OR tree by conducting a top-down uniform-cost search [Nilsson, 1971].
The developed AND/OR tree contains several AND trees, each having a cost that estimates its global plausibility. The cost of a partial AND tree is computed as a tuple (m, n), where m represents the number of the deductive inference steps in the tree, and n represents the number of the non-deductive inference steps (which, in the present version of the MTL system, could be made by analogy, inductive prediction or abduction). The ordering relationship for the cost function is defined as follows:

(m1, n1) < (m2, n2) if and only if n1<n2 or (n1=n2 and m1<m2)

This cost function guarantees that the system will find the justification tree with the fewest number of non-deductive inference steps. In particular, it will find a deductive tree (if one exists) and the deductive tree with the fewest inference steps (if several exist).

As a general strategy, the system always tries to justify a given predicate (for instance, "grows(Vietnam, rice)" in Figure 1) by deduction. If it succeeds, then it tries to justify the resulting predicates ("water-in-soil(Vietnam, high)", "temperature(Vietnam, warm)" and "soil(Vietnam, fertile-soil)"). However, if it fails, then it tries to justify the predicate by using as many plausible reasoning methods as possible. It will try these methods in order: first analogy, then inductive prediction and lastly abduction. If one of them produces a plausible inference step, then the system tries the remaining ones in order to confirm or to contradict it. If no contradiction is found, the inference step is accepted. This method (although quite simple and definitely a necessary topic of future research) is related to that employed by humans [Collins & Michalski, 1989]. Indeed, Collins and Michalski argue that people solve problems by pursuing different "lines of reasoning". They estimate the "strength" of each line of reasoning, and make their conclusion on the basis of this evaluation. If the lines lead to the same conclusion, they have a strong belief in the result. If the lines lead to different conclusions, and the associated "strengths" are roughly similar, people restrain from making any decisive conclusion.

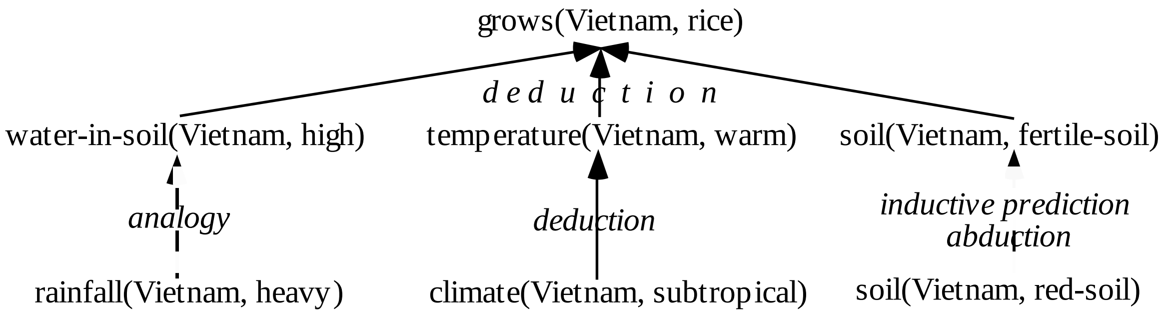

Figure 1. A plausible justification of Example 1.

It should be noticed that, although at the level of a given inference step, the current MTL system applies the different reasoning methods in a predefined order, globally (at the level of the resultant justification tree) there is no predefined order. For instance, in the case of the justification tree in Figure 1, the order of the inference steps was: deduction, analogy, deduction, inductive prediction and abduction. In general, this order depends of the relationship between the KB and the input. Therefore, the MTL method is an example of a dynamic and deep (i.e. at the level of individual inference steps) integration of single strategy learning methods (each learning method corresponding to a specific type of inference).

The next sections present briefly the way the different inference steps in Figure 1 have been made.

4.2 Deduction

Two inference steps in Figure 1 are the results of deductions based on the deductive rules in Table 2, as shown in Table 5.

Table 5. The justifications of the deductive steps in Figure 1.

Deduction 1:

∀ x, water-in-soil(x, high) & temperature(x, warm) & soil(x, fertile-soil) ==> grows(x, rice)

<u>water-in-soil(Vietnam, high) & temperature(Vietnam, warm) & soil(Vietnam, fertile-soil)</u>

grows(Vietnam, rice)

Deduction 2:

∀ x, climate(x, subtropical) ==> temperature(x, warm)

<u>climate(Vietnam, subtropical)</u>

temperature(Vietnam, warm)

4.3 Analogy

Analogical inference is the process of transferring knowledge from a known entity S to a similar but less known entity T. S is called the *source* since it is the entity that serves as a source of knowledge, and T is called the *target* since it is the entity that receives the knowledge. The central intuition supporting this type of inference is that if two entities, S and T, are similar in some respects, then they could be similar in other respects as well. Therefore, if S has some feature, then one may infer by analogy that T has a similar feature.

In the present version of MTL we use a simple form of analogy based on plausible determinations defined as follows:

P(x, y) --> Q(x, z) (P plausible determines Q) meaning
∀S, ∀T { If ∃y [P(S, y) & P(T, y)] then it is probably true that ∃z [Q(S, z) & Q(T, z)] }
where P and Q are first order logical expressions.

Otherwise stated, if the source S and the target T are characterized by the same feature P (i.e., P(S, y_0)=true and P(T, y_0)=true) then it is probably true that they are also characterized by a same feature Q. Therefore, if Q(S, z_0)=true then one may infer by analogy that Q(T, z_0)=true.

We use the term "probably true" to express that the determination-based analogy we are considering is a weak method that does not guarantee the truth of the inferred knowledge. This is similar to the mutual dependency rules introduced by [Collins & Michalski, 1989; Michalski, 1992], but different from the determination rules introduced by [Davies & Russell, 1977] which guarantee the truth of the inferred knowledge.

The analogical inference step in Figure 1 was made by using the plausible determination
    rainfall(x, y) --> water-in-soil(x, z)
    (i.e. the rainfall of an area determines the quantity of water in the soil of that area)
as illustrated in Figure 2.
As can be seen in Figure 2, Philippine and Vietnam are similar from the point of view of "rainfall" (in both cases this is heavy). Therefore, one may infer by analogy that the two countries are also similar from the point of view of "water-in-soil". Thus, the system concluded that "water-in-soil(Vietnam, high)" from the fact "water-in-soil(Philippine, high)".

Analogy:

$$\begin{tikzcd}[column sep=huge]
\text{rainfall(Philippine, heavy)} \arrow[r, leftrightarrow, "\textit{similar}"] \arrow[d, "\textit{determines}"'] & \text{rainfall(Vietnam, heavy)} \arrow[d, "\textit{determines ?}"] \\
\text{water-in-soil(Philippine, high)} \arrow[r, leftrightarrow, "\textit{similar}"] & \text{water-in-soil(Vietnam, high)}
\end{tikzcd}$$

Figure 2. Hypothesizing Vietnam's water-in-soil through analogy.

One should notice that a plausible determination rule indicates only what kind of knowledge could be transferred from a source to a target (knowledge about "water-in-soil", in the case of the considered determination), and in what conditions (the same type of "rainfall"). It does not indicate, however, the exact relationship between the type of the rainfall (for instance, "heavy") and the quantity of water in soil ("high"). The exact relationship is indicated by the source entity ("Philippine"). Therefore, a plausible determination rule alone (without such a source entity), cannot be used in the inference process.

The determination rules have been previously used in an explanation-based learning framework by [Mahadevan, 1989] and [Widmer, 1989]. In MTL, however, they are only one way of implementing analogical reasoning. In general, the MTL method is intended to incorporate different forms of analogy, based on different kinds of similarities, such as similarities among causes, relations, and meta-relations [Carbonell,1986; Gentner,1983; Kedar-Cabelli, 1985; Kodratoff, 1990; Michalski, 1992; Porter, Bareiss & Holte,1990; Winston, 1986].

4.4 Inductive prediction

Inductive prediction consists in finding an inductive generalization of a set of examples of a concept and in applying it in order to predict if a new instance is (or is not) a positive example of the concept.

The generalization of the examples could be obtained through a process of empirical or constructive generalization. The generalization process is empirical if it involves only descriptors from the description space of the examples, and is constructive if it introduces new descriptors which do not belong to the description space of the examples. A detailed characterization of empirical and constructive generalization is given in [Michalski, 1992].

In the current version of MTL, we use an inductive generalization method that determines the most specific generalization of a set of positive examples that does not cover any of the negative examples. In general, the result will be a disjunction of conjunctive expressions. Moreover, the system is keeping all the examples in order to update the generalization when new examples become available.

One inference step in Figure 1 was the result of inductive prediction. Indeed, in order to prove that "soil(Vietnam, fertile-soil)" is true, the system looked into the KB for examples of "fertile-soil". Then it inductively generalized them to a rule that was used to predict the inference step from Figure 1, as indicated in Table 6.

It is important to stress that the system keeps the learned rule in the KB as an inductive generalization. Therefore, future applications of this rule are also inductive predictions.

Let us also notice that the rule in Table 6 was obtained through an empirical generalization process because it is expressed only in terms of the descriptors used in the examples.

Table 6. Making an inference through inductive prediction.

Examples from KB:

soil(Greece, red-soil) : : > soil(Greece, fertile-soil)

terrain-type(Egypt, flat) & soil(Egypt, red-soil) : : > soil(Egypt, fertile-soil)

Inductive generalization:

∀ x, soil(x, red-soil) ––> soil(x, fertile-soil)

with the positive examples: (x<-Greece)(x<-Egypt)

Predicted inference:

soil(Vietnam, red-soil) ––> soil(Vietnam, fertile-soil)

4.5 Abduction

In general, abduction is defined as follows [Josephson, 1991]:

D is a collection of data;

H explains D;

No other hypothesis is able to explain D as well as H does;

Therefore, H is probably true.

In general, abduction involves two steps, generation of explanatory hypotheses and selection of the "best" hypothesis.

In the current version of MTL we consider two forms of abduction:

a) tracing backward a deductive rule
If D is to be explained and H ==> D then hypothesize H.
In particular, *if* $H=H_1 \,\&\, H_2 \,\&\, \ldots \,\&\, H_n$ *and* $H_2 \,\&\, \ldots \,\&\, H_n$ *is true then hypothesize* H_1.

b) hypothesizing an ISA relationship (i.e. d_1 ISA d_2)
If $P(a, d_2)$ *is to be explained and* $P(a, d_1)$ *is true then hypothesize that* $P(a, d_1) \longrightarrow P(a, d_2)$.

Choosing the "best" abductive hypothesis is the most difficult problem of abductive learning. This is somewhat simplified in the context of MTL because the system is trying to make an inference step through as many plausible inference methods as possible and abduction is the last one to try (as shown in section 4.1). Therefore, if an inference "H —> D" has been made through some other form of reasoning, abduction is used only to confirm this inference or to contradict it (i.e. to prove "H —> C", where D & C = false).
In the absence of the above criterion, the system chooses the abductive hypotheses in the following order:
- prefer the ISA abductions;
- prefer to backtrace the rule $H_1 \,\&\, H_2 \,\&\, \ldots \,\&\, H_n \Longrightarrow D$ with the highest number of true antecedents;
- prefer to backtrace the rule that has the highest number of known instances;
- prefer the simplest hypothesis.

In the case of the plausible justification tree in Figure 1, the system made an ISA abduction confirming the previously made inductive prediction from Table 6. Indeed, "soil(Vietnam, fertile-soil)" needed to be proven and "soil(Vietnam, red-soil)" was known to be true. Therefore, the system abduced the ISA relationship from Table 7.

Table 7. Abducing an ISA relationship.
Abduction: soil(Vietnam, red-soil) —> soil(Vietnam, fertile-soil)

4.6 Generalization of the plausible justification tree

Once a justification tree was successfully created, the system analyzes the individual implications associated with the elementary inference steps to determine if these implications could be locally generalized within the constraints of the KB that were used to make the inference steps. After the implications are generalized locally, the system unifies them globally, and builds a generalized justification tree. This technique is an extension of the one elaborated by [Mooney & Bennet, 1986]. The extension concerns the way individual implications are generalized, by using the knowledge from which they were derived. The idea is to replace each implication A —> B (or A ==> B) with the least general generalization of all the similar implications that could be obtained from the knowledge that produced it [Tecuci & Michalski, 1991b].

A deductive implication is replaced by the deductive rule that generated it. This is a deductive generalization.

An analogical implication is generalized by considering the knowledge used to derive it.
In our example, the implication

rainfall(Vietnam, heavy) —> water-in-soil(Vietnam, high)

was obtained by analogy with "rainfall(Philippine, heavy)" and "water-in-soil(Philippine, high)", based on the determination

rainfall(x, y) --> water-in-soil(x, z).

Because the system would infer "water-in-soil(x, high)" for any x such that "rainfall(x, heavy)", the analogical implication is generalized to:

$\forall$ x, rainfall(x, heavy) —> water-in-soil(x, high).

This is a generalization based on analogy.

An implication obtained through inductive prediction is generalized to the rule that produced it. Therefore, the implication from Table 6 obtained through inductive prediction would be replaced with the inductive generalization from Table 6.

An abductive implication obtained by tracing backward a deductive rule would be generalized to that rule. However, for an abduced ISA relationship there is no knowledge that could be used to generalize it. Therefore it would remain unchanged in the explanation structure.

An implication obtained through several forms of reasoning is generalized to the least general expression that covers the generalizations corresponding to individual reasoning methods. Therefore, the implication from Figure 1, that was obtained both through inductive prediction and abduction, is generalized to

$\forall$ x, soil(x, red-soil) ––> soil(x, fertile-soil)

which is the least general generalization of the rule in Table 6 and the abduced ISA relationship in Table 7.

The generalization of the implications from Figure 1 form the explanation structure shown in Figure 3.

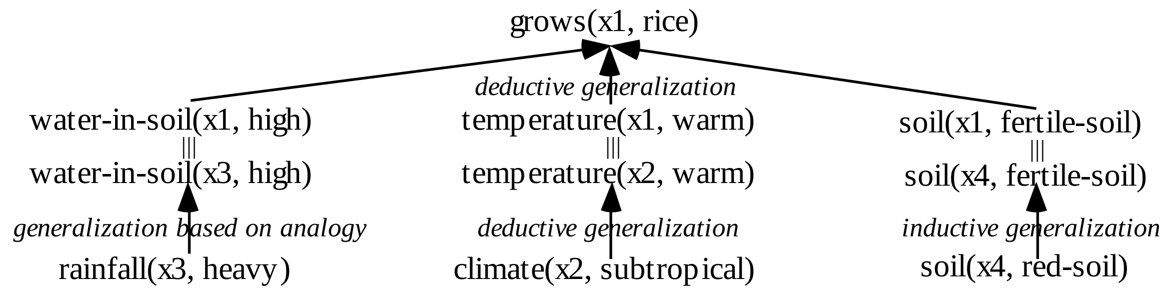

Figure 3. Explanation structure.

The most general unification of the connection patterns in Figure 3 is (x1=x2=x3=x4=x). By making these unifications one obtains the tree in Figure 4 which represents the most general plausible generalization of the justification tree from Figure 1.

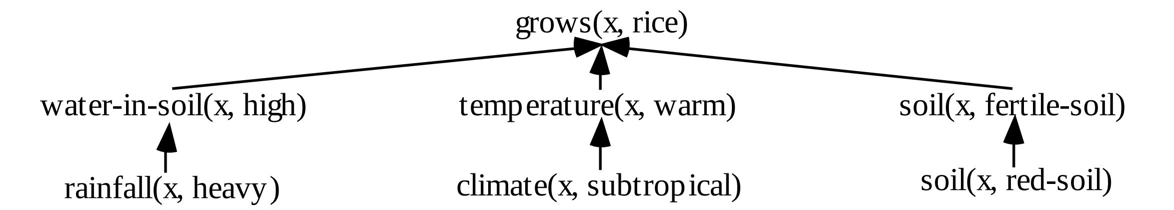

Figure 4. A generalization of the plausible justification tree from Figure 1.

An interesting research direction suggested by the generalization of the plausible justification tree is to investigate different forms of generalizations, not only deductive and inductive, but possibly also analogical, abductive, etc.

4.7 Generalization of the KB

As indicated in Table 2, the system may improve the KB by learning different types of knowledge.

In this case it generalized the KB by learning a new fact (by analogy)
water-in-soil(Vietnam, high),
positive examples of the determination

rainfall(x, y) --> water-in-soil(x, z)
(x<-Philippine, y<-heavy, z<-high),
(x<-Vietnam, y<-heavy, z<-high).
and a rule (by empirical generalization)
∀ x, soil(x, red-soil) –-> soil(x, fertile-soil)
with the positive examples: (x<-Greece), (x<-Egypt), (x<-Vietnam).

If Example 1 is the only input, then the system also extracts several concept definitions from the trees in Figures 1 and 4. For instance, it stores the leaves of the general tree in Figure 4 as an *operational definition* of the concept "grows(x, rice)":
rainfall(x, heavy) & climate(x, subtropical) & soil(x, red-soil) : : > grows(x, rice)
It also stores the upper part of the tree as the *abstract definition* of "grows(x, rice)":
water-in-soil(x, high) & temperature(x, warm) & soil(x, fertile-soil) : : > grows(x, rice)
Finally, it stores the upper part of the tree in Figure 1 as an *abstraction* of Example 1:
water-in-soil(Vietnam, high) & temperature(Vietnam, warm) &
soil(Vietnam, fertile-soil) : : > grows(Vietnam, rice)

5 Learning from a New Positive Example

5.1 Generalization of the plausible justification tree

Let us now consider that the system receives Example 2 in Table 3. As indicated in Section 3, the system tries to generalize the current justification tree T_u in Figure 4 so that to cover a justification of the new positive example. In the same time, it may also generalize the KB, if this is needed in order to entail the new T_u.

First of all, the system determines the instance of the general tree in Figure 4, corresponding to Example 2 in Table 3 (see Figure 5).

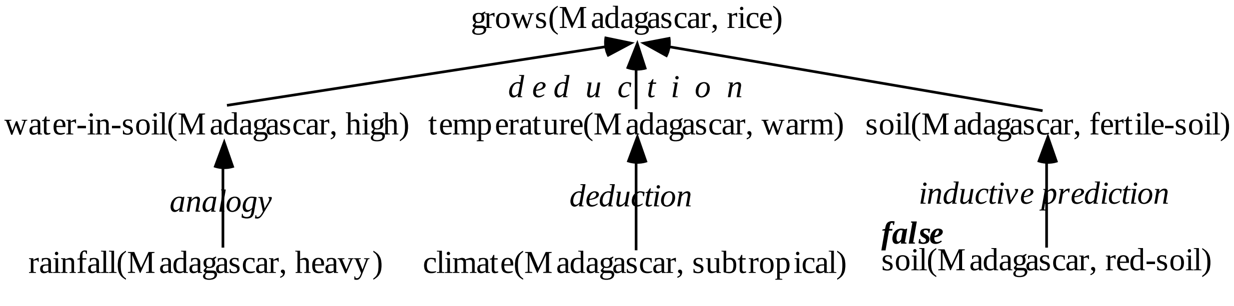

Figure 5. Instance of the general justification tree corresponding to Example 2.

Then the system analyzes the leaf predicates and the inference steps from this proof tree. If the leaf predicates are true and the inference steps are plausible, then the tree in Figure 5 is a plausible justification of the new positive example that is already covered by the general justification tree in Figure 4. This ends the processing of the current example.

However, the tree in Figure 5 is not a correct justification of Example 2 because the leaf predicate "soil(Madagascar, red-soil)" is not true. Therefore, the system uses the deductive rule "∀x, soil(x, loamy) ==> soil(x, fertile-soil)", from Table 2, and builds the plausible justification tree in Figure 6.

It is important to notice that the plausible justification tree of Example 2 has been built by using the plausible justification tree of the previous example. This not only facilitates the process of building the justification tree, but also the process of generalizing the general tree T_u, as will be shown in the following. Moreover, it shows some similarities between our method and human learning which involves the use of the explanations of previous examples in the process of building an explanation for a new example [Wisniewski & Medin, 1991].

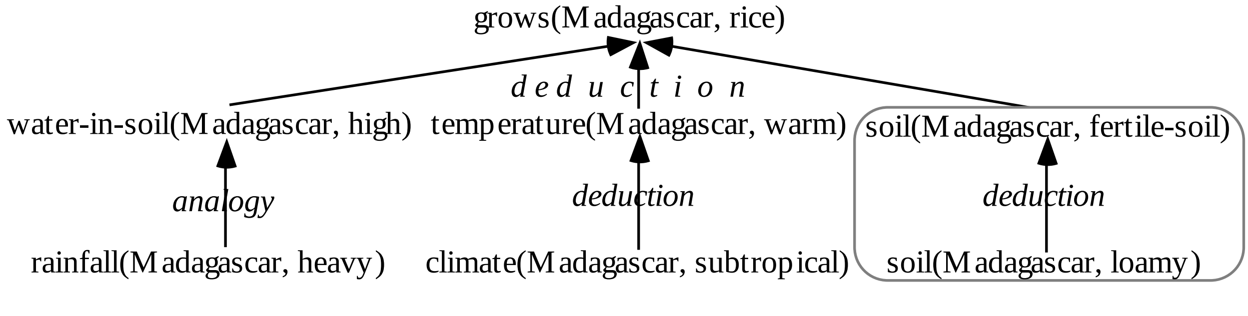

Figure 6. Plausible justification of Example 2 in Table 3.

The next step of the learning process is to build the explanation structure in Figure 7 that has three general components to be unified:
- the part of the tree in Figure 4 that covers part of the tree in Figure 6,
- the part that is specific to the tree in Figure 4, and
- the generalization of the part of the tree in Figure 6 that is specific to it (this generalization being made according to the procedures described in section 4.6).

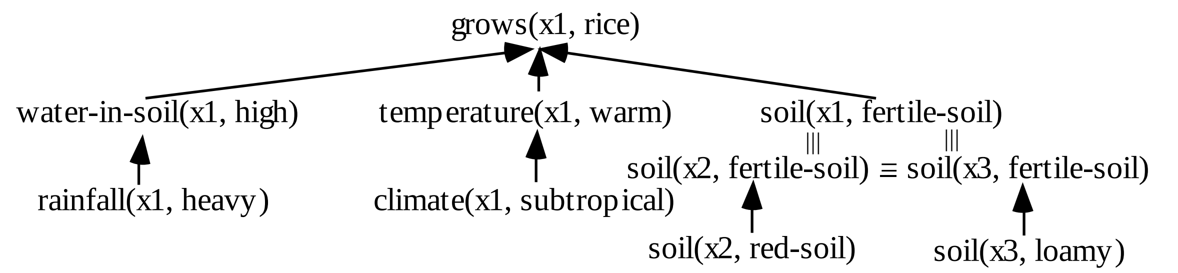

Figure 7. Explanation structure covering the plausible justification in Figure 6.

As the result of the unification of the connection patterns in Figure 7, one obtains the general justification tree in Figure 8 that covers the justification trees of both Example 1 and Example 2. It should be noticed that, although the justification trees of individual positive examples are AND trees, the generalization of these trees is, in general, an *AND/OR tree.* This is also the case with the tree in Figure 8. Indeed, "grows(x, rice)" is an AND node and "soil(x, fertile-soil)" is an OR node.

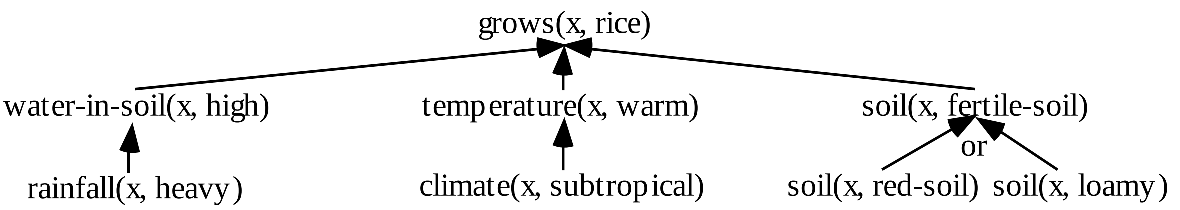

Figure 8. General justification tree covering the justifications of Example 1 and Example 2.

5.2 Generalization of the KB

The result of learning from Example 2 consists of a new fact

water-in-soil(Madagascar, high),

and a new positive example of the determination "rainfall(x, y) --> water-in-soil(x, z)"

(x<-Madagascar, y<-heavy, z<-high).

If Example 2 is the last example, then the system extracts the abstract and operational definitions of "grows(x, rice)" from the tree in Figure 8. In particular, the operational definition would be:

terrain(x, flat) & climate(x, subtropical) & (soil(x, red-soil) or soil(x, loamy))
: : > grows(x, rice)

6 Learning from negative examples

6.1 Specialization of the general justification tree

Let us now consider that the system receives the Negative Example 3 from Table 3. As indicated in Section 3. The system tries to specialize the general justification tree T_u so as not to cover any justification of the negative example. In the same time, it may need to specialize the KB so as to entail the new T_u while rejecting the previous T_u.

Again the system builds the instance of the general justification tree in Figure 8, corresponding to this new example (see Figure 9). This tree would lead to the wrong conclusion that the current input is a positive example of "grows(x, rice)". Therefore, the tree must contain some false leaf facts or false implications. These have to be detected, and both the general justification tree in Figure 8 and the KB should be specialized, so as no longer to contain them. One should notice that this is a limited specialization of the KB. Further specializing the KB so that no longer to entail *any* plausible justification of the negative example does not seem to be an obvious goal for a plausible reasoner that, by definition, may also reach some false conclusions.

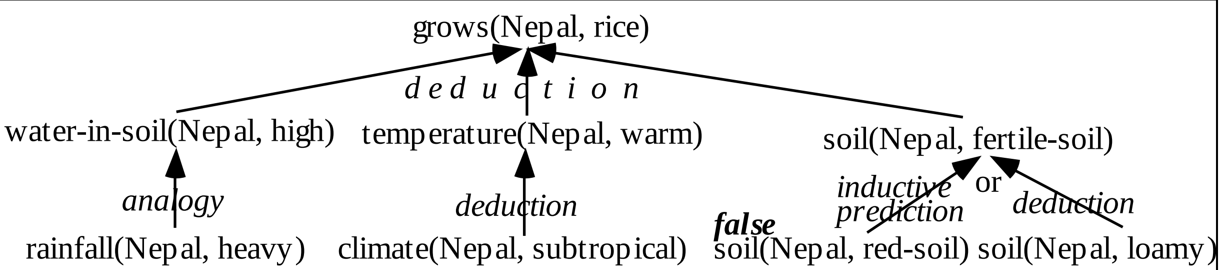

Figure 9. The instance of the justification tree in Figure 8 which is a wrong proof tree.

Because the tree in Figure 9 is an AND/OR tree, one should make sure to prove that enough of the leaf facts and implications are false. For instance, "soil(Nepal, red-soil)" in Figure 9 is false. However, because the node "soil(Nepal, fertile-soil)" is an OR node, the tree may still entail "grows(Nepal, rice)". Therefore, one should show that an implication is false.

Deciding which is the false implication is a difficult problem. In the current version of MTL, the implications hypothesized to be false are selected according to the following criteria:
- select the weakest implications (first abduction, then inductive prediction, then analogy, and lastly deduction);
- among the selected implications select those for which the corrections of the KB and of the general justification tree cause the minimum loss of coverage of the known instances;
- among the selected ones, select those for which the corrections produce a minimum increase in the complexity of the modified knowledge pieces;
- choose arbitrarily from the remaining hypotheses.

In the considered example, hypothesizing which is the false implication was simple because the justification tree from Figure 9 contains one analogical implication and three deductive implications. Therefore, the analogical implication was considered to be the false one:

rainfall(Nepal, heavy) -/-> water-in-soil(Nepal, high) (a)

The corresponding implication from the current general justification tree is

rainfall(x, heavy) —> water-in-soil(x, high) (b)

which was derived from the determination

rainfall(x, y) --> water-in-soil(x, z) (c)

Consequently, the system will try to specialize the rule (c) so as no longer to cover (a), by taking into account the known instances of (b) and (c):

rainfall(Philippines, heavy) —> water-in-soil(Philippines, high)
rainfall(Vietnam, heavy) —> water-in-soil(Vietnam, high)
rainfall(Madagascar, heavy) —> water-in-soil(Madagascar, high)

together with the known properties of the involved objects (Nepal, Philippine, Vietnam, and Madagascar).

The inductive learner of MTL will suggest, in this case, to specialize the determination (c) by adding the left-hand side predicate "terrain(x, flat)":

rainfall(x, y) & terrain(x, flat) --> water-in-soil(x, z)

The same specialization is applied to the implication (b). Thus, the general justification tree in Figure 8 is specialized as indicated in Figure 10.

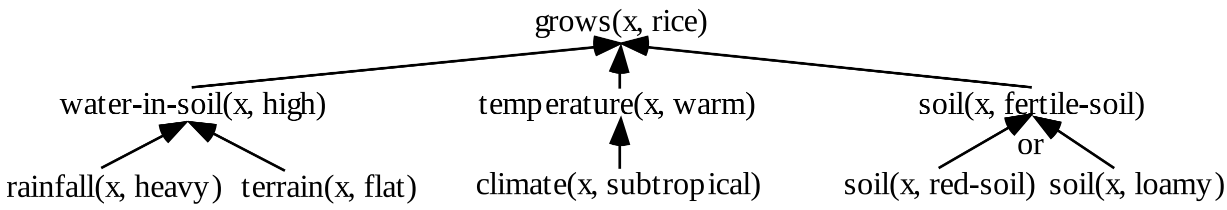

Figure 10. Specialization of the general justification tree.

6.2 Specialization of the KB

As a result of learning from the Negative Example 3, the system discovered a negative example of the plausible determination rule in Table 2, and specialized it, by conjunctively adding a left hand side predicate:

rainfall(x, y) & terrain(x, flat) --> water-in-soil(x, z)
with the positive examples: (x<-Philippine, y<-heavy, z<-high),
(x<-Vietnam, y<-heavy, z<-high),
(x<-Madagascar, y<-heavy, z<-high).
with the negative example: (x<-Nepal, y<-heavy).

It has also specialized accordingly the general justification tree T_u.

Because Negative Example 3 is the last input example, the system extracts from the tree T_u the operational and abstract definitions indicated in Table 4.

7 Basic cases

An important feature of the presented method is that it behaves as a single-strategy learning method whenever the applicability conditions of such a method are satisfied, and the learning task of MTL is specialized to the learning task of the single-strategy method.
This feature is important because it shows that the MTL method is a generalization of the integrated learning strategies which not only take advantage of the complementarity of the integrated strategies (as has been shown in the previous sections), but also inherits the

features of these strategies.

The next sections show that the MTL method may behave as:
- explanation-based learning, learning by abduction, or learning by analogy, when the input consists of only one positive example;
- multiple-example explanation-based generalization, when the input consists of a sequence of positive examples;
- empirical or constructive inductive generalization when the input consists of a sequence of positive and negative examples.

7.1 Explanation-based learning

Let us suppose that, in addition to the rules in Table 2, the KB also contains the following deductive rules:

∀x, rainfall(x, heavy) ==> water-in-soil(x, high)
∀x, soil(x, red-soil) ==> soil(x, fertile-soil)

In such a case, the justification trees in Figures 1 and 4 become logical proofs, and the result of learning from Example 1 is an operational definition of "grows(x, rice)". Thus, the MTL method reduces to explanation-based learning [DeJong & Mooney, 1986; Mitchell, Keller & Kedar-Cabelli, 1986].

7.2 Learning by abduction

Let us now suppose that the relationship between "rainfall" and "water-in-soil" is not a determination, but a deductive implication

∀x, rainfall(x,heavy) ==> water-in-soil(x,high)

and the KB does not contain examples of the predicate "soil". In this case, in order to build the justification tree of Example 1, the system needs only to create the explanatory hypothesis

soil(Vietnam, red-soil) ––> soil(Vietnam, fertile-soil)

as shown in Figure 11. Therefore, the result of learning is the created explanatory hypothesis, and the MTL method reduces to abductive learning.

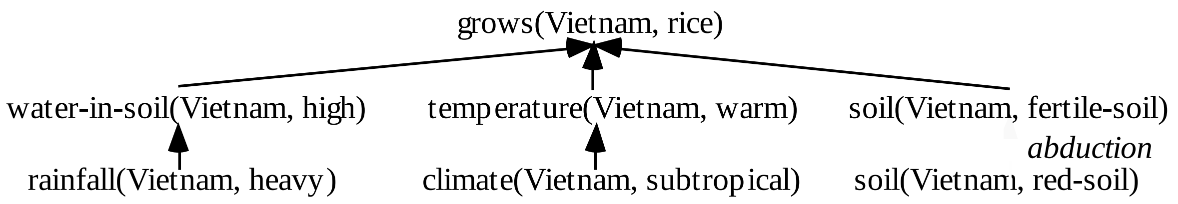

Figure 11. Using abduction to build a justification tree of the input.

7.3 Learning by analogy

If the only background knowledge that is related to Example 1 is

Facts:

rainfall(Philippine, heavy),

water-in-soil(Philippine, high)

Determination:

rainfall(x, y) --> water-in-soil(x, z)

then the system can only infer that "water-in-soil(Vietnam, high)", by analogy with "water-in-soil(Philippine, high)", as shown in Figure 2. Thus, in this case, the MTL method reduces to analogical learning.

7.4 Multiple-example explanation-based generalization

If the input of the system consists only of positive examples, that are deductively entailed by the KB, then the presented MTL method behaves as the multiple example explanation-based generalization, or mEBG, that was developed, among others, by [Hirsh, 1989; Kedar-Cabelli, 1985; Pazzani, 1988].

7.5 Empirical and constructive inductive generalization

Finally, let us assume that the KB does not contain the determination and the deductive rules shown in Table 2, and the input consists of all the examples from Table 3. In this case, each input is new, neither confirming nor contradicting the KB. Therefore, each example is interpreted as representing a single inference step that define a tree, as shown in the top part of Figure 12.

The MTL method will compute the least general generalization of the trees corresponding to the positive examples, generalization that does not cover the trees corresponding to the negative examples (see the bottom of Figure 12). The result of learning is therefore an operational definition of "grows(x, rice)", that represents the common properties of the positive examples that are not properties of the negative examples. Thus, in this case, the MTL method behaves as empirical or constructive inductive generalization [Michalski, 1992].

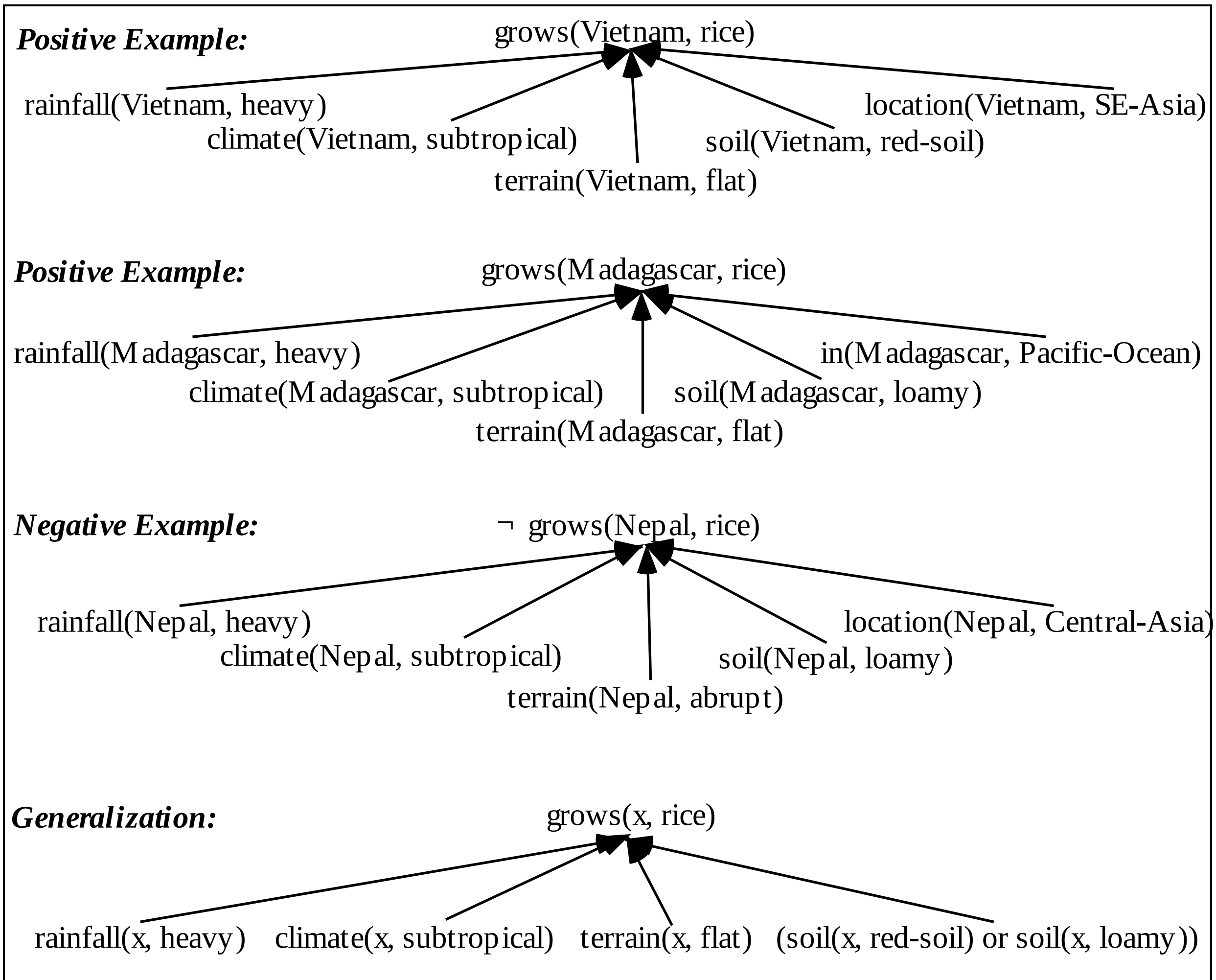

Figure 12. Empirical (or constructive) inductive generalization of examples.

8 Discussion and Conclusion

In this paper we proposed a general framework for multistrategy learning that is based on a dynamic integration of the elementary inference steps employed by the single-strategy learners. This framework was illustrated with a specific multistrategy task-adaptive learning (MTL) method.

There are several dimensions of generality of this framework.

First of all, it is extensible in that new types of inference, and therefore learning strategies, could naturally be added to the MTL method.

Secondly, it allows the use of different search strategies in the process of building plausible justification trees. The strategy employed in the current MTL method is a uniform-cost search of an AND-OR tree. However, one could employ any other search strategy (not only exhaustive but also heuristic).

Thirdly, it is general with respect to the knowledge representation, allowing learning from a great variety of knowledge pieces.

Finally, it is not only a framework for the integration of single-strategy learning methods, but also one for the generalization of these strategies, for the following reasons:

- the learning task subsumes the learning tasks of the integrated learning strategies;
- the MTL method behaves as a single-strategy learning method, whenever the applicability conditions for such learning are satisfied.

This approach to multistrategy learning has also revealed an new research direction in the theory of generalization by suggesting that, with each type of inference may be associated a certain type of generalization. Consequently, one could perform not only deductive and inductive generalizations, but also generalizations based on analogy, on abduction, etc.

The presented framework and method is also an illustration of synergistic combination of learning strategies. Indeed, the MTL method may learn in situations in which none of the integrated single-strategy methods would be sufficient.

Obviously, humans learn through a kind of multistrategy method. Although we do not claim that the presented method is a model of human learning, some features are similar to those employed by humans. These are the building of the justification tree of an example by using the justification trees of the previous examples [Wisniewski & Medin, 1991], and the use of multiple lines of reasoning in the justification of a plausible inference step [Collins & Michalski, 1989].

There are also several limitations and necessary developments of the presented framework and method that need to be addressed by the future research.

One limitation has already been mentioned in section 6.1: during learning from a negative example the KB is not specialized enough so that to guarantee that it no longer entails any justification tree that would prove that the example is positive.

Also, the presented method does not yet deal with noisy input. This is an intrinsically difficult problem for a plausible reasoner that may itself make wrong inferences. However, because the MTL method is a generalization of methods that could deal with noisy input, it inherits these capabilities. For instance, as in EBL, it may reject as noisy a negative example if it can build a deductive proof showing that the example is positive. Or, it may reject the negative example if the required specializations of the KB would determine a significant loss of coverage of instances of the knowledge pieces to be specialized.

The present versions of the integrated learning strategies (especially learning by analogy) are simple and should be replaced by more powerful ones.

Also new symbolic and even subsymbolic methods (as, for instance, reinforcement learning or neural network learning) should be integrated into the MTL method. This will also require elaboration of generalization techniques specific to each new strategy.

The method may also be extended so that to learn also from other types of input (like general pieces of knowledge, or input already known).

Another important research direction regards the extension and the application of the MTL method to the problem of knowledge acquisition from a human expert. In this case, the method would be extended with an important interactive component that would allow the system to ask different questions to the human expert, in order to decide on the best learning actions to take [Tecuci, 1991, 92]. In general, the human expert would be asked to solve the problems that are intrinsically difficult for a learning system as, for instance, *the credit-assignment problem* (i.e. assigning credit or blame to the individual decisions that led to some overall result) and *the new terms problem* (i.e. extending the representation language with new terms when this cannot represent the concept or the rule to be learned).

Acknowledgments

I have benefited from many useful discussions with Ryszard Michalski and from his work on the inferential theory of learning. I am also grateful to Ray Mooney and the reviewers for many useful comments and criticisms.

This research was done in the GMU Center for Artificial Intelligence. Research of the Center is supported in part by the National Science Foundation Grant No. IRI-9020266, in part by the Office of Naval Research grant No. N00014-91-J-1351, and in part by the Defense Advanced Research Projects Agency grant No. N00014-91-J-1854, administered by the Office of Naval Research.

References

Bergadano F., & Giordana A. (1990). Guiding Induction with Domain Theories. In Y. Kodratoff & R.S. Michalski (eds), *Machine Learning: An Artificial Intelligence Approach*, (Vol. 3). San Mateo, CA: Morgan Kaufmann.

Cox M., & Ram A. (1991). Using Introspective Reasoning to Select Learning Strategies. *Proceedings of the First International Workshop on Multistrategy Learning* (pp.217-230). Harpers Ferry, WV.

Carbonell J. (1986). Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition. In R.S. Michalski, J.G. Carbonell, & T.M. Mitchell (eds) *Machine Learning: An Artificial Intelligence Approach*, (Vol. 2). San Mateo, CA: Morgan Kaufmann.

Collins, A., & Michalski, R.S. (1989). The Logic of Plausible Reasoning: A Core Theory. *Cognitive Science*, 13, 1-49.

Danyluk, A.P. (1987). The Use of Explanations for Similarity-Based Learning. *Proceedings of the International Joint Conference on Artificial Intelligence* (pp. 274-276). Milan, Italy: Morgan Kaufmann.

Davies T.R. & Russell S.J. (1987). A Logical Approach to Reasoning by Analogy. *Proceedings of the International Joint Conference on Artificial Intelligence* (pp. 264-270). Milan, Italy: Morgan Kaufmann.

DeJong, G. & Mooney, R. (1986). Explanation-Based Learning: An Alternative View. *Machine Learning*, 1, 145-176.

De Raedt L. & Bruynooghe M. (1991). CLINT: A Multistrategy Interactive Concept Learner and Theory Revision System. *Proceedings of the First International Workshop on Multistrategy Learning* (pp. 175-190). Harpers Ferry, WV.

Flann, N., & Dietterich, T. (1989). A Study of Explanation-based Methods for Inductive Learning. *Machine Learning*, 4, 187-266.

Genest, J., Matwin, S., & Plante, B. (1990). Explanation-based Learning with Incomplete Theories: A Three-step Approach. In B. Porter & R. Mooney (eds), *Machine Learning: Proceedings of the Eighth International Workshop.* Texas, Morgan Kaufmann.

Gentner D. (1983). Structure-Mapping: a Theoretical Framework for Analogy. *Cognitive Science*, 7, 155-170.

Hirsh, H. (1989). Incremental Version-space Merging: A General Framework for Concept Learning. *Doctoral dissertation*, Stanford University.

Josephson J. (1991). Abduction: Conceptual Analysis of a Fundamental Pattern of Inference. *Technical Research Report 91-JJ*, Laboratory for Artificial Intelligence Research, The Ohio State University.

Kedar-Cabelli S. (1985). Purpose-Directed Analogy. *Proceedings of the Cognitive Science*

Society (pp.150-159). Irvine, California.

Kodratoff, Y. (1990). Using Abductive Recovery of Failed Proofs for Problem Solving by Analogy. In B. Porter & R. Mooney (eds), *Machine Learning: Proceedings of the Eighth International Workshop,* Texas, Morgan Kaufmann.

Kodratoff, Y., & Michalski R.S., (eds). (1990). *Machine Learning: An Artificial Intelligence Approach* (Vol. 3). San Mateo, CA: Morgan Kaufmann.

Lebowitz, M. (1986). Integrated Learning: Controlling Explanation. *Cognitive Science*, 10, 219-240.

Mahadevan S. (1989). Using Determinations in Explanation-based Learning: A Solution to Incomplete Theory Problem. *Proceedings of the Sixth International Workshop on Machine Learning.* Ithaca, NY: Morgan Kaufman.

Michalski, R. S. (1983). A Theory and Methodology of Inductive Learning. In R.S. Michalski, J.G. Carbonell, & T.M. Mitchell (eds.), *Machine Learning: An Artificial Intelligence Approach* (Vol. 1). Tioga Publishing Co.

Michalski, R.S. (1990). Toward a Unified Theory of Learning: Multistrategy Task-Adaptive Learning. *Reports of Machine Learning and Inference Laboratory*, MLI 90-1, Center for Artificial Intelligence, George Mason University.

Michalski, R.S. (1992). Inferential Learning Theory as a Conceptual Basis for Multistrategy Learning, in this issue.

Michalski R.S., Carbonell J.G., Mitchell T.M. (eds). (1983). *Machine Learning: An Artificial Intelligence Approach* (Vol. 1). Tioga Publishing Co.

Michalski R.S., Carbonell J.G., Mitchell T.M. (eds). (1986). *Machine Learning: An Artificial Intelligence Approach* (Vol. 2). San Mateo, CA: Morgan Kaufmann.

Minton, S., Carbonell, J.G. (1987). Strategies for Learning Search Control Rules: An Explanation-based Approach. *Proceedings of the International Joint Conference on Artificial Intelligence* (pp. 228-235). Milan, Italy: Morgan Kaufmann.

Mitchell T.M. (1978). Version Spaces: An Approach to Concept Learning. *Doctoral dissertation,* Stanford University.

Mitchell, T.M., Keller, T., & Kedar-Cabelli, S. (1986). Explanation-Based Generalization: A Unifying View. *Machine Learning*, 1, 47-80.

Mooney R., Bennet S. (1986). A Domain Independent Explanation Based Generalizer. *Proceedings of the Fifth National Conference on Artificial Intelligence* (pp. 551-555). Philadelphia, PA: Morgan Kaufmann.

Mooney R. & Ourston D. (1991). A Multistrategy Approach to Theory Refinement. *Proceedings of the First International Workshop on Multistrategy Learning* (pp. 115-130). Harpers Ferry, WV.

Morik K. (1992). Balanced Cooperative Modeling, in this issue.

Nilsson, N. (1971). Problem Solving Methods in Artificial Intelligence. McGraw-Hill.

Pazzani, M.J. (1988). Integrating Explanation-Based and Empirical Learning Methods in OCCAM. *Proceedings of the Third European Working Session on Learning* (pp. 147-166). Glasgow, Scotland.

Porter B., Bareiss R., & Holte R. (1990). Concept Learning and Heuristic Classification in Weak-Theory Domains. In J. W. Shavlik and T. Dietterich (eds), *Readings in Machine Learning.* Morgan Kaufmann.

Reich, Y. (1991). Macro and Micro Perspectives of Multistrategy Learning. *Proceedings of the First International Workshop on Multistrategy Learning* (pp. 97-112). Harpers Ferry, WV.

Saitta L. & Botta M. (1992). Multistrategy Learning and Theory Revision. in this issue.

Shavlik J.W., & Towell G.G. (1990). An Approach to Combining Explanation-based and Neural Learning Algorithms. In J. W. Shavlik and T. Dietterich (eds), *Readings in Machine Learning.* Morgan Kaufmann.

Tecuci G. (1988). DISCIPLE: A Theory, Methodology, and System for Learning Expert Knowledge. *Ph.D. Thesis*, University of Paris-South.

Tecuci G. (1991). Steps Toward Automating Knowledge Acquisition for Expert Systems. *Proceedings of the AAAI-91 Workshop on Knowledge Acquisition: From Science to Technology to Tools.* Anaheim, California.

Tecuci G. (1992). Automating Knowledge Acquisition as Extending, Updating, and Improving a Knowledge Base. *IEEE Transactions on Systems, Man and Cybernetics,* to appear.

Tecuci G. & Kodratoff Y. (1990). Apprenticeship Learning in Imperfect Theory Domains. In Y. Kodratoff & R.S. Michalski (eds), *Machine Learning: An Artificial Intelligence Approach* (Vol. 3). San Mateo, CA: Morgan Kaufmann.

Tecuci G., & Michalski R.S. (1991a). A Method for Multistrategy Task-adaptive Learning Based on Plausible Justifications. In L. Birnbaum, & G. Collins (eds), *Machine Learning: Proceedings of the Eighth International Workshop* (pp. 549-553). Chicago, IL: Morgan Kaufmann.

Tecuci G., & Michalski R.S. (1991b). Input Understanding as a Basis for Multistrategy Task-adaptive Learning. In *Proceedings of the International Symposium on Methodologies for Intelligent Systems* (pp. 419-428). Charlotte, NC: Springer-Verlag.

Whitehall, B.L. (1990). Knowledge-based Learning: Integration of Deductive and Inductive Leaning for Knowledge Base Completion. *PhD Thesis*, University of Illinois at Champaign-Urbana.

Widmer, G. (1989). A Tight Integration of Deductive and Inductive Learning. *Proceedings of the Sixth International Workshop on Machine Learning.* Ithaca, NY: Morgan Kaufman.

Widmer, G. (1991). Learning by Plausible Reasoning and its Application to a Complex Musical Problem. *Proceedings of the First International Workshop on Multistrategy Learning* (pp. 411-418). Harpers Ferry, WV.

Wilkins, D.C. (1990). Knowledge Base Refinement as Improving an Incorrect and Incomplete Domain Theory. In Y. Kodratoff & R.S. Michalski (eds), *Machine Learning: An Artificial Intelligence Approach* (Vol. 3). San Mateo, CA: Morgan Kaufmann.

Winston P.H. (1986). Learning by Augmenting Rules and Accumulating Censors. In R.S. Michalski, J.G. Carbonell, & T.M. Mitchell (eds), *Machine Learning: An Artificial Intelligence Approach* (Vol. 2). San Mateo, CA: Morgan Kaufmann.

Wisniewski E.J., & Medin D.J. (1991). Is it a Pocket or a Purse ? Tightly Coupled Theory and Data Driven Learning. In L. Birnbaum and G. Collins (eds), *Machine Learning: Proceedings of the Eighth International Workshop* (pp. 564-568). Chicago, IL: Morgan Kaufmann.