Proceedings of the First International Workshop on Multistrategy Learning, Harpers Ferry, WV, November, 1991. 217

Using Introspective Reasoning to Select Learning Strategies

Michael Cox and Ashwin Ram
cox@cc.gatech.edu ? ashwin@cc.gatech.edu Georgia Institute of Technology ? College of Computing Atlanta, GA 30332-0280

Abstract

In order to learn effectively, a system must not only possess knowledge about the world and be able to improve that knowledge, but it also must introspectively reason about how it performs a given task and what particular pieces of knowledge it needs to improve its performance at the current task. Introspection requires a declarat?ive representation of the reasoning performed by the system during the performance task. This paper presents a taxonomy of possible reasoning failures that can occur during this task, their declarative representations, and their associations with particular learning strategies. We propose a theory of Meta-XPs, which are explanation structures that help the system identify failure types and choose appropriate learning strategies in order to avoid similar mistakes in the future. A program called Meta-AQUA embodies the theory and processes examples in the domain of drug smuggling.

1 Introduction

In order to learn effectively, a system must not

only possess knowledge about the world and be able to improve that knowledge, but it also must introspectively reason about how it performs a given task and what particular pieces of knowledge it needs to improve its performance at the current task. In addition, the learner needs to focus its learning if it is to avoid the combinatorial explosion of inferences and search necessary in complex, unrestricted situations.

The approach to learning taken in this research is failure-driven. "Failures" are not simply performance errors, but include expectation failures, or anomalous situations which do not match the constraints on a given concept. (In fact, an expectation failure could occur even if the performance task is successful.) When such a failure occurs, the system posts a knowledge goal which drives the reasoner to explain or otherwise resolve the gaps in its knowledge. The knowledge goals of a system are the questions that a system poses about the world and events within the world. In order to learn from a failure and to avoid repeating the mistake, the system needs to identify the cause of the failure and then, depending upon the cause, apply a given learning strategy.

In this paper, we propose a theory of Meta-

XPs1, which are causal explanation structures that explain how and why an agent reasons, and that help the system in the learning task. Our theory of reasoning and learning is based on these structures. There are two broad classes of Meta-XPs. General Meta-XPs record a trace of the reasoning performed by a system. Introspective MetaXPs are structures used to explain and learn from a reasoning failure. They associate a failure type with a particular set of learning strategies and point to likely sources of the failure within the general Meta-XP. Our theory deals with three types of reasoning failures:

Novel Situation - An expectation failure can arise when the reasoner does not have the appropriate knowledge structures to deal with a situation. The situation is said to be anomalous with respect to the current knowledge in the system. In such a situation, the reasoner could use a variety of learning strategies, including explanation-based generalization, refinement, and index learning.

Incorrect World Model - Even if the reasoner has knowledge structures that are applicable to the situation, these knowledge structures may be incomplete or incorrect. Learning in such situations is usually incremental, and involves strategies such as elaborative question asking applied to the reasoning chain and generalization techniques in conceptual memory.

Mis-Indexed Structure - The reasoner may have an applicable knowledge structure, but it may not be indexed in memory such that it can be retrieved using the particular cues provided by the context. In this case the system must add a new index, or generalize an existing index based on the context. If on the

1Meta-XPs are based on explanation patterns (XPs) described in Kass, Leake, & Owens (1986), Ram (1990a), and Schank (1986).

other hand, the reasoner retrieves a structure that later proves inappropriate, it must specialize the indices to this structure so the retrieval will not recur in similar situations.

We propose a multi-strategy learning approach in which the reasoning system records a declarative trace of its own reasoning process using a general Meta-XP. The data structure holds explicit information concerning the manner in which knowledge gaps are identified, the reasons why hypotheses are generated, how hypotheses are verified, and the basis for choosing particular reasoning methods for each of these. If the system encounters a reasoning failure, it then uses introspective Meta-XPs to examine the declarative reasoning chain. The introspective Meta-XP performs two functions: it aids in blame assignment (determining which knowledge structures are missing, incorrect or inappropriately applied), and it aids in the selection of appropriate learning algorithms to recover and learn from the reasoning error. Such self-explanations augment a system's ability to introspectively reason about its own knowledge, gaps within this knowledge, and the reasoning processes which attempt to fill these gaps. The use of explicit Meta-XP structures allow direct inspection of the reasons by which knowledge goals are posted and processed, thus enabling a system to improve its ability to reason and learn.

Section 2 first presents an implemented example to motivate the problem. Next the methodology used to support introspective reasoning is outlined in section 3. Section 4 explains the reasoning model upon which representations for reasoning traces are based. Section 5 covers the representation of introspective structures that capture the three failure types listed above, whereas section 6 illustrates how learning is associated with the structures. The paper closes with discussion and future direction for research.

2 Motivational Example: The Drug Bust

This research implements an introspective version of AQUA (Ram, 1989, 1991), called Meta-AQUA. AQUA is a question-driven story understanding system that learns about Middle Eastern terrorist activities. Its performance task is to "understand" the story by building causal explanations that link the individual events into a coherent whole. The Meta-AQUA version adds introspective reasoning and learning using Meta-XP structures. Unlike AQUA, Meta-AQUA does not actually parse the sentences; since this research does not deal with the natural language understanding problem, we assume that input sentences are already represented conceptually. To illustrate the type of introspection Meta-AQUA performs and the type of learning that results, consider the following passage:

S1: A police dog sniffed at a passenger's luggage in the Atlanta airport terminal.

S2: The dog suddenly began to bark at the luggage.

S3: At this point the authorities arrested the passenger, charging him with smuggling drugs.

S4: The dog barked because it detected two kilograms of marijuana in the luggage.

A number of inferences can be made from this story, many of which may be incorrect, depending on the knowledge of the reader. Meta-AQUA's knowledge includes general facts about dogs and sniffing, including the fact that dogs bark when threatened, but it has no knowledge of police drug dogs in particular. It also knows of past terrorist smuggling cases, but has never seen a case of drug interdiction. Nonetheless the program is

able to recover and learn from the erroneous inferences this story generates.

The line of reasoning that Meta-AQUA produces by processing this story is as follows:

S1 produces no inferences other than the observation that sniffing is a normal event in the life of a dog.

However, S2 produces an anomaly because the system's definition of bark specifies that the object of the bark is animate. In this example, the program (incorrectly) believes that dogs bark only when threatened by animate objects. Since luggage is inanimate, there is a contradiction, leading to an expectation failure. This anomaly causes the understander to ask why the dog barked at an inanimate object. This question may lead the system to learn something useful about dogs at some point in the future. Until this question is answered, however, the system can only assume (again, incorrectly) that the luggage somehow threatened the dog.

S3 asserts an arrest scene which reminds Meta-AQUA of a prior incident of weapons smuggling by terrorists. The system then infers the existence of a smuggling bust that includes detection, confiscation, and arrest scenes. Because baggage searches are the only detection method the system knows, the sniffing event remains unconnected to the rest of the story.

Finally, S4 causes the question generated by S2 "Why did the dog bark?" to be retrieved, and the understanding task is resumed. Instead of revealing the anticipated threatening situation, S4 produces another hypothesis. The program prefers the explanation given by S4 over the earlier one, because it links more of the story together (e.g., see Alterman, 1985; Ng & Mooney, 1990; Thagard, 1989).

220

The system uses the trace of its reasoning process (stored in a general Meta-XP) to review the understanding process. It characterizes the reasoning error as one in which there is an expectation failure caused by the incorrect retrieval of a known explanation ("dogs bark when threatened by animate objects", erroneously assumed to be applicable), and a missing explanation ("the dog barked because it detected marijuana", the correct explanation in this case). Using this characterization as an index, the system retrieves the introspective Meta-XP XP- Novel-Situation-AlternativeRefuted.

This composite Meta-XP consists of three primitive Meta-XPs: XP-NovelSituation, XP-Mis-IndexedStructure, and XP-IncorrectWorld-Model. The XP-NovelSituation directs an explanation-based generalization (EBG) algorithm (DeJong & Mooney, 1986; Mitchell, Keller, & KedarCabelli, 1986) to be applied to the node representing the explanation of the bark. Since the detection scene of the drug-bust case and the node representing the sniffing are unified due to the explanation given in S4, the explanation is generalized to drug busts in general. The general explanation is then indexed in memory. The XP-MisIndexed-Structure directs an indexing algorithm to the defensive barking explanation. It recommends that the explanation be re-indexed so that it is not retrieved in similar situations in the future. Thus the index for this XP is specialized so that retrieval occurs only on animate objects, not physical objects in general. The XP- Incorrect-World-Model directs the system to examine the source of the story's anomaly. The solution is to alter the conceptual memory representation so that the constraint on the object of dog-barking instantiations is generalized to physical

objects, not just animate objects.

Though the program is directly provided an explanation which links the story together, Meta-AQUA performs more than mere rote learning. It learns to avoid the mistakes made during the processing of the story. The application of Meta-XPs allows the system to use the appropriate learning strategy (or multiple strategies) to learn exactly that which the system needs to know to process similar situations in the future correctly. This is essentially a case-based or experience-based approach, which relies on the assumption that it is worth learning about one's experiences since one is likely to have similar experiences in the future (see, e.g., Hammond, 1986; Kolodner & Simpson, 1984; Ram, 1990b; Schank, 1982).

3 Methodology

We assume that understanding involves building causal explanations of the input, which provide conceptual coherence to the story by tying the pieces of the story together. Explanations are built by applying known explanation patterns to the events in the story. Expectation failures arise when the world differs from the system's expectations. For example, the system may be faced with an anomalous situation in which the explanation pattern that the system believes to be applicable turns out to be contradicted in the story.2 When the system encounters an anomalous situation, it tries to retrieve and apply a known explanation to the anomalous concept. The process of explanation generates questions, or knowledge goals, representing what the system needs to know in order to be able to explain similar situations in the future, thus avoiding repeated similar failures (Ram,

2 If the system predicts a performance failure in a situation which turns out to be successful, we still say that the system has encountered a (prediction) failure.

1990a, 1991).

Explanation patterns (XPs) are similar to justification trees, linking antecedent conditions to their consequences. The XP is essentially a directed, acyclic graph of concepts, connected with RESULTS, ENABLES and INITIATES links. A RESULTS link connects a process with a state, while an ENABLES link connects a precondition state to a process. An INITIATES link connects two states.

The set of sink nodes in the graph is called the PRE-XP-NODES. These nodes represent what must be present in the current situation for the XP to apply. One distinguished node in this set is called the EXPLAINS node. It is bound to the concept which is being explained. Source nodes are termed XP-ASSERTED- NODES. All other nodes are INTERNAL-XP-NODES.

For an XP to apply to a given situation, all PRE-XP-NODES must be in the current set of beliefs. If they are not, then the explanation is not appropriate to the situation. If the structure is not rejected, then all XP-ASSERTED- NODES are checked. For each XP- ASSERTED node verified, all INTERNAL- NODES connected to it are verified. If all XP-ASSERTED-NODES can be verified, then the entire explanation is verified. Gaps in the explanation occur when one or more XP-ASSERTED-NODE remains unverified. Each gap results in a question, which provides the system with a focus for reasoning and learning, and limits the inferences pursued by the system. Thus a question or knowledge goal can be viewed as a goal to learn.

The background knowledge used in the current implementation consists of a framebased conceptual hierarchy, a case library of past episodes, and an indexed collection of XPs.

For the task of story understanding, MetaAQUA employs the algorithm outlined in figure 1. First, the outer loop inputs a sentence representation and checks to see if the concept can answer a prior question. If it can, the reasoning associated with the question is resumed. Otherwise, the concept is passed on to the understanding algorithm. The understanding algorithm consists of four phases: Question identification, hypothesis generation, verification, and review/learning.

Input Sentence

Suspend Task

More
Input?

Answers
Previous

Question?

Pose Question
Interesting
?

Can Generate
Hypothesis?
Skim

Resume
Previous Task

Start

N

Y
N

N

Y

Y

Generate
Hypothesis

Y

Suspend Task
Can
Verify? N

Verify
Hypothesis

Learn

Is there an
appropritate
Meta-XP?

Y

Y

N

Halt
N

Figure 1. Meta-AQUA control flow

The first phase looks for questions associated with the concept by checking the concept for interesting characteristics. Meta-AQUA considers explanations, violent acts, and anomalies to be interesting. Explanations and violent acts are detected by the concept type of the input. Anomaly checking is performed by comparing the input to the conceptual

definitions found in the conceptual hierarchy. If a concept contradicts a constraint, then an anomaly exists3 and a question is posed. Such questions will represent the knowledge goals of the program. If no anomaly is detected, then the concept is instantiated and control passes back.

If a knowledge goal is posted, then the understander attempts to answer the question by generating a hypothesis. The basis of this decision, i.e., what knowledge is relevant in making the determination, is then recorded in the Meta-XP. Strategies for hypothesis generation include application of known explanation patterns ("XP application"), casebased reasoning (Hammond, 1986; Kolodner & Simpson, 1984), and analogy. If none of these applies, then the process is suspended until a later opportunity.

When a hypothesis is generated it is passed to the verification subsystem. Strategies for hypothesis verification include devising a test (currently not implemented), comparison to known concepts, and suspension of the reasoning task.

The system reviews the chain of reasoning after the verification phase is complete. The review process examines the general Meta-XP trace to see if there was a reasoning failure. If a failure occurred, then the review process searches for an introspective explanation. If a Meta-XP is retrieved, then it is applied to the error. Meta-AQUA then checks to see if all XP-ASSERTED-NODES are in the set of current beliefs. If so, the learning algorithm associated with the XP is executed. If there are XP-ASSERTED-NODES not in the set of current beliefs, then a question is posed on the Meta-XP itself.

3See Leake (1989) and Ram (1989) for more sophisticated XP-based approaches to anomaly detection.

Since learning is moderated by the XP application algorithm, it is necessary to represent the understanding process outlined above in a declarative manner. This allows matching and syntactic functions to be applied to the prior reasoning. Further, it allows the system to pose knowledge goals about aspects of the reasoning process itself.

4 A model of reasoning about knowledge goals

The AQUA system embodies a theory of motivational explanation based on decision models (Ram, 1990a) which model the decision process that an agent goes through in deciding whether to perform an action. For example, the religious fanatic explanation for suicide bombing is a decision model describing why a bomber should choose to perform a terrorist strike in which the bomber dies. AQUA's model claims that an agent first considers its goals, goal priorities, and the expected outcome of performing the action. The agent then makes a decision whether or not to enter into such a role, and if so, performs the action. This paper extends the model to account for reasoning about knowledge goals.

Reasoning about knowledge goals is performed in a similar manner. A set of states, priorities, and the expected strategy outcome prompt the reasoner to make a strategy decision. Based on its general knowledge, current representation of the story, and any inferences that can be drawn from this knowledge, the reasoner chooses a particular reasoning strategy. Once executed, a strategy may produce further questions and hypotheses. Each execution node explicitly represents its main result (structure returned by the function) and its side-effect.

These decide-compute combinations are chained into threads of reasoning such that each one initiates the goal which drives the next. Though the chains can vary widely, in the task of question-driven story understanding, the chains take the form shown in figure 2. Learn/Review

G

G

G

G
Understanding

Question

Identification

Generate

Hypothesis

- Case-Based Reasoning
- Explanation
- Analogy
- Suspend

Available Strategies:

Available Strategies:

- Question Posing
- Skimming

Alternative Strategies:

Verify

Hypothesis

- Test Hypothesis
- Compare To Input
- Suspend

Available Strategies:

Dependent on
Introspective
Meta-XP

Available Strategies:
Figure 2. Phases of understanding

This reasoning process is recursive in nature. For example, if a hypothesis generates a new question, then the reasoner will spawn a recursive regeneration of the sequence.

When insufficient knowledge exists on which to base a decision, a useful strategy is to simply defer making the decision. The reasoning task is suspended and later continued if and when the requisite

knowledge appears. This is a form of opportunistic reasoning (Birnbaum & Collins, 1984; Hammond, 1988; Hayes-Roth & Hayes-Roth, 1979; Ram, 1989).

A general Meta-XP, representing the trace of the reasoning process, is a chain of decidecompute-nodes (D-C-NODES). These nodes record the processes that formulate the knowledge goals of a system, together with the results and reasons for performing such mental actions. As such, the trace of reasoning is similar to a derivational analogy trace as described by Carbonell (1986). Such a MetaXP is a specific explanation of why a reasoner chooses a particular reasoning method and what results from the strategy. Like an XP, the Meta-XP can be a general structure applied to a wide range of contexts, or a specific instantiation which records a particular thought process.

One distinguishing property of general MetaXPs is the notion that a decision at one stage is often based on features in previous stages. For example, the decision of how to verify a hypothesis may be based on knowledge used to initially construct the hypothesis. This property is particularly true of the learning stage, which by definition is based on prior processing.

An understanding system may attempt to retrieve and apply a Meta-XP, much the way standard XPs are used in explanation. If the antecedent conditions of the Meta-XP exist, then the structure will point to an appropriate learning algorithm without having to analyze all current states in the story representation. This approach provides significant speedup learning, relying on past successes and failures instead of reasoning from first principles. For example, even though some subquestions on an erroneous hypothesis are verified, Meta-XPs will direct the search for the blame on the basis of the decision to use a given hypothesis generation strategy, not on

224

the basis of the verification strategy.

5 Representation of Introspective Meta-XPs

A Meta-XP is similar to a standard XP in that it is an explanatory causal structure. The major difference between the two is that instead of presenting a causal justification for a physical relation (such as why people look like their ancestors) or a volitional role relation (such as why a person performs a given action), a Meta-XP explains how and why an agent reasons in a particular manner. Thus the representation of a Meta-XP must be able to account for reasoning failures and successes. The three types of failures discussed in the introduction (novel situations, incorrect or incomplete world knowledge, and mis-indexed knowledge structures) can be accounted for with the complementary notions of expectation failure and retrieval failure. Though successful predictions produce no learning in Meta-AQUA, the mental event has a representation.

To illustrate the representation, let node A be an actual occurrence of an event in the world, an explanation, or an arbitrary proposition. Let node E be the expected occurrence. Now if the two propositions are identical so that A = E, or A is a superset of E, then a successful prediction has occurred. If on the other hand, A is a subset of E, then there are more questions remaining on the predicted node E. If there are unanswered questions, then the system will wait for more information before it introspects. Such cases are not represented in our current implementation, though there are cases in which one would want to reason about partial computations.

Failures occur when A ? E. This state exists when either A and E are disjoint or there are conflicting assertions within the two nodes.

For example, A and E may be persons, but E contains a slot specifying gender = male, whereas A contains the slot gender = female.

A E

Successful

Prediction

=

Actual
Outcome
Expected
Outcome

Mentally
Initiates

Mentally
Results
Mentally
Results

New

Input

Reasoning

Chain

domain domain

co-domain
co-domain

domain

co-domain

Figure 3. Successful prediction

The representation of a successful prediction is shown in figure 34. The EXPLAINS node of the XP is the node marked "Successful Prediction". It is mentally-initiated by the equals relation between A and E. The node A results from either a mental calculation or an input concept. The expected node E mentallyresults from some reasoning trace.

Expectation failures occur when the reasoner predicts one event or feature, but another occurs instead. The structure representing such a failure is nearly identical to the representation for successful prediction, except that the outcome is initiated by a notequals relation instead of the equals relation. Figure 4 shows the representation of an expectation failure.

4One should note that figures depict network representations of equivalent frame structures used in the implementations. Slot-filler and other relations are often represented explicitly as frame structures. Thus the ACTOR slot of event X with value Y is equivalent to the relation frame ACTOR having domain X and co-domain Y.

A E

Expectation

Failure

?

Actual
Outcome
Expected
Outcome

Mentally
Initiates

Mentally
Results
Mentally
Results

New

Input

Reasoning

Chain

domain domain

co-domain
co-domain

domain

co-domain

Figure 4. Expectation failure

Retrieval failure has the same possibilities, although the difference here is that instead of an expectation (E) being present, it is instead absent due to the inability of the system to retrieve the knowledge structures that would predict E (see figure 5). To represent these conditions Meta-AQUA uses standard nonmonotonic logic values of in (in the current set of beliefs) and out (out of the current set of beliefs) (Doyle, 1979). Added to these are the values hypothesized-in (weakly assumed in), hypothesized-out (weakly assumed out), and hypothesized (unknown) (Ram, 1989). Thus absolute retrieval failure is represented by A [truth = in] = E [truth = out]. Cuts across links in the figure signify causal relations for which the truth slot of the frame is out.

A novel situation is structurally like a retrieval failure, except the node M has a truth value of out. That is, there is no item in memory that can be retrieved and reasoned with to produce the expectation of a concept like A.

Using this notation, the system can represent five possible combinations. They are: novel situation, novel situation with expectation

failure, retrieval failure, and expectation failure combined with retrieval failure.

A E

Retrieval

Failure

=

Actual
Outcome
Expected
Outcome

Mentally
Initiates

Mentally
Results
Mentally
Results

New

Input

Reasoning

Chain

domain domain

co-domain
co-domain

domain

co-domain truth = out

M

Mentally
Initiates

truth = out

truth = out

truth = out

Figure 5. Retrieval failure

Table 1 summarizes the possibilities along with the associated learning to be applied. Note that the node A is assumed in for all entries. In addition, for the two combination Meta-XPs in the table, E? represents the concept that should have been predicted, but was not. M? is the memory item that should have triggered its retrieval, but did not.

6 Associating Learning Strategies with Introspective Meta-XPs

Novel situations occur when A ? E and the E node's truth slot is either hypothesizedin or out. In the case of E being hypothesized-in, there is an accompanying expectation failure. When a novel situation is identified, Meta-AQUA performs EBG on the node A so that the frame can be applied to a wider set of future events. The Meta-XP for novel situations also directs an indexing algorithm to the same node so that it will be retrieved in similar situations.

Table 1. Truth values on Meta-XP nodes

There are two instances of XP-MisIndexed-Structure. One is the case in which the EXPLAINS node is an expectation failure. In the other instance, the EXPLAINS node is a retrieval failure. In the former, the Meta-XP directs a specialization algorithm to assure that the retrieval will not recur given similar situations. The latter case has an XP- ASSERTED-NODE pointing to a node in memory (M) that must be in. If this can be verified, then the Meta-XP directs an indexing algorithm to examine the indices of M, looking for an index compatible with the index calculated for A. If found, this index is generalized so that the current cues provided by the context of A will retrieve E. If no such index is found then a new index is computed. If M cannot be found, then a reasoning question is raised concerning the possibility that M exists. The question is represented as a knowledge goal and indexed by the context of A, and the process is suspended.

For the failure type XP-IncorrectWorld-Model, only one instance is currently represented (see figure 6). The instance of this type of failure is an inconsistency with a known fact and a constraint on the isa-hierarchy. This constraint is one that previously caused an anomaly during the question identification phase of reasoning. When the program invokes the Meta-XP, it will check if the two assertions are siblings in the hierarchy. If this is true, then the program will generalize the constraint to its parent on the basis of induction. The constraint is then marked as being hypothesized-in. The reasoning chain which led to this hypothesis is then indexed off the hypothesis so that it can be retrieved when the constraint is used in future stories. If the anomalous assertion is re-encountered in another situation, then the hypothesis is verified.

E E? M M? RC SP EF RF Learning Successful
Prediction in ? in ? in in out out No Learning.

Novel EBG on A. Situation out ? out ? out out out in Index A by context.

Retrieval
Failure out ? in ? out out out in Generalize index on M.

Novel Sit. + hypo EBG on A. Expectation -in out in out out out in in Index A by context. Failure Specialize index on M.

Expectation hypo Specialize index on M. Failure + -in out in in in out in in Generalize index on M?. Retrieval Failure
Key: ?=don't care; hypo-in=hypothesized-in; RC= Reasoning Chain; SP=Successful Prediction; EF=Expectation Failure; RF=Retrieval Failure.

A failure may also be due to the inferences used to base a decision in the hypothesis generation phase. The error is found by searching all hypothesis generation D-C- NODES on the path from the EXPLAINS node of A to E performing elaborative question asking (Ram, 1990b). This case has not yet been represented declaratively. MetaAQUA reasons about it using a general search heuristic for blame assignment.

?

A

Falsifies

domain co-domain

Constraint

isa isa

Parent

Figure 6. XP-Incorrect-World-Model

Figure 7 shows the composite Meta-XP which is used to direct learning in the example from section 2. The XP combines an XP-NovelSituation, an XP-Mis-IndexedStructure, and an XP-IncorrectWorld-Model. A, the actual outcome, is bound to the explanation from S4, whereas E, the expected outcome, is bound to the explanation that dogs bark at objects which threaten them. C is bound to the constraint that dogs bark at animate objects. The concept in memory, M, is bound with the index used to retrieve E.

A E

Expectation

Failure

?

Actual
Outcome
Expected
Outcome

Mentally
Initiates

Mentally
Results
Mentally
Results

New

Input

domain domain

co-domain
co-domain

domain

co-domain

Actual
Outcome Expected Outcome

domain domain

=

co-domain

Mentally
Initiates

domain

E?

co-domain

Retrieval

Failure

Hypothesis

Generation

Question
Identification

Question

Mentally
Results

Mentally
Enables

Explains

domain

co-domain

Decision
Basis
domain

domain

Decision
Basis
C
co-domain

Falsifies

domain

co-domain

truth = out

truth = hypothesized-in

co-domain

M
co-domain

Figure 7.
XP-Novel-Situation-Alternative-Refuted

7 Discussion and Future Research

Although the implementation presented here is preliminary, we have demonstrated that use of introspection by applying Meta-XPs to declarative representations of the reasoning process can aid a reasoner's ability to perform blame assignment, and direct the learning algorithms which allow a reasoner to recover from failures and to learn not to repeat the failure. The use of Meta-XP structures aids in the blame assignment problem, since all points in the reasoning chain do not have to be inspected. This helps in controlling the search process. Because answers may not be available at the time questions are posed, an opportunistic approach allows the system to improve its knowledge incrementally and to

answer its questions at the time the information it needs becomes available. The representation also allows the system to pose questions about its own reasoning.

The use of Meta-XPs in reasoning about knowledge goals during story understanding provides a number of benefits. Because general Meta-XPs make the trace of reasoning explicit, an intelligent system can directly inspect the reasons supporting specific conclusions. This avoids hiding knowledge used by the system in procedural code. Instead there exists an explicit declarative expression of the reasons a given piece of code is executed. With these reasons enumerated, a system can explain how it reached a given failure and can retrieve an introspective explanation of such.

One of the greatest benefits of using introspective Meta-XPs is their ability to apply learning tasks that are appropriate to a given situation without having to blindly search all possible learning choices. Many current multi-strategy systems (e.g., Genest et al., 1990; Tecuci & Michalski, 1991) apply learning algorithms in a predefined order. If the first fails, then the next strategy is tried. Much effort may be wasted in worst-case scenarios.

Much work remains to be done with the failure type of Incorrect-World-Model, including the formulation of a representation for deciding when to use the heuristic search briefly mentioned in the paper. Other strategies remain to be created. The task of knowing when an assertion is incorrect, not just incomplete, is a difficult but interesting research problem.

Another effort under way is to represent failures in choosing the correct reasoning strategy to use in understanding. We propose to extend Meta-AQUA to learn control information by representing Meta-XPs that

point to potential problems with the reasoning choices made in each phase. The failure type Incorrect Reasoning Choice occurs when the reasoner has an appropriate knowledge structure to reason with and index to the structure in memory, but incorrectly chooses the wrong knowledge because the reasoning method it decided to use turned out to be inappropriate or inapplicable. An analysis of the choice of reasoning methods will result in learning control strategies designed to modify the heuristics used in this choice.

Most systems assume noise-free input. Those that deal with noise seldom analyze the source or causes of the noise. A robust story understander should be able to reason about the validity of input concepts, including the possibility of intentional deception by characters in a story. Thus an interesting extension of this research currently being pursued is combining story understanding with problem solving in the domain of detective investigations. Declarative process representations similar to that of story understanding are being developed. Parallel to story understanding sequences of: identify question fi generate hypothesis fi verify fi learn/review, problem-solving sequences would be represented as: identify problem fi

generate solution fi test fi learn/review. Meta-XPs would then be used to reason about and improve the problem-solving process of the reasoner.

Acknowledgements

This research was supported by the National Science Foundation under grant IRI-9009710. The authors thank Joel Martin for insightful comments and criticism, and Sue Farrell for proofing an earlier draft of the paper.

References

Alterman, R. A Dictionary Based on Concept Coherence, Artificial Intelligence, Vol. 25, pp. 153-186, 1985.
?
Birnbaum, L., and Collins, G. Opportunistic Planning and Freudian Slips, Proceedings of the Sixth Annual Conference of the Cognitive Science Society, Boulder, CO, 1984.

Carbonell, J.G. Derivational Analogy: A theory of reconstructive problem solving and expertise acquisition, in Machine Learning: An Artificial Intelligence Approach, R. Michalski, J. Carbonell, and T. Mitchell (Eds.), Morgan Kaufmann Publishers, San Mateo, CA., 1986.

DeJong, G., and Mooney, R. ExplanationBased Learning: An alternative view, Machine Learning, Vol. 1, No. 2, pp. 145- 176, 1986.

Doyle, J. A Truth Maintenance System, Artificial Intelligence, Vol. 12, pp. 231-272. 1979.

Genest, J., Matwin, S., and Plante, B. Explanation-Based Learning with Incomplete Theories: A three-step approach, in Proceedings of Seventh International Conference on Machine Learning, Austin, TX, (June), pp. 286-294, 1990.

Hammond, K.J. "Case-Based Planning: An integrated theory of planning, learning and Memory." Ph.D. thesis, Research Report 488, Yale University, New Haven, CT, (October), 1986.

Hammond, K.J. Opportunistic Memory: Storing and recalling suspended goals, in Proceedings of a Workshop on Case-Based Reasoning, Clearwater Beach, FL, (May), pp. 154-168, 1988.

Hayes-Roth, B., and Hayes-Roth F. A. Cognitive Model of Planning, Cognitive Science, Vol. 2, pp. 275-310, 1979.

Kass, A., Leake, D., and Owens, C. "SWALE: A program that explains," in Schank, 1986.

Kolodner, J.L., and Simpson, R.L. A Case for Case-Based Reasoning, in Proceedings of the Sixth Annual Conference of the Cognitive Science Society, Boulder, CO, 1984.

Leake, D. Anomaly Detection Strategies for Schema-Based Story Understanding, in Proceedings of the Eleventh Annual Conference of the Cognitive Science Society, Hillsdale, NJ: Lawrence Erlbaum Associates, pp. 490-497, 1989.

Mitchell, T., Keller, R., and Kedar-Cabelli, S. Explanation-Based Generalization: A unifying view, Machine Learning, Vol. 1, No. 1, 1986.

Ng, H., and Mooney, R. On the Role of Coherence in Abductive Explanation. in Proceedings of the 8th National Conference on Artificial Intelligence, pp. 337--342, Boston, MA. 1990.

Ram, A. "Question-driven Understanding: An integrated theory of story understanding, memory and learning," PhD. Thesis, Research Report 710, Yale University, New Haven, CT (May), 1989.

Ram, A. Decision Models: A theory of volitional explanation, in Proceedings of Twelfth Annual Conference of the Cognitive Science Society, Cambridge, MA, (July), pp. 198-205, 1990a.

Ram, A. Incremental Learning of Explanation Patterns and Their Indices, in Proceedings of Seventh International Conference on Machine Learning, Austin, TX, (June), pp. 313-320, 1990b.

Ram, A. A Theory of Questions and Question Asking, The Journal of the Learning Sciences, Vol. 1, No. 3 & 4, in press.,1991.

Schank, R. C. Dynamic Memory: A theory of reminding and learning in computers and people, Cambridge University Press, Cambridge, MA, 1982.

Schank, R. C. Explanation Patterns: Understanding mechanically and creatively, Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

Tecuci, G., and Michalski, R. A Method for Multistrategy Task-Adaptive Learning Based on Plausible Justifications, in Proceedings of the the Eighth International Conference on Machine Learning, Chicago, IL, July, 1991.

Thagard, P. Explanatory Coherence. Behavioral and Brain Sciences, Vol. 12, No 3, pp. 435--502, 1989.