Technical Report GIT-CC-93/47, College of Computing, Georgia Institute of Technology, Atlanta, GA, 1993. In R.C. Schank, A. Kass, & C.K. Riesbeck, editors, *Inside Case-Based Explanation*, chapter 7, pages 207–261, Lawrence Erlbaum Associates, 1994.

AQUA: Questions that drive the explanation process

Ashwin Ram
College of Computing
Georgia Institute of Technology

Editor’s Introduction

In the doctoral dissertation from which this chapter is drawn, Ashwin Ram presented an alternative perspective on the processes of story understanding, explanation, and learning. The issues that Ram explores in that dissertation are similar to those that are explored by the other authors in this book, but the angle that Ram takes on these issues is somewhat different. Ram's exploration of these processes is organized around the central theme of question asking. For Ram, understanding a story means identifying questions that the story raises, and questions that it answers. Question asking also serves as a lens through which each of the sub-processes is viewed: the retrieval of stored explanations, for instance, is driven by a library of what Ram calls “XP retrieval questions”; likewise, evaluation is driven by another set of questions, called “hypothesis verification questions.”

To some extent, any program that builds explanations can be thought of as asking questions. In fact, any program that solves a problem and sets up goals and sub-goals can be described as asking itself how a particular goal can be achieved. Ram’s contribution is to make as many of questions as possible explicit. By doing so, he helps his readers separate implementation details from important issues; It's the questions that the program asks which is important, not the details of how they get asked.

The AQUA program, which is Ram's implementation of this question-based theory of understanding, is a very complex system, probably the most complex among the programs described in this book. AQUA covers a great deal of ground; it implements the entire case-based explanation process in a *question-based manner. This breadth poses a problem for this volume, since it isn't possible to cover all that ground in this chapter. We have had to ignore many aspects of the program. We have focussed on the high-level description of the questions the program asks, especially the questions it asks when constructing and evaluating explanations of volitional actions.*

Introduction: Question-driven understanding

Story understanding is a goal-directed process. The memory of an understanding system is never quite complete: knowledge structures may be missing, they may have “gaps” in them, or they may not be indexed correctly in memory. When one reads a story, these gaps give rise to questions about the input. The point of reading is to find answers to these questions, and to learn by filling in the gaps in one's world model. Questions represent the “knowledge goals” of the understander, things that the understander wants to learn about.

Most teachers have had the experience of thinking that their students understood some material because they were asking the “right questions.” Children ask questions constantly in an attempt to understand and learn about the world around them. Even as adults, we express our curiosity in the form of questions, often to ourselves, as we wonder about novel situations, explore new hypotheses, and become interested in various issues. The ability to ask questions, it seems, is central to the processes of reasoning and learning.

This chapter presents a question-based theory of explanation, story understanding, and learning. Our main contribution to Schank's theory of explanation patterns is that the case-based explanation process in our model, while similar to that used by the SWALE program [Kass, Leake, and Owens, 1986], is formulated in a question-based framework. Our emphasis is on the questions that underlie the creation, verification, and learning of explanations, and is complementary to the creative adaptation process modelled in SWALE. Furthermore, we focus on the use of possibly incomplete explanation patterns with questions attached to them, and the learning that occurs as these questions are answered. Finally, we propose a content theory of volitional explanations that is used for motivational analysis in story understanding.

We will discuss a computer program, called AQUA (Asking Questions and Understanding Answers), that is based on our model of question-driven understanding and learning. The main point of the research is to create a model of a dynamic understander that is driven by its questions or goals to acquire knowledge. Rather than being "canned," the understander is always changing as its questions change. Such an understander reads similar stories differently and forms different interpretations as its questions and interests evolve. The intent is not to design a system that can acquire the “right” understanding of a topic, or form the “right” explanation for a story, but one that is able to wonder and to ask questions about the unusual aspects of its input. As it learns more about the domain, the system asks better and more detailed questions, and creates better and more detailed explanations. This kind of questioning forms the origins of creativity; rather than being satisfied with available explanations, a creative person asks questions and explores the explanations in novel ways.

Question generation is the process of identifying what the reasoner needs to explain and learn about. Learning occurs incrementally as the reasoner's questions are answered through the creation and evaluation of explanations. Although the computer model is being used to explore cognitive issues such as the ones previously mentioned, there are also practical benefits of a system that can represent and reason explicitly about its own goals. Such a system can focus its limited resources on relevant aspects of its environment while paying less attention to irrelevant ones. This allows it to spend more time drawing inferences that are relevant and useful to its goals. This is important in reasoning situations in which the reasoner might draw a combinatorially large set of inferences, in learning situations in which it is impractical to focus attention on every aspect of a situation and remember every novel aspect, and in explanation situations in which it is difficult to evaluate the utility of a candidate explanation without reasoning about the goals that the explanation might support. The reasoning system needs a principled way to determine which inferences are worth drawing, which concepts are worth learning, or which explanations are worth pursuing. In order to ensure that the system does not spend its limited resources trying to infer everything it can, its knowledge goals are used to focus the inferencing, learning and explanation process on information that is useful to the goals of the system.

The role of questions in explanation and learning

The basic assumption of our theory is that asking questions is central to understanding. To illustrate what this means, consider the following story (New York Times, April 14, 1985):

> **S-1: Boy Says Lebanese Recruited Him as Car Bomber.**
>
> JERUSALEM, April 13 -– A 16-year-old Lebanese was captured by Israeli troops hours before he was supposed to get into an explosive-laden car and go on a suicide bombing mission to blow up the Israeli Army headquarters in Lebanon. ... What seems most striking about [Mohammed] Burro's account is that although he is a Shiite Moslem, he comes from a secular family background. He spent his free time not in prayer, he said, but riding his motorcycle and playing pinball. According to his account, he was not a fanatic who wanted to kill himself in the cause of Islam or anti-Zionism, but was recruited for the suicide mission through another means: blackmail. [p. A1]

If one wants to explain this story and learn more about the motivations of the terrorists in the Middle East, this story is interesting because it is anomalous. The usual stereotype of the Shiite religious fanatic does not hold here. Instead, this story raises many new questions. Some of the questions that were voiced by a class of graduate students when this story was read to them were:

1. Why would someone commit suicide if he was not depressed?
2. Did the kid think he was going to die?

3. Are car bombers motivated like the Kamikaze?
4. Does pinball lead to terrorism?
5. Who blackmailed him?
6. What fate worse than death did they threaten him with?
7. Why are kids chosen for these missions?
8. Why do we hear about Lebanese car bombers and not about Israeli car bombers?
9. Why are they all named Mohammed?
10. How did the Israelis know where to make the raids?
11. How do Lebanese teenagers compare with American teenagers?

Some of these questions seem reasonable, (*e.g.*, "Did the kid think he was going to die?"), but some are rather silly in retrospect (*e.g.*, "Does pinball lead to terrorism?"). Some, though perfectly reasonable, are not central to the story but relate to other issues that a given student was reminded of, was wondering about, or was interested in (*e.g.*, "Why do we hear about Lebanese car bombers and not about Israeli car bombers?").

The claim is that an understander has questions already extant in memory before it begins to read a story. These questions are left over from the understander's previous experiences. As the understander reads the story, it remembers these questions and thinks about them again in a new light. This raises further questions for the understander to think about. Many of these questions seek explanations, which are knowledge structures that allow the understander to answer its questions based on a causal understanding of the situation (*e.g.*, "Kids are chosen because they are more gullible"). Explanations, in turn, can give rise to further questions (*e.g.*, "Are Lebanese teenagers more gullible than American teenagers?").

Ultimately, the understander is left with several new questions that it may or may not have asked before. Certainly, after reading the blackmail story, one expects to have several questions representing issues one was wondering about that were not resolved by the story. For example, in this story, it turns out that the boy was blackmailed into going on the bombing mission by a terrorist group that was threatening his parents. This makes one think about the question "What are family relations like in Lebanon?" This question remains in memory after reading the story. To the extent that one is interested in this question, one will read stories about the social life in Lebanon, and one will relate other stories to this one. To cite another example, one of the students in the class repeatedly related the story to his readings on the IRA because he was interested in similar issues about Ireland.

Understanding is a process of relating what one reads to the questions that one already has. These questions represent the knowledge goals of the understander, *i.e.*, the things that the understander wants to learn [Dehn, 1989; Leake and Ram, 1993; Hunter, 1989; Ram, 1989, 1991, 1993; Ram and Hunter, 1992; Schank and Ram, 1988]. The purpose of building explanations is to find answers to these questions and, thus, to arrive at a more complete understanding of the issues one is interested in. However, while doing this, many new questions are often raised. These questions are stored in memory and, in turn, guide the understanding of future stories and affect the interpretations that are drawn. This process is shown in figure 1.

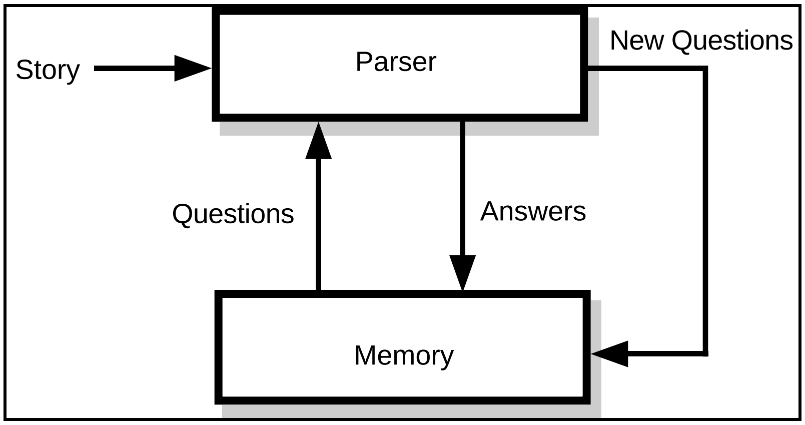

Figure 1: Question-driven understanding.

In contrast to the traditional view of understanding as a “story in, representation out” process, we view understanding as a “questions + story in, answered questions + new questions out” process. A theory of story understanding, therefore, must include a theory of memory, explanation, and learning in addition to a theory of parsing.

Although this type of reasoning may not be conscious, explanation and learning are motivated by a reasoner's goals and interests. When the reasoner encounters difficulties during understanding, planning, or any other task, it remembers the nature of these difficulties and learns in order to perform its tasks better in the future. The knowledge goals of the reasoner, which arise from these very difficulties, are used to focus the reasoning and learning process. Our model is very different from other approaches that rely on properties of the domain to determine what needs to be learned because it relies on the goals of the reasoner. For example, one might propose a rule, similar to that discussed by DeJong [1983], that the understander generalize a new schema whenever it reads a story in which a preservation goal (P-GOAL) is violated in a novel manner. But this should be so only if noticing violations of this P-GOAL is actually useful to the program. Any such rule must make a statement about the goals of the program, not just about the content of the domain. A similar argument can be made for the use of knowledge goals, or questions, to focus inference generation for understanding, explanation, or diagnosis [Ram, 1990c, 1991, 1993; Ram and Cox, 1993; Ram and Hunter, 1992; Ram and Leake, 1991]. A goal-based model of explanation and learning is a plausible account of human behavior, and also has computational advantages for the design of learning programs.

Our theory of questions is based on a theory of understanding tasks. In addition to parser-level tasks such as noun group connection, pronoun reference, and so on, these tasks include higher-level tasks such as the integration of facts with what the understander already knows, the detection of anomalies in the text that identify deficiencies in the understander's model of the domain, the formulation of explanations to resolve those anomalies, the confirmation and refutation of potential explanations, and the learning of new explanations for use in understanding future situations. These are the basic tasks that an understander needs to be able to perform.

In order to carry out these tasks, the understander needs to integrate the text, which is often ambiguous, elliptic and vague, with its world knowledge, which is often incomplete. In formulating an explanation, for example, the understander may need to know more about the situation than is explicitly stated. However, it is impossible to anticipate when a particular piece of knowledge will be available to the understander because the real world (in the case of a story understanding program, the story) will not always provide exactly that piece of knowledge at exactly the time that the understander requires it. Thus, the understander must be able to suspend questions in memory and to reactivate them just when the information needed becomes available. In other words, the understander must be able to remember what knowledge is needed and why.

Furthermore, the system's understanding of any real world domain can never be quite complete. Conventional script-, frame- or schema-based theories assume that understanding means finding an appropriate script, frame or schema in memory and fitting it to the story. Schemas in memory are assumed to be “correct” in the sense that they are completely understood and constitute a correct model of the domain. If an applicable schema is found, an instance of the schema is created and applied to the story. The story is then assumed to be “understood.” However, this model is inadequate because an understander's memory is always incomplete, especially in poorly understood domains. Some knowledge structures may be missing; others may have gaps in them. These gaps correspond to what the understander has not yet understood about the domain. Even if a schema appears to be correct, novel experiences or stories may reveal flaws in the schema or a mismatch with the real world. Furthermore, the schema may not be indexed correctly in memory.

Understanding tasks, therefore, generate information subgoals or questions, which represent what the understander needs to know to perform the current task, be it explanation, learning, or any other cognitive task. These questions constitute the specific knowledge goals of the understander. Learning is a process of seeking answers to these questions in the input, which in turn raises new questions while answering old ones.

For example, in order to understand the blackmail story S-1, the system must understand the motivations of the would-be suicide bomber (an explanation task). In other words, it must formulate the question “Why would the boy have done the suicide bombing?” The desired explanation for the suicide bomber's actions constitutes an answer to this question. The explanation task gives rise to further questions and subquestions: “Did the boy think he was going to die?” “Was the boy a religious fanatic?” Ultimately, the system finds an answer to a question in the input story, which enables it to complete its explanation task.

Types of questions

A functional theory of questions must be based on a taxonomy of types of knowledge goals that arise from the underlying understanding tasks that the questions serve. To develop a taxonomy of these knowledge goals, we asked several subjects to voice the questions that occurred to them as stories were read out to them. We then analyzed those questions and grouped them according to the understanding task (*e.g.*, hypothesis verification) that they were relevant to. The groupings were revised based on a functional analysis of the knowledge required for the subtasks in the computational theory of story understanding and explanation, the subtasks, in turn, being mutually refined based on our analysis of the question data.

It is interesting to note that, although our main taxonomic criteria were functional, the taxonomy fits the data well. Thus, we hypothesize that the theory, although intended as a computational model of an active reader, is also a plausible cognitive model. This is supported by the fact that the model is consistent with psychological data on question asking (*e.g.,* [Graesser, Person, and Huber, 1992; Scardamalia and Bereiter, 1991]). The goal-based approach is also consistent with psychological data on goal orientation in learning (*e.g.*, [Barsalou, 1991; Ng and Bereiter, 1991; Steinbart, 1992; Wisniewski and Medin, 1991]; see also [Leake and Ram, 1993]) and in focus of attention and inferencing (*e.g.,* review by Zukier [1986]).

We propose the following taxonomy of knowledge goals for story understanding:

Text goals: Knowledge goals of a text analysis program, arising from text-level tasks. These are the questions that arise from basic syntactic and semantic analysis that needs to be done on the input text, such as noun group attachment or pronoun reference. An example text goal is to find the referent of a pronoun.

Memory goals: Knowledge goals of a dynamic memory program, arising from memory-level tasks. A dynamic memory must be able to notice similarities, match incoming concepts to stereotypes in memory, form generalizations, and so on. An example memory goal might be to look for an event predicted by stored knowledge of a stereotyped action, such as asking what the ransom will be when one hears about a kidnapping.

Explanation goals: Goals of an explainer that arise from explanation-level tasks, including the detection and resolution of anomalies, and the building of motivational and causal explanations for the events in the story in order to understand why the characters acted as they did or why certain events did or did not occur. An example explanation goal might be to figure out the motivation of a suicide truck bomber mentioned in a story.

Relevance goals: Goals of any intelligent system in the real world, concerning the identification of aspects of the current situation that are “interesting” or relevant to its general goals. An example is looking for the name of an airline in a hijacking story if the understander were contemplating travelling by air soon.

Each question focuses on a different aspect of a story. For example, explanation questions focus on different types of anomalies, and on explanations for these anomalies. Asking an anomaly detection question is essential to detecting the corresponding anomaly. For example, asking the question “Does the actor want the outcome of this action?” is essential to the detection of a goal violation anomaly in the sense that the program will not notice the anomaly if it does not focus on the goals of the agent, that is, if it does not think of asking the question.

To put this another way, the questions asked by the understander influence its final understanding. Thus it is important for the understander to ask the “right” questions in order to achieve a detailed understanding of the situation. For the purpose of understanding stories involving motivations of people, we have developed a taxonomy of motivational questions that focus on those motivational aspects of stories that are needed to understand such stories.

In addition to their theoretical role in the model of story understanding, knowledge goals have also played an implementational role in the research by providing a uniform mechanism for the integration of various cognitive processes. For example, knowledge goals arising from, say, memory tasks are indexed in memory and used in the same way as knowledge goals arising from explanation tasks. A knowledge goal generated from one task may be suspended and satisfied opportunistically during the pursuit of some other task at a later stage or even during the processing of a different story. Implementational details of AQUA's opportunistic memory architecture may be found in Ram [1989].

In the remainder of this chapter, we will focus on questions arising from explanation goals, which are the basis for AQUA's explanation construction, verification, and learning methods.

AQUA’s explanation patterns

When a new story or situation is processed, it is understood in terms of knowledge structures already in memory. As long as these structures provide expectations that allow the reasoner to function effectively in the new situation, there is no problem. However, if these expectations fail, the reasoner is faced with an anomaly. The world is different from its expectations. In order to learn from this experience and to continue processing the story, the reasoner must be able to explain what it does not understand. To do this, it needs to know why it had those expectations in the first place. It also needs to explain why the failure occurred, that is, to identify the knowledge structures that gave rise to the faulty expectations, and to understand why its domain model was violated in this situation. Finally, it must update its knowledge structures and store the new experience in memory for future use. Explanation is a central aspect of this process of understanding and learning.

The construction of explanations is also known as abduction, or inference to the best explanation. In AQUA, this is carried out through a case-based reasoning process in which previous explanation structures, represented as explanation patterns, are retrieved and applied to the anomaly at hand. This allows AQUA to understand the situation as well as to understand its own failure to model the situation correctly.

Explanation patterns (XPs) in AQUA have four components:

- **PRE-XP-NODES**: Nodes that represent what is known before the XP is applied. One of these nodes, the EXPLAINS node, represents the particular action being explained.
- **XP-ASSERTED-NODES**: Nodes asserted by the XP as the explanation for the EXPLAINS node. These comprise the premises of the explanation.
- **INTERNAL-XP-NODES**: Internal nodes asserted by the XP in order to link the XP-ASSERTED-NODES to the EXPLAINS node.
- **LINKS**: Causal links asserted by the XP. These taken together with the INTERNAL-XP-NODES are also called the internals of the XP.

An explanation pattern is a directed, acyclic graph of conceptual nodes connected with causal LINKS, which in turn could invoke further XPs at the next level of detail. The PRE-XP-NODES are the sink nodes (consequences) of the graph, and the XP-ASSERTED-NODES are the source nodes (antecedents or premises). The difference between XP-ASSERTED-NODES and INTERNAL-XP-NODES is that the former are merely asserted by the XP without further explanation, whereas the latter have causal antecedents within the XP itself. An XP applies when the EXPLAINS node matches the concept being explained and the PRE-XP-NODES are in the current set of beliefs. The resulting hypothesis is confirmed when all the XP-ASSERTED-NODES are verified.

Ultimately, the graph structure underlying an XP bottoms out in primitive inference rules of the type used by MARGIE [Rieger, 1975] or PAM [Wilensky, 1978]. Schank [1986] describes XPs as the “scripts” of the explanation domain. Unlike scripts, however, XPs are flexible in the sense that their internal structure allows them to be useful in novel situations, while retaining the advantages of pre-stored structures in stereotypical situations. Access to an XP's causal internals is essential to the incremental question-based learning process in AQUA.

Explanation Types

Explanations can be divided into two broad categories, physical and volitional:

Physical explanations

Physical explanations link events with the states that result from them, and further events that they enable, using causal chains similar to those of Rieger [1975] and Schank and Abelson [1977]. Physical explanations answer questions about the physical causality of the domain. For example, if the system had never read a story about a car bombing before, it might encounter an anomaly: “How can a car be used to blow up a building?” The answer to this question is a physical explanation:

1. A car is a physical object.
2. A car can contain explosives.

3. A car can be propelled by driving it.
4. Explosives can be blown up by the sudden impact of a car colliding with a building.
5. A building can be blown up by blowing up explosives in its immediate vicinity.

Thus the explanation is that the bomber drove an explosive- laden car into the building, the impact caused the explosives to detonate, which caused the building to blow up.

Volitional explanations

The particular content of the causal knowledge represented in explanation patterns depends, of course, on the domain of interest. AQUA deals with volitional explanations, which link actions that people perform to their goals and beliefs, yielding an understanding of the motivations of the characters. For example, consider the following story:

> **S-2: Suicide bomber strikes Israeli post in Lebanon.**
>
> SIDON, Lebanon, November 26 -– A teenage girl exploded a car bomb at a joint post of Israeli troops and pro-Israeli militiamen in southern Lebanon today, killing herself and causing a number of casualties, Lebanese security sources said. ... A statement by the pro-Syrian Arab Baath Part named the bomber as Hamida Mustafa al-Taher, born in Syria in 1968. The statement said she had detonated a car rigged with 660 points of explosives in a military base for 50 South Lebanon Army men and Israeli intelligence and their vehicles.

in the suicide bombing story S-2, the understander needs to explain why the girl performed an action that led to her own death. An explanation for this anomaly, such as the religious fanatic explanation, must provide a motivational analysis of the reasons for committing suicide.

AQUA has two broad categories of explanatory knowledge:

1. **Abstract explanation schemas** for why people do things. These are standard high-level explanations for actions, such as “Actor does action because the outcome of action satisfies a goal of the actor.”
2. **Explanatory cases**. These are specific explanations for particular situations, such as “Shiite Moslem religious fanatic goes on suicide bombing mission.”

For example, an explanation of type 1 for story S-2 might be “Because she wanted to destroy the Israeli base more than she wanted to stay alive.” An explanation of type 2 would be simply “Because she was a religious fanatic.” The internal causal structure of the latter explanation could then be elaborated to provide a detailed motivational analysis in terms of explanations of the first type if necessary.

Both types of explanatory knowledge are represented using volitional XPs with the internal structure discussed earlier. Volitional XPs relate the actions in which the characters in a story are involved to the outcomes that those actions had for them, the goals, beliefs, emotional states and social states of the characters as well as priorities or orderings among the goals, and the decision process that the characters go through in considering their goals, goal-orderings and likely outcomes of the actions before deciding whether to perform those actions. A volitional explanation involving the planning decisions of a character is called a decision model [Ram, 1990a]. The decision model has the following components:

The outcome of an action: Every action results in some set of states that may or may not be beneficial to the people involved in that action, depending on their goals at that time. The outcome of an action, therefore, must be modelled from the point of view of a particular volitional agent involved in that action (see also Carbonell [1979]). The most common volitional participants are actor and planner, but any role involving a volitional agent must potentially be explained.

The decision process: Every agent involved in an action makes a decision about whether to participate in that particular volitional role (actor, planner, object, etc.) in the action. Such decisions represent the planning process that the agent underwent prior to the action. A complete model of this process requires a sophisticated vocabulary of goals, goal interactions, and plans, such as that of Wilensky [1983] or Hammond [1986]. There are three basic kinds of decisions:

1. **Choice**: The agent chooses to participate or not to participate in a given volitional role in some action. The explanation must describe why he made this choice.
2. **Agency**: The agent is induced to participate or not to participate in a given volitional role in an action. This is similar to the previous case in that the agent “enters” the action of his own volition. The difference is that here the agent is acting under the agency of another agent. Thus the reasoner must be able to model inter-agent interactions [Domeshek, 1992; Ram, 1984; Schank and Abelson, 1977; Wilensky, 1983].
3. **Coercion**: The agent is forced to participate or not to participate in a given volitional role in an action. This case arises when an agent is physically coerced into participation or non-participation.

Considerations in decisions: The system also needs to reason about what an agent was considering as he made a particular decision. Considerations model the goals and beliefs of an agent, along with orderings among these goals and expected outcome of the action being considered. Considerations are composed of three constituents: (1) goals considered by the agent while deciding whether or not to participate in an action, (2) goal-orderings, the agent's prioritization of these goals, and (3) the expected-outcome: the agent's beliefs about what the outcome of the action is likely to be. If the actual outcomes do not match with the agent's expectations and goals, the system can use these representations to reason about the failure of the agent's plans (see also Domeshek [1992]; Jones [1992]; Owens [1991]).

An example of an explanation pattern is shown in figure 2.

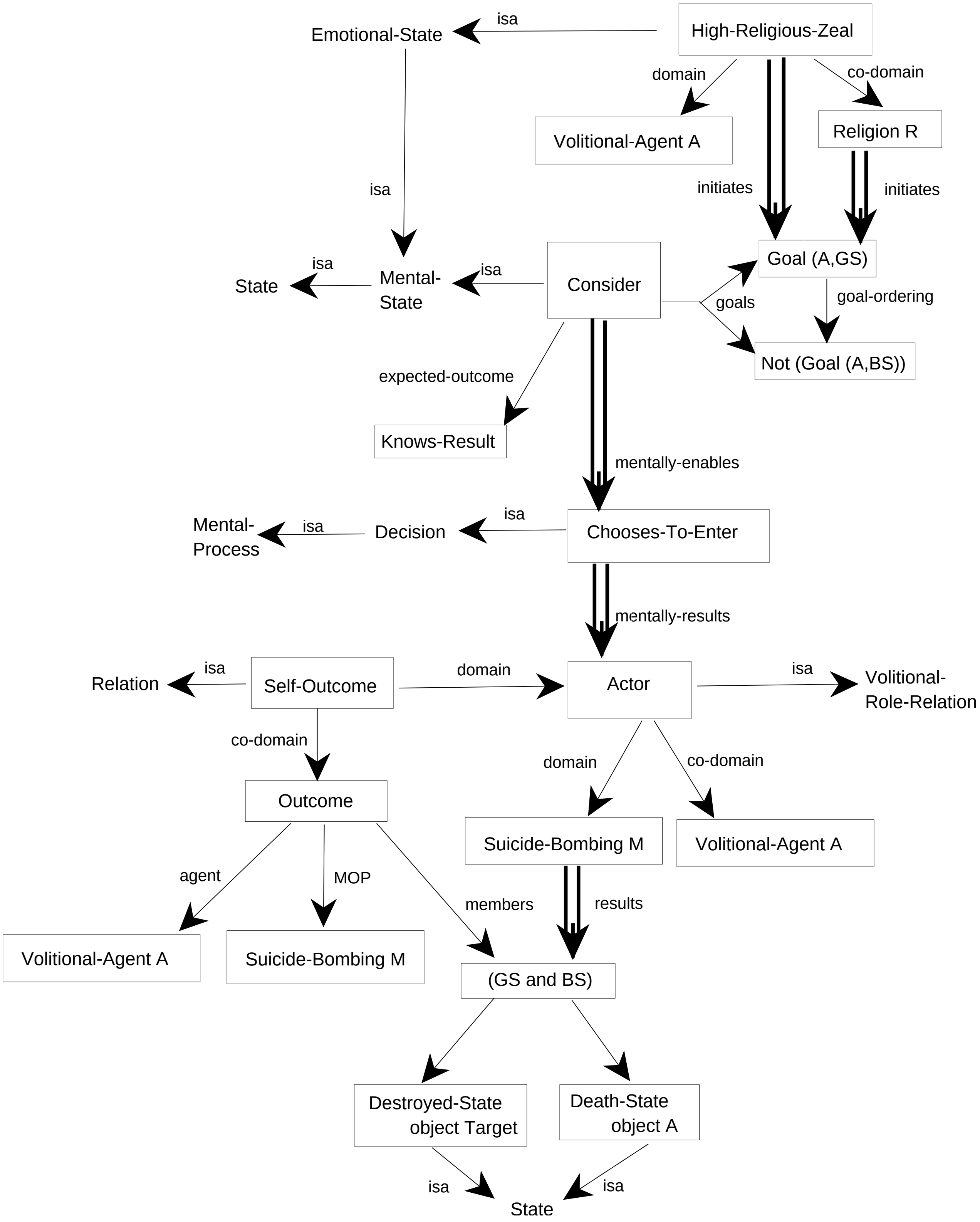

Figure 2: Representation of XP-RELIGIOUS-FANATIC.

An explicit representation of the planning and decision-making process of the characters in the story allows AQUA to explain their actions in terms of their goals and beliefs. Each of these constituents may itself need to be explained further. For example, the system might question the social or mental (*e.g.*, emotional) states that initiated a particular goal or goal-ordering in an agent, or how a particular belief about the outcome of an action came about. Explanations, therefore, may need to be elaborated according to the demands of the story and the goals of the system.

A question-based model of explanation

We now turn to the process implications of the model for the tasks of text interpretation and explanation in the context of story understanding and learning, and discuss the implementation of the model in the AQUA program. Methodologically, we used AQUA as a testbed for exploring issues of interpretation, learning, explanation, and interestingness in an integrated framework.

AQUA's basic goal in reading is to answer its questions and to improve its understanding of the domain (terrorism). AQUA's output consists of answers to old questions about the domain plus, of course, new questions. AQUA is driven by its questions or knowledge goals. It is a dynamic program, and it reads similar stories differently and forms different interpretations as its questions and interests evolve. AQUA would reread a story differently from the way it first read the story because the questions and explanations generated during the first reading affect the questions raises on the second reading. Further details of the question-driven understanding process may be found in Ram [1989, 1991].

Because questions represent the knowledge goals of the understander, they also provide the focus for learning. In addition to asking questions, therefore, AQUA can learn from answers to these questions. AQUA improves its explanatory knowledge of its domain by incremental refinement of this knowledge using answers to questions that arise from the explanation process [Ram, 1993]. AQUA retrieves past explanations from situations already represented in memory, and uses them to build explanations to understand novel stories about terrorism. In doing so, the system refines its understanding of the domain by filling in gaps in these explanations, by elaborating the explanations, by learning new indices for the explanations, or by specializing abstract explanations to form new explanations for specific situations. This is a type of incremental learning since the system improves its explanatory knowledge of the domain in an incremental fashion rather than by learning complete new explanations from scratch.

The basic process of goal-based understanding involves the generation of knowledge goals seeking information required by various understanding tasks, the transformation of these knowledge goals into subgoals, and the matching of pending knowledge goals to information in the story. One might think of this as a process of question transformation, in which a reasoner generates questions which then trigger a parsing process which can in turn generate more questions.

Each of AQUA's knowledge goals type is expressed as a question that focuses on a different aspect of the story. For example, explanation questions focus on different types of anomalies, and on explanations for these anomalies. Asking an anomaly detection question is essential to detecting the corresponding anomaly. Similarly, asking a hypothesis verification question is essential to verifying a proposed explanation for an anomaly, and to the learning that results. For the purpose of understanding stories involving motivations of people, we have developed a taxonomy of motivational questions that focus on those motivational aspects of stories that are needed to build volitional explanations based on the planning/decision model discussed earlier. This taxonomy is part of a larger taxonomy of questions based on the understanding tasks that AQUA needs to perform when it reads a story. Let us consider the explanation-related tasks in more detail.

The AQUA Processing Cycle

The performance task in AQUA is to “understand” human interest stories, that is, to construct explanations of the actions observed in the story that causally relate the actions to the goals, plans and beliefs of the actors and planners of the actions. Such an explanation is called a volitional explanation, and the process of constructing these explanations is called motivational analysis. In general, an explanation consists of several inference rules connected together into a graph structure with several antecedents and one or more consequents, as discussed earlier. Construction of such explanations is typically done by chaining together inference rules through a search process (*e.g.*, [Rieger, 1975; Wilensky, 1981; Morris and O'Rorke, 1990]), through a weighted or cost-based search (*e.g.*, [Hobbs, Stickel, Appelt, and Martin, 1990; Stickel, 1990]), or through a case-based reasoning process in which previous explanations for similar situations are retrieved and adapted for the current situation (*e.g.*, [Schank, 1986; Kass, Leake, and Owens, 1986; Ram, 1989; Ram, 1990a]). The latter method, which is the basis for AQUA's approach to motivational analysis, is similar to the use of explanation schemas to build explanations (*e.g.*, [Mooney and DeJong, 1985]) since it relies on the instantiation of “large” knowledge structures (cases or schemas) rather than the chaining together of “small” knowledge structures (inference rules). Rather than defend the case-based reasoning approach here, we will simply state the assumptions implicit in this approach [Ram, 1993]:

Efficiency assumption: It is more efficient to retrieve and apply larger knowledge structures (here, XPs) than to construct them from scratch each time out of smaller knowledge structures or inference rules.

Content assumption: There are too many possible ways in which inference rules can be connected together, many of which will be irrelevant or meaningless. The content of the explanations produced through case-based reasoning is likely to be better than those produced through exhaustive search through inference rules, because cases (here, XPs) contain experiential knowledge about the ways in which the rules are actually connected together in real situations.

Typicality assumption: Situations encountered in the real world are typical of the kinds of situations that are likely to be encountered in the future because the world is reasonably stable and regular. Thus it is worthwhile creating a new case (here, XP) to represent novel experiences because remembering this case will make it easier to process similar situations in the future.

The processing cycle in AQUA has three interacting steps: **Read**, **Explain** and **Learn**. The interaction between these steps is managed through a question-based agenda system, in which tasks are **suspend**ed if there is insufficient information to run them, and **restart**ed when the questions seeking the missing information are answered:

- **Read** the story.

  Leaving aside the natural language aspects of the task, this is equivalent to processing a sequence of input facts representing the individual events in the story.
- **Explain** each action in the story.

  Build hypothesis trees representing possible explanations for the motivations of the actor, planner, and any other volitional agents involved in the action.
- **Suspend** the explanation task until one of the hypotheses in a hypothesis tree is confirmed.
- **Restart** the suspended task when this happens.

  Confirm or refute associated hypotheses, as appropriate.
- **Learn** when a hypothesis is confirmed.

The READ step

In this step, AQUA reads a piece of text, guided by the questions in memory. It tries to answer these questions using the new piece of information.

Read some text, focussing attention on interesting input as determined below. Build minimal representations in memory.

Retrieve extant knowledge goals or questions indexed in memory that might be relevant, *i.e.*, whose concept specifications are satisfied by the new input. Use these questions as an interestingness measure to focus the read above.

Answer the questions retrieved in the previous step. Unify the answer with each question, and restart the suspended process represented by the task specification. *E.g.*, if the question is in service of hypothesis verification:

> **Answer question** by either confirming or refuting it.
>
> **Propagate** back to the hypothesis that the question originated from.
>
> **Confirm/refute hypotheses**. If the verification questions of a hypothesis are confirmed, confirm the hypothesis and refute its competitors. If any verification question of a hypothesis is refuted, refute the corresponding hypothesis.

Explain the new input if necessary, *i.e.*, if interesting and not already explained.

The EXPLAIN step

The **Explain** step implements the basic explanation cycle in AQUA. The outline of this step is as follows. Further details of the explanation process are discussed in the section below.

Detect anomalies in input by asking anomaly detection questions

Formulate XP retrieval questions

Retrieve XPs that might help explain the anomaly

Apply XP to input:

- If in applying the XP an anomaly is detected, characterize the anomaly and explain it recursively.
- If the XP is applicable to the input:
  - **Construct hypothesis** by instantiating the XP.
  - **Construct verification questions** to help verify or refute the new hypothesis.
  - **Index questions** in memory to allow them to be found in the next step.

Answer questions by reading further, focussing attention on input concepts that trigger questions in memory.

Confirm/refute hypotheses when their verification questions are answered, as appropriate.

The LEARN step

Since questions represent the knowledge goals of the understander, they provide the focus for learning. AQUA can:

Generalize novel answers to its questions.

Index these answers in memory, so that the task that originally generated the question would now find the information instead of failing.

As currently implemented, AQUA's memory consists of about 700 concepts represented as frames, including about 20 abstract XPs, 10 stereotypical XPs, 50 MOPs (most of which deal with the kinds of actions encountered in suicide bombing stories), 250 relations (including causal and volitional relations), and 20 interestingness heuristics (most of which are represented procedurally). The range of stories that AQUA can handle is limited only by the XPs in memory. We have focussed mostly on the domain of newspaper stories about suicide bombing, such as stories about religious fanatics, depressed teenagers, Kamikazes, and so on, although it would be straightforward to extend the program to other domains. This chapter focusses mainly on the explain step, and on the corresponding knowledge goals for the task of learning from explanations and explanation failures.

A closer look at the EXPLAIN step

The **Explain** step in the above understanding cycle implements the case-based explanation algorithm summarized below:

Input: *R*, a volitional-role-relation (actor, planner) between an action or MOP *M* and a volitional-agent *A*. By definition, *A* appears in the *R* slot of *M*.

Output: *H*, a hypothesis tree.

Algorithm:

- Invoke anomaly detection algorithm to determine whether *R* is anomalous.
- If so, create a root node for *T* and place the anomaly *a* at the root.
- Identify the set of anomaly category indices $\{I_a\}$ based on the anomaly *a*.

- Determine the set of situation indices $\{I_S\}$ by retrieving abstractions of *M*.
- Determine the set of character stereotype indices $\{I_C\}$ by matching *A* to known stereotypes.
- For each $\{I_a, J_S, I_C\}$ combination, retrieve any *XP* that is indexed by this combination (explanation pattern retrieval). This provides a set of potentially applicable explanation patterns {*XP*}.
- For each *XP* in this set {*XP*}, match the EXPLAINS node of *XP* to *R*. Retain the set of explanation patterns {*XP*} for which this match succeeds.
- For each *XP* in this new set {*XP*}, create hypotheses *H* as follows (explanation pattern application):
    - instantiate *XP*
    - unify EXPLAINS node of *XP* with *R*
    - instantiate INTERNAL-XP-NODES and LINKS of *XP*
    - instantiate pending questions attached to *XP*, if any
    - create a new node in *T* to represent the hypothesis *H* and attach it as a child of the root node representing the anomaly *a*
- For each *H* in the set of hypotheses, verify *H* as follows (hypothesis verification):
    - instantiate the XP-ASSERTED-NODES *n* of the *XP* that was instantiated to form *H*
    - create a hypothesis verification question *HVQ* from each *n* that is not already known to be true in the story
    - create a new node in *T* for each *HVQ* of *H* and attach it as a child of the node representing *H*
    - invoke hypothesis evaluation algorithm to determine current best hypothesis
- When all the *HVQ*s of any hypothesis *H* are verified (question answering), verify the hypothesis *H* and refute its competitors. Note that questions may be answered later while processing this or other stories.

The input to the algorithm is a volitional-role-relation, which is defined as a relation between an action or MOP (the domain of the relation) and a volitional-agent (the co-domain of the relation). The relation represents the fact that the agent is the actor or planner of an action (the two types of facts which require motivational analysis). The output is a hypothesis tree, a structure that represents the multiple possible explanations for the relation. The algorithm has four basic parts: anomaly detection, XP retrieval, XP application, and hypothesis verification. Let us examine these parts in more detail.

Anomaly detection

Anomaly detection refers to the process of identifying an unusual fact or situation description that needs explanation. The fact may be unusual in the sense that it violates or contradicts some piece of information in memory. Alternatively, the fact may be unusual because, while there is no explicit contradiction, the reasoner fails to integrate the fact satisfactorily in its memory. Anomaly detection in AQUA is done through a series of anomaly detection questions based on the goals, goal-orderings, plans, beliefs and decisions represented in AQUA's decision models [Ram, 1991; see also Leake, 1989a]. For example, the question “Did the actor want the outcome of his action?”

allows AQUA to notice a goal-violation anomaly in which an agent performs an action which violates the agent's own goals. AQUA's taxonomy of anomaly detection questions is discussed later.

Each type of anomaly is associated with abstract explanation schemas that form the anomaly category indices for the XPs representing specific explanatory cases. For example, the anomaly goal-violation is associated with the abstract XPs xp-not-know-outcome and xp-goal-sacrifice. This is used in the XP retrieval step discussed below.

In addition to the reasoning about the actor of an action, AQUA also considers the planner's reasons for planning that action, if the actor and the planner are different. This involves building a similar decision model from the planner's point of view. AQUA builds separate explanations for the actor and planner of every event if they are different. If the actor and planner are the same person, group, or institution, the default explanation for the planner is simply that the planner planned the action because the planner wanted to carry it out successfully.

Explanation pattern retrieval

When faced with an anomalous situation, AQUA tries to retrieve one or more previously known explanatory cases or, if no cases are available, abstract explanation schemas that would explain the situation. An applicable XP is one whose PRE-XP-NODES can be unified with the current situation, with the EXPLAINS node being unified with the particular action being explained. Since it is computationally infeasible to match the PRE-XP-NODES of every XP with every action being explained, AQUA uses a set of indices as a heuristic to identify potentially relevant explanatory cases. Learning the right indices for an XP is therefore an important component of AQUA's learning process.

In general, XPs are indexed by stereotypical descriptions of their EXPLAINS nodes, and a description of the anomaly to be explained. For example, in order to explain an action performed by a volitional agent, AQUA uses three types of indices to retrieve potentially relevant XPs: (1) the anomaly category index, which identifies classes of XPs relevant to the given anomaly, (2) the situation index, which identifies XPs relevant to a particular situation (action or MOP), and (3) the character stereotype index, which identifies XPs relevant to a particular stereotype that the agent can be viewed as. XP retrieval is done through the generation of XP retrieval questions, which are described in more detail later.

Explanation pattern application

Once a set of potentially applicable XPs is retrieved, AQUA tries to use them to resolve the anomaly. This involves instantiating the INTERNAL-XP-NODES and LINKS of each XP, and filling in the details through elaboration and specification. The PRE-XP-NODES of the XP are merged with corresponding nodes in the story representation. The instantiated XP is called an explanatory hypothesis, or simply hypothesis. If there are gaps in the XP, represented as pending questions attached to the XP, the questions are instantiated and the story representation is checked to see if the questions can be answered.

Hypothesis verification

The final step in the explanation process is the confirmation or refutation of possible explanations, or, if there is more than one hypothesis, discrimination between the alternatives. A hypothesis is a causal graph that connects the premises of the explanation to the conclusions via a set of intermediate assertions. The premises of the explanation are the XP-ASSERTED- NODES of the XP. XP-ASSERTED-NODES that assert facts that are not already known to be true in the story are turned into hypothesis verification questions (HVQs) for the hypothesis. If all the HVQs are confirmed, the hypothesis is confirmed (and its competitors refuted). If any HVQ is disconfirmed, the hypothesis is refuted.

The reasoner may use other methods for evaluating candidate hypotheses as well. Ram and Leake [1991] discuss several explanation evaluation methods, including those used by AQUA. As with the other steps in the explanation cycle, explanation evaluation is also implemented as a question-based process. At the end of this step, the reasoner is left with one or more alternative hypotheses. Partially confirmed hypotheses are maintained in a data dependency network called a hypothesis tree, along with questions representing what is required to verify these hypotheses.

Questions and explanation

From the point of view of questions, the process model for the task of explanation can be formulated as follows:

Anomaly detection

- Ask anomaly detection questions based on the goals, goal-orderings, plans, beliefs, and decisions represented in decision models.

XP retrieval

- Ask XP retrieval questions based on the indices used by AQUA and attempt to match the current situation to the PRE-XP-NODES of an available XP.
- Retrieve specific XPs based on XP retrieval questions.
- Apply specific XPs or abstract XPs if no specific XPs are found.

XP application

- Ask XP applicability questions based on the INTERNAL-XP-NODES and LINKS of the XP. Suspend XP application if necessary.
- (At this point, the XP may be tweaked, as in the SWALE program [Kass, Leake, and Owens, 1986].)
- Instantiate nodes of the XP.

- Instantiate links of the XP.

Hypothesis verification

- Ask hypothesis verification questions (HVQs) based on the XP-ASSERTED-NODES of the XP.
- Suspend hypothesis verification if necessary.
- Confirm/refute hypotheses later when HVQs are answered, and select the best hypothesis based on hypothesis evaluation criteria.

A taxonomy of explanation questions

The process model for question-driven explanation provides a functional basis for a taxonomy of questions that arise from the task of explanation. We now discuss the taxonomy of explanation questions in greater detail. This taxonomy is based on the explanation tasks that AQUA needs to perform when it reads a story. The categories will be illustrated using program transcripts showing examples of questions asked by AQUA as it reads a car bombing story.

Anomaly detection questions

Anomaly detection refers to the process of identifying an unusual fact that needs explanation. Anomalies fall into two categories:

1. **Physical anomalies**: Anomalies arising from reasoning about the physical causality behind the observed events. *E.g.*, if this is the first suicide bombing story one has read, one might think about the question “How can a car be used as a bomb?”
2. **Volitional anomalies**: Anomalies arising from reasoning about why the characters in the story acted as they did. *E.g.*, “Why would someone commit suicide if they were not depressed?”

Questions that help the understander in this task are called anomaly detection questions. These questions focus the understander on a particular aspect of the situation that might be anomalous. Once an anomaly is detected, the understander uses the anomaly characterization to retrieve potential explanations to resolve the anomaly. Thus the questions involved in the rest of the explanation process can be thought of as anomaly resolution questions.

Since the domain of AQUA focuses on human interest stories, the program deals mainly with volitional anomalies. Volitional anomalies can be categorized into two broad types:

1. **Contradictory knowledge:** Explicit contradiction of input or inferred information with knowledge or stereotypical information in memory. *E.g.*, the anomaly “Why do they choose kids for bombing missions?” arises if one expects that the terrorists would recruit well-trained military men for these difficult missions. [Note: AQUA uses a simple template-based natural language generator to describe concepts in memory. In the following transcripts, the generator output has been cleaned up to some extent for the sake of readability. Since AQUA actually uses the representation of questions in memory, not their printed output, the actual form of the output is not relevant to the operation of the program.]

S-3: Terrorists recruit boy as car bomber.

```
Trying to explain WHY DID THE TERRORIST GROUP RECRUIT THE BOY TO DO
                THE CAR BOMBING?

Anomaly! THE BOY is not a typical MILITARY AGENT.
```

2. **Missing knowledge:** Lack of explanatory knowledge in memory. These anomalies arise, not out of explicit contradictions, but out of the lack of some information that was expected to be present in memory. In other words, these anomalies arise out of the noticing of gaps in memory. *E.g.*, the anomaly "Why would a person who was not a fanatic go on a suicide bombing mission?" arises when the understander realizes that its standard religious fanatic explanation is inapplicable, and it has no further explanations that apply to this situation. Similarly, in the following example, AQUA detects an anomaly for a novel action for which it has no explanations in memory:

S-4: The terrorist group surrendered to the Israeli police.

```
Trying to explain WHY DID THE TERRORIST GROUP SURRENDER TO THE
                ISRAELI POLICE

Anomaly! THE SURRENDER violates THE GOAL OF THE TERRORIST GROUP
                TO PRESERVE THE INDEPENDENCE OF THE TERRORIST GROUP.

Characterized outcome as a BAD outcome for the ACTOR

Searching for stereotypical XPs

Anomaly! No XPs for why THE TERRORIST GROUP SURRENDERED TO THE
                ISRAELI POLICE.
```

The first anomaly in the above transcript arises from a contradiction, whereas the second one arises when AQUA encounters a gap in its memory. If AQUA did have an applicable XP to start with, or is able to fill this gap by learning a new XP that explains why terrorists surrender, the XP would be retrieved and applied to the story. This action would then be "explained" (and therefore not anomalous) by virtue of the fact that it has been fitted into an XP that the program has.

Volitional anomalies are detected by asking a series of anomaly detection questions about the input. These questions arise by questioning different parts of the planning/decision models discussed earlier. For example, the understander could question the goals of the agent, or his beliefs about the expected outcome of the action, or his volition in choosing to perform the action. Each of these questions uncovers a different type of anomaly, and proposes a different type of explanation for the anomaly. For example, questioning the goals of the agent allows the understander to detect goal violation anomalies, in which the agent performs an action that violates his own goals.

In order to notice anomalies and build explanations based on the planning/decision model of volition, AQUA's questions must be generated from this model. This is done by walking over the representation of abstract XPs and questioning the applicability conditions of these XPs. For example, the abstract XP "Actor chooses to perform an action with a negative outcome because he did not know that the action would result in this outcome" relies on the beliefs of the actor about the outcome of the action. This is represented as one of the PRE-XP-NODES of the XP. This node gives rise to the question: Did the actor know that the action would have this negative outcome?

Before asking this question, however, AQUA must determine whether the outcomes of the action are indeed negative from the point of view of the actor. Thus the XP “Actor chooses to perform an action because he wants the outcome of the action” must be tried first. This XP gives rise to the question: Does the actor want the outcome of the action? This question must be asked before the question about the actor's beliefs.

In principle, the understander could order its XPs at run-time by checking their applicability conditions to see which ones presuppose the others. Since AQUA does not learn new abstract XPs, abstract XPs are statically organized into a hierarchy that determines the order in which they will be checked. AQUA does learn new stereotypical XPs, which are specific versions of abstract XPs for particular situations; however, this does not require modification of the hierarchy of abstract XPs since new stereotypical XPs are indexed using existing abstract XPs as the category index. AQUA traverses down this hierarchy, generating questions based on the PRE-XP-NODES of each XP. These questions, therefore, can be viewed as comprising a discrimination net of volitional questions, in which each question raises further questions if its answer seems anomalous .

For example, the question "Does the actor want the outcome of the action?", if answered negatively, would signal an anomaly and raise further questions such as "Did the actor know the outcome of the action?" and "Is there another result of this action, perhaps currently unknown, which the actor desired even at the expense of the outcome that he didn't want." These questions are indexed in memory and used to determine the interesting aspects of the story. The above questions represent the fact that AQUA is interested in the beliefs of the boy, as well as further results of the suicide bombing mission. When these questions are answered later in the story, the corresponding explanation is re-activated, causing AQUA to focus on the inferences relevant to its questions. The theory of question-driven understanding, therefore, provides a principled method for determining interestingness and focussing attention.

Anomaly detection questions can be categorized as follows:

Decision questions: These questions focus on the decision that the actor took when he decided to do the action. Therefore, this is also a taxonomy of the planning decisions one would consider when deciding to do an action.

Personal goals:

- Does the actor want the outcome of this action?
- Does the actor want to avoid a negative outcome of not doing this action?
- Does the actor want a positive outcome of this action more than he wants to avoid a negative outcome of doing the same action?

- Does the actor enjoy doing that action?
- Does the actor habitually do this action?

Instrumentality:

- Is this action instrumental to another action that the actor wants to carry out?
- Is this action part of a larger plan that the actor is carrying out?

Interpersonal goals:

- Does the actor want a positive outcome of this action for someone he likes?
- Does the actor want to avoid a negative outcome of this action for someone he likes?
- Does the actor want a negative outcome of this action for someone he dislikes?
- Does the actor want a positive outcome of this action for a group that he belongs to?
- Does the actor want to avoid a negative outcome of this action for a group that he belongs to?
- Does the actor feel gratification in doing good for others?

Social control:

- Did someone with social control over the actor ask him to perform the action?
- Did someone with social control over the actor force him to perform that action?

Knowledge and beliefs:

- Did the actor know the probable outcomes of the action?
- Did the actor believe that the action would have a positive outcome for him?
- Did the actor know about the possible negative outcome of the action?

Interference questions: These questions focus on possible interference from external sources.

- Did someone want to block the actor's goal?
- Did someone want to prevent this state of the world? Would this state of the world violate this person's goals?
- Did someone want the actor to be involved in this action?
- Did the actor accidentally get involved in this action?

For example, consider the above story S-4 again. These are the questions that lead to the anomaly begin detected in this story:

S-4: The terrorist group surrendered to the Israeli police.

```
Trying to explain WHY DID THE TERRORIST GROUP SURRENDER TO THE ISRAELI POLICE?

Did THE TERRORIST GROUP want the outcome of THE SURRENDER?

Characterizing the outcome CAPTURED-STATE
    of THE SURRENDER
     from the point of view of THE TERRORIST GROUP (the ACTOR)

DOES THE TERRORIST GROUP WANT TO ACHIEVE THE CAPTURED STATE OF THE
     TERRORIST GROUP

Anomaly! THE SURRENDER violates THE GOAL OF THE TERRORIST GROUP TO
             PRESERVE THE INDEPENDENCE OF THE TERRORIST GROUP.

Characterized outcome as a BAD outcome for the ACTOR
```

In the above output, only the questions that led to the anomaly are shown. To take a more complete example, consider the following abbreviated version of story S-1:

S-5: Terrorists recruit boy as car bomber.

A 16-year-old Lebanese got into an explosive-laden car and went on a suicide bombing mission to blow up the Israeli army headquarters in Lebanon. ... The teenager was a Shiite Moslem but not a religious fanatic. He was recruited for the mission through another means: blackmail.

When AQUA reads this story, it asks the following anomaly detection questions which focus on the personal goals in the actor's decision model:

```
Trying to explain WHY DID THE TEENAGE LEBANESE BOY DO THE SUICIDE
                  BOMBING?

Was THE SUICIDE BOMBING instrumental to another action?

Does THE TEENAGE LEBANESE BOY typically do SUICIDE BOMBINGs?

Did THE TEENAGE LEBANESE BOY want the outcome of THE SUICIDE BOMBING?

Characterizing the outcomes DEATH-STATE and DESTROYED-STATE
     of THE SUICIDE BOMBING
     from the point of view of THE TEENAGE LEBANESE BOY (the ACTOR)

DOES THE TEENAGE LEBANESE BOY WANT TO ACHIEVE THE DEATH OF THE
     TEENAGE LEBANESE BOY?

Anomaly! The SUICIDE BOMBING violates THE GOAL OF THE BOY TO PRESERVE THE LIFE
STATE OF THE BOY.

DOES THE TEENAGE LEBANESE BOY WANT TO ACHIEVE THE DESTRUCTION OF `
     ISRAELI ARMY HEADQUARTERS IN LEBANON?
No relevant GOALS found

Did THE TEENAGE LEBANESE BOY want to avoid a negative outcome of
     not doing THE SUICIDE BOMBING?
No relevant OUTCOMES found
Did THE TEENAGE LEBANESE BOY enjoy doing the SUICIDE BOMBING?
```

```
No relevant GOALS found

Did THE TEENAGE LEBANESE BOY habitually do SUICIDE BOMBINGs?
No relevant ACTIVITIES found

Characterized outcome as a BAD outcome for the ACTOR
```

In addition to the above questions, which focus on the actor's reasons for performing an action, AQUA also considers the planner's reasons for planning that action, if the actor and the planner are different. Many of the questions are similar to the above questions. For example, AQUA would ask whether the planner wanted an outcome of the action, whether he knew the outcome of the action, etc. Thus for the above story, AQUA also asks the following questions:

```
Trying to explain WHY DID THE TERRORIST GROUP PLAN THE SUICIDE
               BOMBING?

Did THE TERRORIST GROUP want the outcome of THE CAR BOMBING?

Characterizing the outcomes DEATH-STATE and DAMAGED-STATE
    of THE SUICIDE BOMBING
     from the point of view of THE TERRORIST GROUP (the PLANNER)

DOES THE TERRORIST GROUP WANT TO ACHIEVE THE DEATH OF THE TEENAGE
     LEBANESE BOY?
No relevant GOALS found

DOES THE TERRORIST GROUP WANT TO ACHIEVE THE DESTRUCTION OF THE
     ISRAELI ARMY HEADQUARTERS IN LEBANON?
Matches typical GOALS

Dud THE TERRORIST GROUP typically do SUICIDE BOMBINGs?
Matches typical ACTIVITIES

No anomaly detected
```

In addition, AQUA also asks the following questions which focus on the interaction between the planner and the actor:

Planner questions: Given an action that was planned and executed by different people or groups:

- Did the action result in a positive outcome for both the planner and the actor?
- Did the planner select the actor knowing that the action would result in a negative outcome for the actor?

For example, if one wonders why the planner gave the necessary resources to the actor and then realizes that one of the planner's goals was achieved by the action, one can hypothesize that a contract or exchange exists between the two.

When answered, anomaly detection questions either result in an anomaly being detected, in which case the anomaly needs to be resolved through explanation, or in no anomaly being detected, in which case an explanation has implicitly been found. For example, if the answer to “Did the action result in a positive outcome for the actor?” is yes, there is no anomaly; the explanation implicit in this question is the abstract one of “Actor does action to satisfy a goal.” An explanation at this level is sufficient in many situations; a deeper explanation is required only when there is an anomaly at this level.

XP retrieval questions

When faced with an anomalous situation, AQUA tries to retrieve one or more XPs that would explain the situation. In order to do this, AQUA asks a series of XP retrieval questions about the situation. These questions are also called anomaly resolution questions, since their intent is to find explanations that help resolve anomalies. XP retrieval questions focus the understander's attention on a particular aspect of a situation, or allow it to view a situation in a particular way, with the intention of finding an explanation that might underlie it. These questions fall into two broad categories, corresponding to the taxonomy of motivational explanations:

1. **Abstract XP retrieval questions**: These questions focus the understander's attention on the goals, plans and beliefs of the character. For example, the question “Did the boy want to kill himself?” focuses on the boy's goals with respect to this particular outcome, the death of the boy. If the answer to this question is yes, an explanation might be that the boy actually did want to kill himself (a suicidal teenager perhaps), and suicide bombing was just a bizarre way of doing this. Another question of this type is “Did the boy know he was going to die?” This focuses on the boy's beliefs.
2. **Stereotypical XP retrieval questions**: These questions attempt to view the character as belonging to a particular stereotype. Their intent is to enable the understander to retrieve stereotypical XPs specific to the particular situation. For example, the question “Was the boy a zealous Shiite Moslem?” focuses on the religious beliefs of the boy. An affirmative answer to this question would cause AQUA to retrieve the “religious fanatic” XP, since this XP is indexed under a religious fanatic stereotype.

While stereotypical explanations are not guaranteed to be correct, case-based explanation is based on the assumption that it is too inefficient to reason from scratch, starting from the general theory of motivations, in every situation. Stereotypical XPs provide a way to associate specific stereotypical causal rules with specific stereotypical situations and people. The features characterizing these stereotypes result in XP retrieval questions, which in turn allow the system to retrieve large explanation structures (XPs) directly. Applying these XPs results in explanatory hypotheses which can then be evaluated. Thus an XP retrieval question is a heuristic to find XPs that might be relevant without having to do the inferencing required to check if they are indeed relevant, which requires determining whether the causality underlying the situation matches that underlying the XP. If the question results in an XP being retrieved, the understander does the rest of the work to determine if the XP is indeed relevant.

XP retrieval questions at the abstract level of abstract XPs focus on the goals, goal priorities, beliefs, and decisions of the people involved, whereas those at the specific level of stereotypical XPs focus on particular stereotypes of people that are known to be associated with those kinds of actions. Thus XP theory replaces general matching techniques for goals and plans with specific applicability conditions for stereotypical situations. The former are applicable to a greater range of situations but harder to determine; the latter are specific to particular situations but easier to match, infer, apply and verify. This is the basic principle underlying case-based reasoning (and other methods of reasoning based on "large" knowledge structures such as scripts, frames and schemas).

The following taxonomy of XP retrieval questions provides a "content theory" of the knowledge needed to index and retrieve XPs. The taxonomy at the level of abstract XPs mirrors the taxonomy of anomaly detection questions. Since the particular XP retrieval questions at the level of stereotypical XPs depend on the stereotypes currently in memory, this category will be illustrated using examples rather than a taxonomy.

Abstract XP retrieval questions:

Decision anomalies:

- Anomaly: **goal-violation**

  Situation: Actor does action that results in a negative outcome.

  Questions:

  – Did the actor actually want this outcome (*i.e.*, did we misperceive his goals?)

  – Did the action result in another outcome that the actor wanted even at the expense of the negative outcome?

  – Was the actor forced into doing this action?

  – Did the actor know that the action would have this negative outcome? ?

  - Did the actor have enough information about the environment?
  - Did the actor project the effects of the action correctly?

For example, since there is a goal-violation anomaly in the above story, AQUA asks the following abstract XP retrieval questions:

```
Anomaly! The SUICIDE BOMBING violates THE GOAL OF THE BOY TO PRESERVE
    THE LIFE STATE OF THE BOY.

Searching for abstract XPs

DID THE SUICIDE BOMBING RESULT IN A STATE?
and WAS THE GOAL OF THE BOY TO ACHIEVE THE STATE MORE IMPORTANT THAN
    THE GOAL OF THE BOY TO PRESERVE THE LIFE OF THE BOY?

No other RESULTS of THE SUICIDE BOMBING known.
Suspending explanation

DID THE BOY BELIEVE THAT THE SUICIDE BOMBING WOULD RESULT IN
    THE DEATH STATE OF THE BOY?
No relevant BELIEFS found
Suspending explanation
```

These questions are indexed in memory, and are used to determine the interesting aspects of the story. The above questions represent the fact that AQUA is interested in the GOALS and BELIEFS of the boy, as well as further results of the suicide bombing mission. If a question is answered later in the story, the corresponding explanation is re-activated. For example, suppose that S-5 were to be continued as follows:

S-5: ... The boy was told that the car bombing would not cause him any harm.

Answering question: DID THE BOY BELIEVE THAT THE SUICIDE BOMBING
WOULD RESULT IN THE DEATH STATE OF THE BOY?
with: THE BOY DID NOT BELIEVE THAT THE SUICIDE BOMBING WOULD
RESULT IN THE DEATH STATE OF THE BOY.

Restarting suspended explanation
THE BOY DECIDED TO DO THE SUICIDE BOMBING DESPITE THE VIOLATION
OF THE GOAL OF THE BOY DO PRESERVE THE LIFE STATE OF THE BOY because
THE BOY DID NOT KNOW THAT THE SUICIDE BOMBING WOULD RESULT
IN THE DEATH STATE OF THE BOY.

Of course, this still does not explain why the boy did the suicide bombing, but the goal-violation anomaly is resolved. Once this explanation is confirmed, the other hypotheses are retracted.

Continuing with the taxonomy of abstract XP retrieval questions:

- Anomaly: **goal-violation** or **unusual-goal-ordering**

  Situation: Actor does action that results in a positive outcome and a negative outcome.

  Questions:

  - Does the actor prefer to achieve the positive outcome even at the expense of the negative outcome?

  – Does the actor actually want to avoid the negative outcome?

  – Can the goal violated by the negative outcome be pursued later?

  XPs:

  – Goal priority elevation in particular contexts.

  • Goal violated by negative outcome can be pursued later.

  • Goal of positive outcome is temporarily urgent.

  • Short term goals preferred to longer term goals.

  • Personal goals preferred to group goals.

  • Difficult goals postponed.

  • New goals from wanting what others have.

  – Actor's goal priorities were misperceived.

  • Personal differences (individual, parental).

  • Group differences.

  • Cultural differences.

- Anomaly: **bad-plan-choice**

  Situation: Actor does action to achieve a goal even though another action looks better.

  Questions:

  – Did the second action have a negative side effect for the actor?

  – Did the actor know about the second action?

  – Was the actor capable of performing the second action?

  – Is the first action better in the long run? ? Is the cumulative effect of the first action better? ? Does the first action keep more options open?

  – Does the actor enjoy doing the first action?

- Anomaly: **failed-opportunity**

  Situation: Actor doesn't do action that would have resulted in a positive outcome for actor.

  Questions:

  – Did the actor actually want this outcome (*i.e.*, did we misperceive his goals)?

  – Did the actor know that the action would result in the positive outcome?

  – Did the action also result in a negative outcome for the actor?

  – Was the actor capable of performing that action?

- Anomaly: **unmotivated-action**

  Situation: Actor does an action that doesn't satisfy any of his goals.

  Questions:

  – Did the actor think that the action would satisfy one of his goals?

  – Does the action actually satisfy a goal (*i.e.*, did we misperceive the situation)?

Planner anomalies:

- Anomaly: **malicious-intent**

  Situation: Planner knowingly recruits actor for action that results in negative outcome for actor.

  Questions:

  – Was the planner's real intention to achieve a negative outcome for the actor?

  – Did the planner want some other outcome of the action, but also wanted to avoid the negative outcome from happening to himself?

  – Was the planner willing to sacrifice the actor's goal to achieve a goal of his own?

  – Did the planner want both the outcomes, *i.e.*, was he killing two birds with one stone?

  – Was the planner in turn forced to make this decision?

In the above story, for example, the **actor-planner** interaction gives rise to the following questions:

```
The PLANNER, THE TERRORIST GROUP, is not the same as the ACTOR,
THE BOY

Anomaly! The PLANNER, THE TERRORIST GROUP, planned an action with a BAD outcome for
the ACTOR, THE BOY.

Searching for abstract XPs
DID THE TERRORIST GROUP WANT TO ACHIEVE THE DEATH STATE OF THE BOY?

DID THE TERRORIST GROUP WANT TO DESTROY THE ISRAELI ARMY
    HEADQUARTERS?
and DID THE TERRORIST GROUP WANT TO AVOID THE DEATH STATE OF THE
    TERRORIST GROUP?

WAS THE GOAL OF THE TERRORIST GROUP TO DESTROY THE ISRAELI ARMY
    HEADQUARTERS MORE IMPORTANT THAN THE GOAL OF THE TERRORIST GROUP
    TO PRESERVE THE LIFE STATE OF THE BOY?
```

If any of these questions are answered, the corresponding XP is retrieved and applied to the story. If the explanation can be confirmed, the anomaly is resolved.

Stereotypical XP retrieval questions:

- XP: Religious fanatic does suicide bombing.
  Questions:
  – Was the actor religious?
  –(If in the Middle East) Was the actor a Shiite Moslem?

- XP: Depressed teenager commits suicide.
  Questions:
  – Was the actor a stereotypical teenager?
  – Was the actor depressed?

In the current example, S-5, AQUA finds the boy's suicide bombing action anomalous. It tries to retrieve XPs to explain this action and finds the religious fanatic XP indexed under the lebanese-person stereotype and the suicide-bombing MOP:

```
Searching for stereotypical XPs

Asking EQ: IS THE BOY A TYPICAL TEENAGER?
Asking EQ: WHY WOULD A TEENAGER DO A SUICIDE BOMBING?

    Situation index = SUICIDE-BOMBING
    Stereotype index = TEENAGER

No XPs found

Asking EQ: IS THE BOY A TYPICAL LEBANESE PERSON?
Asking EQ: WHY WOULD A LEBANESE PERSON DO A SUICIDE BOMBING?

    Situation index = SUICIDE-BOMBING
    Stereotype index = LEBANESE-PERSON

Retrieved stereotypical XPs:
    XP-RELIGIOUS-FANATIC (category index = XP-SACRIFICE)
```

AQUA also tries to retrieve abstract XPs to explain the anomaly. For example, since the anomaly is one in which the actor performs an action with a negative outcome for himself, AQUA asks the abstract XP retrieval questions that were described earlier. Abstract XPs are used only if no specific stereotypical XPs can be found. For example, consider the question:

```
IS THE GOAL OF THE TEENAGE LEBANESE BOY TO ACHIEVE THE DESTRUCTION OF
    THE ISRAELI ARMY HEADQUARTERS IN LEBANON MORE IMPORTANT THAN THE
    GOAL OF THE TEENAGE LEBANESE BOY TO PRESERVE THE LIFE OF THE
    TEENAGE LEBANESE BOY?
Not known
```

This question, when answered, would lead to xp-sacrifice, but since a stereotypical XP of this type has already been found (xp-religious-fanatic), the abstract XP is not used to explain this anomaly.

XP application questions

Once a set of potentially applicable XPs is retrieved, the understander tries to use them to resolve the anomaly. This involves instantiating the XP, filling in the details through elaboration and specification, and checking the validity of the final explanation. Questions that help the understander elaborate explanations, or collect more information about the input, to help it construct a coherent understanding of the input are called XP elaboration questions and data collection questions, respectively.

When an XP is retrieved, it is instantiated to form a hypothesis. If this process raises any XP application questions, the explanation is suspended until answers to these questions become known. To illustrate this, consider the application of xp-religious-fanatic, retrieved in the previous step, to the current story. These questions correspond to the actual definition of xp-religious-fanatic shown earlier.

```
Applying XP-RELIGIOUS-FANATIC to WHY DID THE BOY DO THE SUICIDE BOMBING.

    THE BOY THE SUICIDE BOMBING
    because THE BOY WAS   A  RELIGIOUS FANATIC.

Unifying EXPLAINS node

Installing NODES

Installing LINKS

    THE BOY IS A RELIGIOUS FANATIC
    because THE BOY IS A SHIITE MOSLEM

    THE BOY WANTS TO ACHIEVE THE DESTRUCTION OF THE ISRAELI ARMY
    HEADQUARTERS
    because THE BOY IS A RELIGIOUS FANATIC

    THE GOAL OF THE BOY TO ACHIEVE THE DESTRUCTION OF THE ISRAELI ARMY
    HEADQUARTERS IS MORE IMPORTANT THAN GOAL TO PRESERVE THE LIFE STATE
    OF THE BOY
    because THE BOY IS A RELIGIOUS FANATIC

    THE BOY DECIDED TO DO THE SUICIDE BOMBING
    because XP-SACRIFICE
```

In this case, there are no XP application questions since the religious fanatic XP is applicable to the situation. However, the resulting hypothesis still needs to be verified.

Hypothesis verification questions

The final step in the explanation process is the confirmation or refutation of possible explanations, or, if there is more than one hypothesis, discrimination between the alternatives. Hypothesis verification questions (HVQs) are questions that arise from this task. For example, although there is no real difficulty in applying the religious fanatic explanation in story S-5, the explanation rests on certain assumptions. To verify the hypothesis, AQUA asks what the religion of the boy was, and whether he believed fanatically in that religion. When it reads that “the boy was a Shiite Moslem but not a religious fanatic,” it answers these questions and refutes the hypothesis.

To generate the HVQs for a hypothesis, AQUA checks each of the XP-ASSERTED-NODES of the hypothesis. In the xp-religious-fanatic example, this raises the following hypothesis verification questions:

```
Installing HVQs to verify XP:

    WHAT IS THE RELIGION OF THE TEENAGE LEBANESE BOY?
    WHAT IS THE RELIGIOUS ZEAL OF THE TEENAGE LEBANESE BOY?
```

This process is repeated for all the possible hypotheses. When this is done, AQUA is left with one or more alternative hypotheses, each with its own set of HVQs. This is represented using a hypothesis tree as described earlier. When new facts come in, AQUA checks to see if these facts would answer any questions in memory. If an HVQ is answered, AQUA re-examines the hypothesis to see whether it has been confirmed or refuted:

- If the HVQ is answered negatively, refute the hypothesis.
- If the HVQ is answered positively and this is the last HVQ for the hypothesis, confirm the hypothesis and refute its competitors.
- In each case, re-evaluate belief in corresponding hypothesis.

Thus when an HVQ is answered, AQUA knows what to do with the answer since the hypothesis structure represents the suspended explanation task that is waiting for the answer. In the current story, the xp-religious-fanatic hypothesis is eventually refuted when AQUA reads the sentence:

S-5: ... The teenager was a Shiite Moslem but not a religious fanatic.

```
Answering question: WHAT IS THE RELIGION OF THE BOY?
                     with: THE BOY IS A SHIITE MOSLEM.

Answering question: WHAT IS THE RELIGIOUS ZEAL OF THE BOY?
                     with: THE BOY IS NOT VERY ZEALOUS ABOUT THE SHIITE
                                  MOSLEM RELIGION.

Refuting hypothesis:
    THE BOY DID THE SUICIDE BOMBING
    because THE BOY WAS A RELIGIOUS FANATIC.
```

Evaluation criteria

We have discussed how explanations are constructed through a case-based reasoning process, resulting in one or more abductive hypotheses. Regardless of how explanatory hypotheses are constructed, however, the evaluation of these hypotheses is a central and difficult problem. We categorize evaluation criteria into structural (or syntax-based) and utility-based (or goal-based) criteria [Ram and Leake, 1991].

Structural criteria

Structural criteria use the structural or syntactic properties of the causal chain to evaluate hypotheses. A goodness measure for each hypothesis is computed based on the length of the causal chain, the number of abductive assumptions, or other such structural properties.

Most structural criteria appeal to Occam's razor by requiring minimality of hypotheses. Simply stated, a hypothesis that is “minimal” with respect to some criterion is preferred over one that is not (*e.g.*, Charniak [1986]; Kautz and Allen [1986]). For example, Konolige [1990] argues that “closure + minimization implies abduction.” To take another example, the TACITUS system for natural language interpretation merges redundancies as a way of getting a minimal interpretation, which is assumed to be a best interpretation [Hobbs, Stickel, Appelt, and Martin, 1990]. Minimality criteria include:

- **Length**: Causal chains with the shortest overall length are preferred.
- **Abductive assumptions**: Explanations requiring the fewest abductive assumptions are preferred.
- **Subsumption**: If two candidate hypotheses are found and one subsumes the other, the more general hypothesis is preferred.

Another approach focuses on the structural relationship of propositions in an explanation rather than minimality:

- **Explanatory coherence**: The cohesion of an explanation is measured, based on the form of connections between an explanation's propositions, and the “best connected” explanation is favored (*e.g.*, Thagard [1989]; Ng and Mooney [1990].)

While structural criteria provide an easy way to evaluate the goodness of a hypothesis, they are may not be sufficient to identify the best explanation in real situations. The shortest explanation, for example, may or may not provide enough information to understand how the details of a given story fit together. In general, explanations are not constructed in a vacuum; there is a real-world task that the reasoner is performing that requires the reasoner to seek an explanation in the first place. The reasoner may also need an explanation to help it with a piece of reasoning that it is trying of perform. Both these types of motivations for explanation influence evaluation criteria.

Utility-based criteria

An reasoner's motivation for explaining will often place additional requirements on candidate explanations, beyond their form. For example, explanations prompted by anomalies must provide particular information, in order to resolve the anomaly. For example, suppose that we expected team X to win over team Y because of the talent of X's star player, but we are told that team X actually lost. If someone explained the loss by "Y scored more points than X,” the explanation would be inadequate. Although it is a correct explanation, it gives no information about why our expectation went wrong. The explanation “X's star was injured and couldn't play” does account for what was neglected in prior reasoning, and consequently is a better explanation. However, this explanation would not be preferred on structural grounds alone. The causal chain underlying that explanation is more complex, so it would not be favored by minimality criteria. Likewise, the explainer of the game has access to only one observation, the fact that team X lost, so coherence metrics that measure how an explanation relates pairs of observations, such as those described by Ng and Mooney [1990], give no grounds for preferring the second explanation.

To state our relevance criterion another way, an explanation must address the failure of the reasoner to model the situation correctly. In addition to resolving the incorrect predictions, it must also point to the erroneous aspect of the chain of reasoning that led to those predictions. An explanation is useful if it allows the reasoner to learn, or to accomplish current tasks. The claim here is that an explanation must be both causal and relevant in order to be useful.

An explanation of an anomaly, therefore, must answer two types of questions:

1. **Why did things occur as they did in the world?** This question focuses on understanding, and learning about, the causal structure of the domain.
2. **Why did I fail to predict this correctly?** This question focuses on understanding, and improving, the organization of the reasoner's own model of the domain.

The answer to the first question is called a domain explanation since it is a statement about the causality of the domain. The answer to the second question is called an introspective or meta-explanation since it is a statement about the reasoning processes of the system.

Each of the above questions relates to a need to collect or organize the missing information that caused the anomaly, and that utility-based evaluation criteria must address. Let us consider the second question first.

Introspective explanations

One of the questions an explanation must address is why the reasoner failed to make the correct prediction in a particular situation. In an XP-based system, this could happen in three ways:

1. **Novel situation**: The reasoner did not have the XPs to deal with the situation.
2. **Incorrect world model**: The XPs that the reasoner applied to the situation were incomplete or incorrect.
3. **Misindexed domain knowledge**: The reasoner did have the XPs to deal with the situation, but it was unable to retrieve them since they were not indexed under the cues that the situation provided.

When an explanation is built, the reasoner needs to be able to identify the kind of processing error that occurred and invoke the appropriate learning strategy to prevent recurrence of the error. For example, if an incomplete XP is applied to a situation, the knowledge activated by the resulting processing error must represent both the knowledge that is missing, and the fact that this piece of knowledge, when it comes in, should be used to fill in the gap in the original XP. Similarly, if an error arose due to a misindexed XP, the explanation, when available, should be used to re-index the XP appropriately. The learning algorithms used in AQUA are discussed in more detail in Ram [1993].

In general, a reasoner can encounter other kinds of difficulties as well (*e.g.,* see Ram and Cox [1993]). Knowledge goals can be categorized by the types of gap or inadequacy in the reasoner's knowledge, by the types of failures or difficulties during processing, or by the types of learning that result. A hypothesis is evaluated from the point of view of knowledge organization goals by checking to see if it provides the information necessary for the type of learning that the reasoner is trying to perform. For example, suppose the reasoner reads a newspaper story about a Lebanese teenager who, it turns out, is blackmailed into going on a suicide bombing mission. Even if the reasoner already knows about terrorism, religious fanatics and blackmail, the story may nevertheless be anomalous if the reasoner has never seen this particular scenario before. The difficulty arises from the fact that blackmail is not ordinarily something that comes to mind when one reads about suicide bombing. Here, the reasoner can learn a new connection between the knowledge structures describing suicide bombing and blackmail, respectively. In order to do this, the explanation must provide the information required to identify the conditions under which a suicide bombing is likely to be caused through blackmail.

This type of analysis is essential in determining whether an explanation is sufficient for the purposes of the reasoning task at hand. In this example, the reasoning task is to satisfy a knowledge organization goal, which is a question that represents a goal to learn by reorganizing existing knowledge in memory.

Domain explanations

Another kind of knowledge goal is a knowledge acquisition goal, which is a question that seeks to acquire new causal knowledge about the domain. Such a question is answered using a domain explanation, which is a causal chain that demonstrates why the anomalous proposition might hold by introducing a set of premises that causally lead up to that proposition. If the reasoner believes or can verify the premises of an explanation, the conclusion is said to be explained. Explanations are often verbalized using their premises or abductive assumptions. However, the real explanation includes the premises, the causal chain, and any intermediate assertions that are part of the causal chain.

In order to be useful, a hypothesis must provide the information that is being sought by the knowledge acquisition goals of the reasoner. For example, if the reasoner has a goal to acquire knowledge about the biochemical properties of a particular virus, a description of a sick patient must provide the biochemical information in order to qualify as an explanation from the point of view of that goal. An alternative hypothesis that provides causal information suggesting how some drug might destroy the virus, while useful from the point of view of curing the patient, may not provide the required information.

AQUA uses the following criteria to evaluate hypotheses. Although AQUA uses a case-based approach using explanation patterns to construct explanations, the criteria listed here are also applicable to other kinds of explanation construction methods which rely on domain knowledge in the form of inference rules, cases, schemas, or other types of knowledge structures.

1. **Believability**: Do I believe the domain knowledge from which the hypothesis was derived? This is an issue for any learning program in a realistic domain for which a correct domain theory is not yet known.

2. **Applicability**: How well does the domain knowledge (the particular rules, cases or schemas) apply to this situation? Did it fit the situation without any modifications?

3. **Relevance**: Does the hypothesis address the underlying anomaly? Does it address the knowledge goals of the reasoner? In AQUA, the hypothesis is evaluated in the context of both knowledge acquisition and knowledge organization goals.

4. **Verification**: How definitely was the hypothesis confirmed or refuted in the current situation? Does the hypothesis spawn new knowledge goals (requiring further information to help verify the hypothesis)?

5. **Specificity**: Is the hypothesis abstract and very general, or is it detailed and specific? This is a structural criterion in the sense that it is based on the structure, and not the content, of the hypothesis. However, the structure of the hypothesis is evaluated in the context of the organization of causal memory.

Intuitively, a “good” explanation is not necessarily one that can be proven to be “true” (criterion 4), but also one that seems plausible (1 and 2), fits the situation well (2 and 5), and is relevant to the goals of the reasoner (criterion 3).

AQUA is a dynamic story understanding program that is driven by its questions or goals to acquire knowledge. Rather than being “canned,” the program is always changing as its questions change; it reads similar stories differently and forms different interpretations as its questions and interests evolve. AQUA judges the interestingness of the input with respect to its knowledge goals [Ram, 1990c], and learns about the domain by answering its questions [Ram, 1993]. Both these processes are goal-based. Here, we are proposing that the evaluation of explanations be goal-based (or question-based) as well.

Using questions to guide explanation

Questions in AQUA's memory represent gaps in AQUA's model of the domain. These questions serve as "knowledge goals", the system's goals to acquire, reorganize, or reformulate knowledge in order to learn more about the domain. Some questions arise from unconfirmed hypotheses that the system is entertaining, or has entertained in a previous story. Other questions arise from other kinds of gaps in the system's knowledge or other kinds of difficulties during processing. Questions play a central role in reasoning and learning. In this chapter, we have focussed on the role of questions in explanation; other aspects of our theory of questions and knowledge goals are discussed in Ram [1989; 1991; 1993], Ram and Cox (1993), and Ram and Hunter [1992].

A program that uses its questions to guide explanation is an improvement over one that processes everything in equal detail, that is, one that is completely data-driven. For example, an understander that is completely text-driven would process everything in detail in the hope that it might turn out to be relevant. To avoid this, the understander should draw only those inferences which would help it find out what it needs to know. In other words, the understander should use its knowledge goals to focus its attention on the interesting aspects of the story, where "interesting" can be defined as "relating to something the understander wants to find out about."

It is useful to focus on questions because they arise from a "need to learn." There are two basic ways in which a fact can turn out to be worth processing in this sense:

Top-down: A fact that helps achieve a knowledge goal, or answers a pending question, is worth focussing on since it allows the reasoning system to continue the reasoning task that required the knowledge in the first place.

Bottom-up: A fact that gives rise to new knowledge goals, or raises new questions, is worth focussing on if the knowledge goals arise from a gap or inconsistency in the reasoning system's knowledge base, since the system may be able to improve its knowledge base by learning something new about the world.

The real issue here, of course, is how much inference should be done at the time the questions are generated, and how much should be done when the input comes in. The answer depends on six factors:

1. **Certainty of inference**: The probability of the inference rules used to find or infer answers to questions, or the likelihood that the conclusions will be true. In a logic system where an inference rule represents a deduction, this probability is 1.
2. **Cost of inference**: The cost of making inferences or of matching and applying inference rules. The cheaper the inference, the more it is worth the system's while to make it.
3. **Usefulness of question**: The usefulness of the conclusion that the question is seeking. Since questions are generated in service of reasoning tasks, this is the same as the importance of performing that task. If the task is very important, it is worth making the inference even if it is very expensive to do so.
4. **Likelihood of question being useful**: The likelihood that the question will be useful, *i.e.*, the likelihood that the knowledge will actually turn out to be useful in performing the reasoning task. If questions are only generated from tasks that absolutely require that knowledge (as opposed to those that may be facilitated by that knowledge if it were present), this likelihood is 1.
5. **Indexing cost**: The cost of keeping indexed questions in memory and matching to them. If there are too many questions in memory, it might be too expensive to find them or to match input to potentially relevant questions. This cost depends on the scheme used to maintain questions in memory, and is discussed below.
6. **Likelihood of question being satisfied**: The above factors are "content-free" heuristics in the sense that the reasoning system does not rely on knowledge of the content or types of questions that it is likely to generate, or on the content or types of inferences that the system is likely to make when given new input. In addition, one would like the system to generate the types of questions or knowledge goals that are likely to match the inferences normally made by the bottom-up processing that is always performed on

> incoming facts. The last criterion for inference control, therefore, is the likelihood of a question being satisfied, which depends on knowledge about the inferences that are likely to be made by the system's own inference processes.

The above heuristics are not represented formally in the AQUA program. In other words, there are no explicit functions to compute each of these metrics and to make a decision based on them. However, the heuristics to determine the utility or interestingness of questions, and to index questions in memory, have been designed keeping these metrics in mind so that the process is efficient. More research is required to develop a theory of inference control based on the above heuristics that can be used by the reasoning system itself (as opposed to by the programmer) in making inference control decisions. The main concern in AQUA has been the formulation and indexing of questions, and their use in focussing AQUA's explanation, understanding and learning processes.

Mechanisms for question management

Maintaining a collection of explicit knowledge goals, represented in AQUA as questions, introduces new issues into the design of AI programs. The goals themselves must be organized, applied, and disposed of when no longer useful. These management tasks were addressed by the following general mechanisms which were required in AQUA:

- **Question retrieval**: finding suspended questions that a new piece of knowledge might satisfy.
- **Question indexing**: storing questions in memory so that they are found almost only when they are relevant.
- **Process scheduling**: restarting suspended tasks that depend on questions when the questions are answered.
- **Hypothesis management**: deleting alternative questions and hypotheses when a question is answered, because their likelihood of being useful decreases since an alternative has been found.

Representation of questions

Question representations have two parts:

1. **Concept specification**: the object of the question, *i.e.*, the desired information. This is represented using a memory structure that specifies what would be minimally acceptable as an answer to the question. In general, a question may seek to acquire new knowledge or to reorganize or elaborate existing knowledge. A new piece of knowledge is an answer to a question if it matches the concept specification completely. The answer could specify more than the question required, of course.
2. **Task specification**: what to do with the information once it comes in, which depends on why the question was generated. This may be represented either as a procedure to be run, or as a declarative specification of the suspended task. When the question is answered, either because the program actively pursued it, or opportunistically while it was processing something else, the suspended process that depends on that information is restarted.

In a sense, a question is similar to an open “slot” in a memory structure. AQUA's initial processing could be viewed as being similar to that of typical “script-based” understanders: Words in the input text are used to instantiate memory structures, and open slots in these memory structures are used as predictions for the rest of the story. However, there are three main differences (expressed here in a "slot-filling" terminology for comparison):

1. Typically, all open slots in newly instantiated structures are used as "requests" or "predictions" and cause the understander to look for fillers for those slots. AQUA, however, uses its interestingness heuristics to mark interesting slots to be used as predictions or questions. In addition, slots can be marked as being interesting by understanding tasks when their values are needed but not yet known.
2. Open slots arise not only from script-like knowledge structures, but also from causal or explanatory structures.
3. Typically, the ultimate task of the understander is to fill in as many of these open slots as possible. However, the action of filling in a slot does not do anything more than provide a value for that slot. In AQUA, however, slots are not filled for their own sake, but rather for the sake of performing some kind of reasoning with that value (*e.g.*, confirming a hypothesis) which may go beyond simple fact-gathering.

Thus AQUA subscribes to the basic slot-filling idea but extends this idea by selecting which slots are worth filling, by using different kinds of knowledge structures to provide slots, and by remembering why particular slots need to be filled so that it can use the filled values when they become known. The uniform representation of questions generated and used by different types of reasoning and learning processes allows us to design an integrated system in an easy and natural manner.

When a question is posed, AQUA searches its memory for a knowledge structure that matches the concept specification of the question. If one is found, the question is immediately answered; if not, the question is indexed in memory and the task is suspended until an answer is found. An answer to a question is a node which matches the concept specification of the question and provides all the information required by the concept specification. The answer node may be created to represent new information provided by the story, or internally generated through inference during other processing. When a question is answered, the answer node is merged with the concept specification, and the task associated with it is run.

Indexing questions

Where should a question be placed in memory? Since a potential answer to a question may arrive at any time, particularly when the question may not even be "active," the question must be indexed in memory exactly where the answer would be placed when it does come in. This ensures that the question will be found without extensive searching through lists of questions. The issue of the amount of inference that should be done at this point was addressed earlier.

In AQUA, questions are indexed in memory on the basis of their concept specifications. For the task of story understanding, these questions are used to generate expectations that guide the system when the concepts to which they are attached are active.

Retrieving questions

When a new fact becomes known, either because it is part of the input (*e.g.*, it is read in the story), or because it is inferred for some other reason, the reasoner needs to retrieve questions in memory that the fact could be relevant to. The questions retrieved in turn determine how useful that fact is. AQUA's question retrieval strategies take advantage of the fact that questions are indexed on the basis of their concept specifications in an inheritance hierarchy. AQUA uses three question retrieval strategies:

Type retrieval: When a new memory structure is activated, questions indexed off the types of the concept are retrieved. The new structure is matched against the concept specification of the question to see whether it provides the desired information. For example, if AQUA reads about a car, it retrieves questions off the “car” concept to see if the car it read about could answer any of these questions.

Relation retrieval: AQUA uses a frame-based representational scheme in which slots and slot fillers specify relations between concepts. For example, the results slot specifies a causal relation of a particular kind between an action and a state. Similarly, the actor slot in an action frame specifies a participatory relation between the action and a volitional-agent. Relations are themselves represented as frames in memory (*e.g.*, see [Wilensky, 1986]), allowing AQUA to reason about the relations themselves. Questions seeking relations between concepts are indexed in the appropriate slots in the frames representing these concepts. This allows AQUA to retrieve questions that seek relations between memory structures (*e.g.*, the connection between a given terrorist attack and the destruction of some building).

Specialization retrieval: Finally, questions may be retrieved, given an input cue, by checking whether some specialization or refinement of that input might address a questions. This allows the understanding process to be sensitive to the questions that the system is currently seeking answers to. Implementational details may be found in Ram [1989].

Question-driven learning

As argued by Hammond [1986] and others, a theory of case-based reasoning must include a theory of learning as well. Traditional case-based learning programs learn new cases by relying on existing cases that are "well understood" to guide them through novel experiences. AQUA extends this idea by not requiring that existing cases be well understood. When a story is understood, for example, AQUA may be left with several unanswered questions that are part of its representation of the story. Much of the learning in AQUA occurs through the answering of previous questions, which leads to the elaboration, modification, or re-indexing of existing knowledge structures (in AQUA's case, XPs) to which these questions are attached. Details of the learning methods in AQUA may be found in Ram [1993]. As AQUA reads, it asks better and more detailed questions about input stories, formulates knowledge goals to answer these question, and learns when its knowledge goals are satisfied (perhaps in a later story). This results in a gradual improvement in AQUA's explanatory knowledge and hence in its ability to explain. Figure 3 shows an example of the question transformation process.

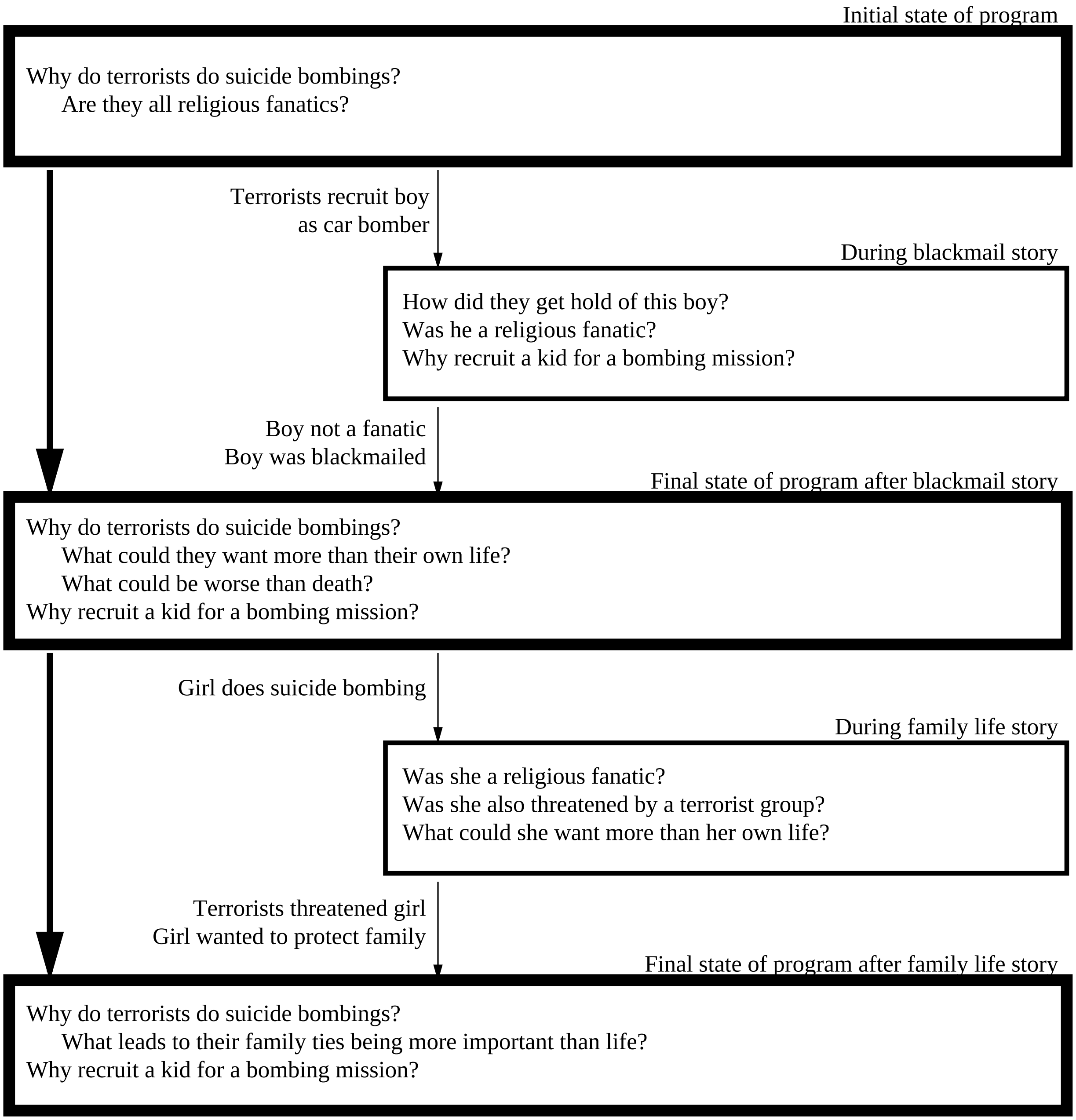
Initial state of program
Why do terrorists do suicide bombings?
Are they all religious fanatics?
Terrorists recruit boy as car bomber
During blackmail story
How did they get hold of this boy?
Was he a religious fanatic?
Why recruit a kid for a bombing mission?
Boy not a fanatic
Boy was blackmailed
Final state of program after blackmail story
Why do terrorists do suicide bombings?
What could they want more than their own life?
What could be worse than death?
Why recruit a kid for a bombing mission?
Girl does suicide bombing
During family life story
Was she a religious fanatic?
Was she also threatened by a terrorist group?
What could she want more than her own life?
Terrorists threatened girl
Girl wanted to protect family
Final state of program after family life story
Why do terrorists do suicide bombings?
What leads to their family ties being more important than life?
Why recruit a kid for a bombing mission?

Figure 3: Question-driven understanding is a process of asking questions and trying to answer them by reading a story. AQUA starts out with a set of questions. As it reads, some of these questions are answered and new questions are raised. After reading the story, AQUA is left with a set of new questions that are the starting point for reading future stories. Here, AQUA has read two stories, one about a boy being blackmailed into going on a suicide bombing mission in which no further details are given, and another about a girl being "persuaded" to commit a suicidal terrorist attack by a terrorist group who threatened her family.

Conclusions

The underlying theme of this research is a focus on the learning goals of the reasoner. In particular, we are developing a theory of knowledge goals, which represent the goals of a reasoner to learn by acquiring new knowledge or reorganizing existing knowledge by learning new indexing structures [Ram, 1991, 1993] or, in general, by reformulating its knowledge in other ways [Ram and Cox, 1993; Leake and Ram, 1993]. Knowledge goals arise from gaps in the reasoner's knowledge which are identified when the reasoner encounters difficulties during processing. Since knowledge goals are often voiced out loud in the form of questions, we have used questions as a device to model goal-driven explanation, understanding and learning processes.

This chapter focussed on the process of question-driven explanation, and on the questions that arise from, and support, the processes involved in explanation. In general, there are several types of knowledge goals that might arise out of difficulties during processing, and different types of learning that correspond to these knowledge goals. We are developing learning algorithms that deal with different types of processing failures, and investigating the extent to which these learning algorithms can be integrated into a single multistrategy learning system. We are currently implementing the Meta-AQUA system [Ram and Cox, 1993], an extension of AQUA which can use multiple explanation strategies to build explanations while reading a story and multiple learning algorithms to learn from the different types of problems and questions that arise during this process.

References

[Barsalou, 1991] L. Barsalou. Deriving Categories to Achieve Goals. In G.H. Bower, editor, *The Psychology of Learning and Motivation: Advances in Research and Theory,* Volume 27, Academic Press, New York, NY.

[Carbonell, 1979] J.G. Carbonell. *Subjective Understanding: Computer Models of Belief Systems*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1979. Research Report #150.

[Charniak, 1986] E. Charniak. A Neat Theory of Marker Passing. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 584--588, Philadelphia, PA, 1986.

[Domeshek, 1992] E.A. Domeshek. *Do the Right Thing: A Component Theory for Indexing Stories as Social Advice*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1992.

[Dehn, 1989] N. Dehn. *Computer Story Writing: The Role of Reconstructive and Dynamic Memory*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1989. Research Report #792.

[DeJong, 1983] G.F. DeJong. An Approach to Learning from Observation. In R.S. Michalski, editor, *Proceedings of the 1983 International Machine Learning Workshop*, pages 171--176, Monticello, IL, 1983.

[Graesser, Person, and Huber, 1992] A.C. Graesser, N. Person, and J. Huber. Mechanisms that Generate Questions. In T.W. Lauer, E. Peacock, and A.C. Graesser, editors, *Questions and Information Systems*, pages 167–187, Lawrence Erlbaum Associates, Hillsdale, NJ, 1992.

[Hammond, 1986] K.J. Hammond. *Case-Based Planning: An Integrated Theory of Planning, Learning and Memory*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1986. Research Report #488.

[Hobbs, Stickel, Appelt, and Martin, 1990] J.R. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as Abduction. Technical Note 499, SRI International, 1990.

[Hunter, 1989] L.E. Hunter. *Knowledge Acquisition Planning: Gaining Expertise Through Experience*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1989. Research Report #678.

[Jones, 1992] E.K. Jones. *The Flexible Uses of Abstract Knowledge in Planning*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1992.

[Kass, Leake, and Owens, 1986] A. Kass, D. Leake, and C. Owens. SWALE: A Program That Explains. In [Schank, 1986], pages 232--254,1986.

[Kass, 1990] A. Kass. *Developing Creative Hypotheses by Adapting Explanations*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1990.

[Kautz and Allen, 1986] H.A. Kautz and J.F. Allen. Generalized Plan Recognition. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, Philadelphia, PA, 1986.

[Konolige, 1990] K. Konolige. A General Theory of Abduction. In *Proceedings of the AAAI Spring Symposium on Automated Abduction*, Stanford, CA, 1990.

[Leake and Ram, 1993] D. Leake and A. Ram. Goal-Driven Learning: Fundamental Issues and Symposium Report. *AI Magazine*, 14(4), 1993, in press.

[Leake, 1989a] D. Leake. Anomaly detection strategies for schema-based story understanding. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pages 490--497, Ann Arbor, MI, 1989.

[Leake, 1989b] D. Leake. *Evaluating Explanations*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1989.

[Lehnert, 1978] W.G. Lehnert. *The Process of Question Answering*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1978.

[Mooney and DeJong, 1985] R.J. Mooney and G.F. DeJong. Learning Schemata for Natural Language Processing. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 681--687, Los Angeles, CA, 1985.

[Morris and O'Rorke, 1990] S. Morris and P. O'Rorke. An Approach to Theory Revision using Abduction. In *Proceedings of the AAAI Spring Symposium on Automated Abduction*, 1990.

[Ng and Bereiter, 1991] E. Ng and C. Bereiter. Three Levels of Goal Orientation in Learning. *The Journal of the Learning Sciences*, 1(3&4):243--271, 1991.

[Ng and Mooney, 1990] H. Ng and R. Mooney. On the Role of Coherence in Abductive Explanation. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 337--342, Boston, MA, 1990.

[Owens, 1991] C. Owens. A Functional Taxonomy of Abstract Plan Failures. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 167–172, Chicago, IL.

[Ram and Cox, 1993] A. Ram and M.T. Cox. Introspective Reasoning using Meta-Explanations for Multistrategy Learning. In R.S. Michalski and G. Tecuci, editors, *Machine Learning IV: A Multistrategy Approach*. Morgan Kaufman Publishers, 1993, in press.

[Ram and Hunter, 1992] A. Ram and L. Hunter. The Use of Explicit Goals for Knowledge to Guide Inference and Learning. *Applied Intelligence*, 2(1):47–73, 1992.

[Ram and Leake, 1991] A. Ram and D. Leake. Evaluation of Explanatory Hypotheses. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 867--871, Chicago, IL, 1991.

[Ram, 1984] A. Ram. *Modelling Characters and their Decisions: A Theory of Compliance Decisions*. M.S. thesis, University of Illinois at Urbana-Champaign, Urbana, IL, 1984. Technical Report T-145, Coordinated Science Laboratory.

[Ram, 1989] A. Ram. *Question-Driven Understanding: An Integrated Theory of Story Understanding, Memory and Learning*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1989. Research Report #710.

[Ram, 1990a] A. Ram. Decision Models: A Theory of Volitional Explanation. In *Proceedings of the Twelvth Annual Conference of the Cognitive Science Society*, pages 198--205, Cambridge, MA, 1990.

Ram, 1990b] A. Ram. Incremental Learning of Explanation Patterns and their Indices. In B.W. Porter and R.J. Mooney, editors, *Proceedings of the Seventh International Conference on Machine Learning*, pages 313--320, Austin, TX, 1990.

[Ram, 1990c] A. Ram. Knowledge Goals: A Theory of Interestingness. In *Proceedings of the Twelvth Annual Conference of the Cognitive Science Society*, pages 206--214, Cambridge, MA, 1990.

[Ram, 1991] A. Ram. A Theory of Questions and Question Asking. *The Journal of the Learning Sciences*, 1(3&4):273--318, 1991.

[Ram, 1993] A. Ram. Indexing, Elaboration and Refinement: Incremental Learning of Explanatory Cases. *Machine Learning*, 10(3):201–248, 1993.

[Rieger, 1975] C. Rieger. Conceptual Memory and Inference. In R.C. Schank, editor, *Conceptual Information Processing*. North-Holland, Amsterdam, 1975.

[Scardamalia and Bereiter, 1991] M. Scardamalia and C. Bereiter. Higher Levels of Agency for Children in Knowledge Building: A Challenge for the Design of New Knowledge Media. *The Journal of the Learning Sciences*, 1(1):37--68, 1991.

[Schank and Abelson, 1977] R.C. Schank and R. Abelson. *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1977.

[Schank and Ram, 1988] R. C. Schank and A. Ram. Question-driven Parsing: A New Approach to Natural Language Understanding. *Journal of Japanese Society for Artificial Intelligence*, 3(3):260--270, 1988.

[Schank, 1982] R. C. Schank. *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge University Press, New York, NY, 1982.

[Schank, 1986] R. C. Schank. *Explanation Patterns: Understanding Mechanically and Creatively*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Steinbart, 1992] P.J. Steinbart. The Role of Questioning in Learning from Computer-Based Decision Aids. In T.W. Lauer, E. Peacock, and A.C. Graesser, *Questions and Information Systems*, pages 273–285, Lawrence Erlbaum Associates, Hillsdale, NJ, 1992.

[Stickel, 1990] M.E. Stickel. A Method For Abductive Reasoning In Natural-Language Interpretation. In *Proceedings Of The AAAI Spring Symposium On Automated Abduction*, 1990.

[Thagard, 1989] P. Thagard. Explanatory Coherence. *Behavioral and Brain Sciences*, 12(3):435--502, 1989.

[Wilensky, 1978] R. Wilensky. *Understanding Goal-Based Stories*. Ph.D. thesis, Yale University, Department of Computer Science, New Haven, CT, 1978.

[Wilensky, 1981] R. Wilensky. PAM. In R. Schank and C.K. Riesbeck, editors, *Inside Computer Understanding: Five Programs plus Miniatures*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1981.

[Wilensky, 1983] R. Wilensky. *Planning and Understanding*. Addison-Wesley, Reading, MA, 1983.

[Wilensky, 1986] R. Wilensky. Knowledge Representation --- A Critique and a Proposal. In J.L. Kolodner and C.K. Riesbeck, editors, *Experience, Memory and Reasoning*, chapter 2, pages 15--28. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.

[Zukier, 1986] H. Zukier. The Paradigmatic and Narrative Modes in Goal-Guided Inference. In R. Sorrentino and E. Higgins, editors, *Handbook of Motivation and Cognition: Foundations of Social Behavior*, pages 465--502. Guilford Press, Guilford, CT, 1986.