Rapid Argumentation Capture from Analysis Reports: The Case Study of Aum Shinrikyo

Mihai Boicu, Gheorghe Tecuci, Dorin Marcu
Learning Agents Center, Volgenau School of Engineering, George Mason University, Fairfax, VA 22030

Abstract— **The availability of subject matter experts has always been a challenge for the development of knowledge-based cognitive assistants incorporating their expertise. This paper presents an approach to rapidly develop cognitive assistants for evidence-based reasoning by capturing and operationalizing the expertise that was already documented in analysis reports. It illustrates the approach with the development of a cognitive assistant for assessing whether a terrorist organization is pursuing weapons of mass destruction, based on a report on the strategies followed by Aum Shinrikyo to develop and use biological and chemical weapons.**

Knowledge engineering, learning agent shell for evidence-based reasoning, problem reduction and solution synthesis, agent teaching and learning, intelligence analysis, cognitive assitant, argumentation, weapons of mass destruction.

I. Introduction

We research advanced knowledge engineering methods for rapid development of agents that incorporate the knowledge of human experts to assist their users in complex problem solving and to teach students. The development of such systems by knowledge engineers and subject matter experts is very complex due to the difficulty of capturing and representing experts' problem solving knowledge.

Our approach to this challenge was to develop multistrategy learning methods enabling a subject matter expert who is not a knowledge engineer to train a learning agent through problem solving examples and explanations, in a way that is similar to how the expert would train a student. This has led to the development of a new type of tool for agent development which we have called *learning agent shell* [1]. The learning agent shell is a refinement of the concept of *expert system shell* [2]. As an expert system shell, the learning agent shell includes a general inference engine for a knowledge base to be developed by capturing knowledge from a subject matter expert. The inference engine of the learning agent shell, however, is based on a general divide-and-conquer approach to problem solving, called problem reduction and solution synthesis, which is very natural for a non-technical subject matter expert, facilitates agent teaching and learning, and is computationally efficient. Moreover, in order to facilitate knowledge reuse, the knowledge base of the learning agent shell is structured into an ontology of concepts and a set of problem solving rules expressed with these concepts. The ontology is the more general part of the knowledge base and is usually relevant to many applications in the same domain, such as military or medicine. Indeed, many military applications will require reasoning with concepts such as military unit or military equipment. Thus, when developing a knowledge-based agent for a new military application, one may expect to be able to reuse a significant part of the ontology of a previously developed agent. The reasoning rules, however, are much more application-specific, such as the rules for critiquing a course of action with respect to the principles of war versus the rules for determining the strategic center of gravity of a force. Therefore the rules are reused to a much lesser extent. To facilitate their acquisition, the learning agent shell includes a multistrategy learning engine, enabling the learning of the rules directly from the subject matter expert, as mentioned above.

We have developed increasingly more capable and easier to use learning agent shells and we have applied them to build knowledge-based agents for various applications, including military engineering planning, course of action critiquing, and center of gravity determination [3].

Investigating the development of cognitive assistants for intelligence analysis, such as Disciple LTA [4] and TIACRITIS [5], has led us to the development of a new type of agent development tool, called *learning agent shell for evidence-based reasoning* [6]. This new tool extends a learning agent shell with generic modules for representation, search, and reasoning with evidence. It also includes a hierarchy of knowledge bases, the top of which is a domain-independent knowledge base for evidence-based reasoning containing an ontology of evidence and general rules, such as the rules for assessing the believability of different items of evidence [7]. This knowledge base is very significant because it is applicable to evidence-based reasoning tasks across various domains, such as intelligence analysis, law, forensics, medicine, physics, history, and others. An example of a *learning agent shell for evidence-based reasoning* is Disciple-EBR [6].

The development of a knowledge-based agent for an evidence-based reasoning task, such as intelligence analysis, is simplified because the shell already has general knowledge for evidence-based reasoning. Thus one only needs to develop the domain-specific part of the knowledge base. However, we still face the difficult problem of having access to subject matter experts who can dedicate their time to teach the agent. This paper presents a solution to this problem. It happens that there are many reports written by subject matter experts which already contain significant problem solving expertise. Thus, rather than eliciting the expertise directly from these experts, a junior professional may capture it from their reports.

We will illustrate this approach by considering a recent

This research was partially supported by the National Geospatial-Intelligence Agency and by George Mason University. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the National Geospatial-Intelligence Agency or the U.S. Government.

report from the Center for a New American Security, "*Aum Shinrikyo: Insights Into How Terrorists Develop Biological and Chemical Weapons*" [8]. This report provides a comprehensive analysis of this terrorist group, its radicalization, and the strategies followed in the development and use of biological and chemical weapons. As stated by its authors: "… this is the most accessible and informative opportunity to study terrorist efforts to develop biological and chemical weapons" [8, p.33]. "This detailed case study of Aum Shinrikyo (Aum) suggests several lessons for understanding attempts by other terrorist groups to acquire chemical or biological weapons" [8, p.4]. "Our aim is to have this study enrich policymakers' and intelligence agencies' understanding when they assess the risks that terrorists may develop and use weapons of mass destruction" [8, p.6].

Indeed, this report presents in detail two examples of how a terrorist group has pursued weapons of mass destruction, one where it was successful (sarin-based chemical weapons), and one where it was not successful (B-anthracis-based biological weapons). We will show how we can use these examples to train Disciple-EBR, evolving it into a cognitive assistant that will help intelligence analysts in assessing whether other terrorist groups may be pursuing weapons of mass destruction. Notice that this process operationalizes the knowledge from the report to facilitate its application in new situations.

We first present a brief summary of the Aum report. Then we explain the process of evidence-based hypothesis analysis using problem reduction and solution synthesis. Finally we present the actual development of the cognitive assistant.

II. Aum Shinrikyo: Insights Into How Terrorists Develop Biological and Chemical Weapons [8]

The first section of the report describes the creation of the Aum cult by Chizuo Matsumoto in 1984 as a yoga school. Soon after that Aum started to develop a religious doctrine and to create monastic communities. From the beginning the cult was apocalyptic, believing in an imminent catastrophe that can be prevented only by positive spiritual action. In 1988, the cult started to apply physical force and punishments toward its members to purify the body, and started to commit illegalities.

The second section of the report analyzes the biological weapons program. The cult first tried to obtain botulinum toxin, but it failed to obtain a deadly strain. However, the cult released the toxin in 20 to 40 attacks in which, luckily, nobody died. Possible causes of the failure were identified as ineffective initial strain of C. botulinum, unsuitable culture conditions, unsterile conditions, wrong post-fermentation recovery, and improper storage conditions. Similarly, the anthrax program and its failure are analyzed.

The third section of the report analyzes the chemical weapons program. While other chemical agents were tested during the program, the main part of the program was based on sarin. Although the program had some problems with mass production, it was generally successful, and produced large quantities of sarin at various levels of purity. Aum performed several attacks with sarin, including: (1) an ineffective attack on a competing religious leader in 1993; (2) an attack, in June 1994, with a vaporization of sarin, intended to kill several judges – the vapors were shifted toward a neighborhood, killing 8 persons and injuring 200; (3) several attacks in the Tokyo Subway on 20 March 1995, killing 13 and injuring thousands.

The fourth section of the report summarizes the main lessons learned: (1) chemical weapons capabilities seem more accessible than biological capabilities for mass killing; (2) effective dissemination is challenging; (3) recurred accidents in the programs did not deter their pursuit; (4) during the transition to violence some leaders joined while others were isolated or killed; (5) law enforcement pressure was highly disruptive even though it was not an effective deterrent; (6) the programs and attacks were conducted by the leadership group only, to maintain secrecy; (7) the hierarchical structure of the cult facilitated the initiation and resourcing of the programs but distorted their development and assessment; (8) contemporaneous assessment of the intentions and capabilities of a terrorist organization are difficult, uncertain and even misleading; (9) despite many mistakes and failures, successes were obtained as a result of the persistence in the programs.

III. Hypothesis Analysis with Disciple-EBR

A class of hypothesis analysis problems is represented in Disciple-EBR as the 7-tuple (O, P, S, Rr, Sr, I, E), as shown in Figure 1. The ontology O is a hierarchical representation of both general and domain-specific concepts and relationships. The general (domain-independent) concepts are primarily those for evidence-based reasoning, such as different types of evidence. The two primary roles of the ontology are to support the representation of the other knowledge elements (e.g. the reasoning rules), and to serve as the generalization hierarchy for learning. The hypothesis analysis problems P and the corresponding solutions S are natural language patterns with variables. They include first-order logic applicability conditions that restrict the possible values of the variables.

A problem reduction rule Rr expresses how and under what conditions a generic hypothesis analysis problem P_g can be reduced to simpler generic problems. These conditions are represented as first-order logical expressions. Similarly, a

The Disciple representation of a class of hypothesis analysis problems is a 7-tuple (O, P, S, Rr, Sr, I, E) where:

O –	ontology of domain concepts and relationships;
P –	class of hypothesis analysis problems;
S –	solutions of problems;
Rr –	problem reduction rules that reduce problems to sub-problems and/or solutions;
Sr –	solution synthesis rules that synthesize the solution of a problem from the solutions of its sub-problems.
I –	Instances of the concepts from O, with properties and relationships;
E –	evidence for assessing hypothesis analysis problems.

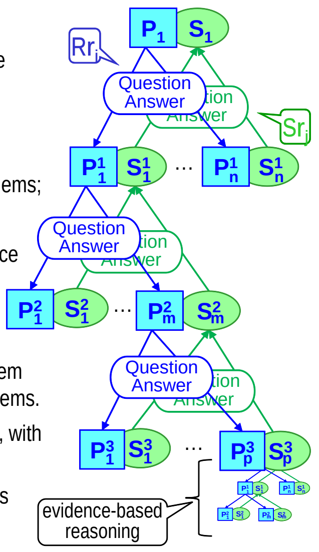


Figure 1. Disciple representation of a class of hypothesis analysis problems.

solution synthesis rule Sr expresses how and under what conditions generic probabilistic solutions can be combined into another probabilistic solution [9]. As mentioned, Disciple-EBR already contains domain-independent problem reduction and solution synthesis rules for evidence-based reasoning.

Disciple-EBR employs a general divide-and-conquer approach to solve a hypothesis analysis problem. For example, as illustrated in the right-hand side of Figure 1, a complex problem P_1 is reduced to n simpler problems P^1_1, … , P^1_n, through the application of the reduction rule Rr_i. If we can then find the solutions S^1_1, … , S^1_n of these sub-problems, then these solutions can be combined into the solution S_1 of the problem P_1, through the application of the synthesis rule Sr_j. The Question/Answer pairs associated with these reduction and synthesis operations express, in natural language, the applicability conditions of the corresponding reduction and synthesis rules, in this particular situation. Their role will be discussed in more detail in the next section.

Specific examples of reasoning trees are shown in Figures 5, 6, and 11, which will be discussed in the next section. In general, a top-level hypothesis analysis problem is successively reduced (guided by questions and answers) to simpler and simpler problems, down to the level of elementary problems that are solved based on knowledge and evidence. Then the obtained solutions are successively combined, from bottom-up, to obtain the solution of the top-level problem.

Figure 2 presents the reduction and synthesis operations in more detail. To assess hypothesis H_1 one asks the question Q which happens to have two answers, A and B. For example, a question like "Which is an indicator for H_1?" may have many answers, while other questions have only one answer. Let's assume that answer A leads to the reduction of H_1 to the simpler hypotheses H_2 and H_3, and answer B leads to the reduction of H_1 to H_4 and H_5. Let us further assume that we have assessed the likeliness of each of these four sub-hypotheses, as indicated at the bottom part of Figure 2. The likeliness of H_2 needs to be combined with the likeliness of H_3, to obtain a partial assessment (corresponding to the answer A) of the likeliness of H_1. One similarly obtains another partial assessment (corresponding to the answer B) of the likeliness of H_1. Then the likeliness of H_1 corresponding to the answer A needs to be combined with the likeliness of H_1 corresponding to the answer B, to obtain the likeliness of H_1 corresponding to all the answers of question Q (e.g., corresponding to all the indicators).

We call the two bottom-level syntheses in Figure 2 *reduction-level syntheses* because they correspond to reductions of H_1 to simpler hypotheses. We call the top-level synthesis *problem-level synthesis* because it corresponds to all the known strategies for solving the problem.

The likeliness may be expressed using symbolic probability values that are similar to those used in the U.S. National Intelligence Council's standard estimative language: {no possibility, a remote possibility, very unlikely, unlikely, an even chance, likely, very likely, almost certain, certain}. However, other symbolic probabilities may also be used, as discussed by Kent [10] and Weiss [11]. In these cases one may use simple synthesis functions, such as, min, max, average, or

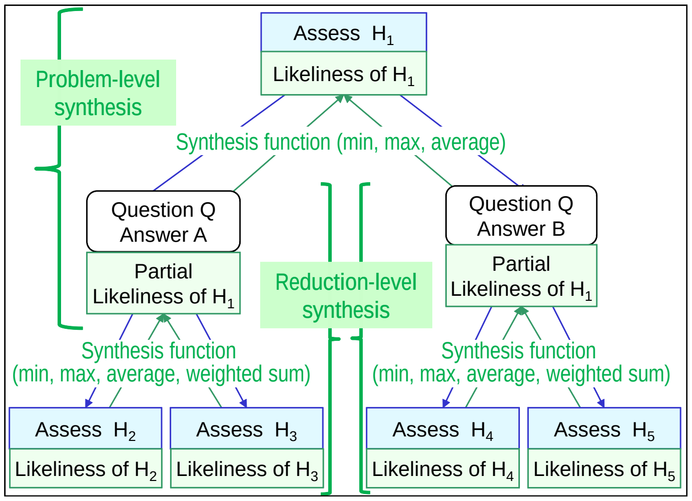

Assess H_1
Likeliness of H_1
Problem-level synthesis
Synthesis function (min, max, average)
Question Q Answer A
Partial Likeliness of H_1
Reduction-level synthesis
Question Q Answer B
Partial Likeliness of H_1
Synthesis function (min, max, average, weighted sum)
Synthesis function (min, max, average, weighted sum)
Assess H_2
Likeliness of H_2
Assess H_3
Likeliness of H_3
Assess H_4
Likeliness of H_4
Assess H_5
Likeliness of H_5

Figure 2. Hypothesis assessment through reduction and synthesis.

weighted sum, as shown in Figure 2 and Figure 3 [12].

As indicated above, Disciple-EBR includes general reduction and synthesis rules for evidence-based reasoning which allow it to automatically generate fragments of the reduction and synthesis tree, like the one from Figure 3. In this case the problem is to assess hypothesis H_1 based on favoring evidence. Which is a favoring item of evidence? If E_1 is such an item, then Disciple reduces the top level assessment to two simpler assessments: "Assess the relevance of E_1 to H_1" and "Assess the believability of E_1". If E_2 is another relevant item of evidence, then Disciple reduces the top level assessment to two other simpler assessments. Obviously there may be any number of favoring items of evidence.

Now let us assume that Disciple has obtained the solutions of the leaf problems, as shown at the bottom of Figure 3 (e.g., "If we assume that E_1 is believable, then H_1 is very likely to be true." "The believability of E_1 is likely.") Notice that what is really of interest in a solution is the actual likeliness value. Therefore, an expression like "The believability of E_1 is likely" can be abstracted to "likely." Consequently, the reasoning tree in Figure 3 shows only these abstracted solutions, although, internally, the complete solution expressions are maintained.

Having obtained the solutions of the leaf hypotheses in Figure 3, Disciple automatically combines them to obtain the likeliness of the top level hypothesis. First it assesses the

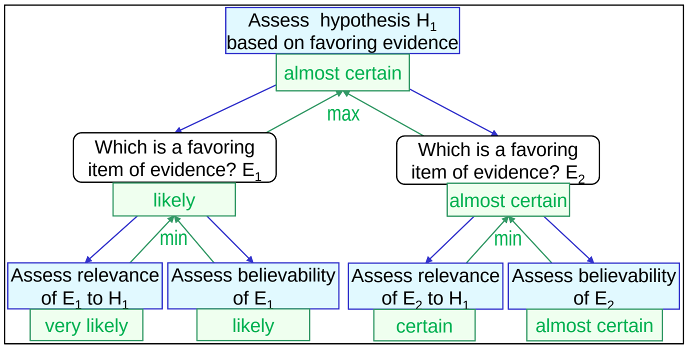

Assess hypothesis H_1 based on favoring evidence
almost certain
max
Which is a favoring item of evidence? E_1
likely
min
Which is a favoring item of evidence? E_2
almost certain
min
Assess relevance of E_1 to H_1
very likely
Assess believability of E_1
likely
Assess relevance of E_2 to H_1
certain
Assess believability of E_2
almost certain

Figure 3. Automated hypothesis assessment through reduction and synthesis.

inferential force of each item of favoring evidence (i.e., E_1 and E_2) on H_1 by taking the min between its *relevance* and its *believability*, because only evidence that is both relevant and believable will convince us that a hypothesis is true. Next Disciple assesses the inferential force of the favoring evidence as the max of the inferential force corresponding to individual items of evidence because it is enough to have one relevant and believable item of evidence to convince us that the hypothesis H_1 is true. Disciple will similarly consider disfavoring items of evidence, and will use an on balance judgment to determine the inferential force of all available evidence on H_1.

To facilitate the browsing and understanding of larger reasoning trees, Disciple also displays them in abstracted (simplified) form, as illustrated in the bottom right side of Figure 5. The top-level abstract problem "start with chaos and destruction" is the abstraction of the problem "Assess whether Aum Shinrikyo preaches that the apocalypse will start with chaos and destruction" from the bottom of Figure 5. The abstract sub-problem "favoring evidence" is the abstraction of "Assess whether Aum Shinrikyo preaches that the apocalypse will start with chaos and destruction, based on favoring evidence." This is a specific instance of the problem from the top of Figure 3 which is solved as discussed above. The user assessed the relevance and the believability of the two items of evidence EVD-013 and EVD-014, and Disciple automatically determined and combined their inferential force on the higher-level hypotheses.

IV. Agent Development Methodology

Figure 4 presents the main stages of evolving the Disciple-EBR agent shell into a specific cognitive assistant for hypotheses analysis. The first stage is system specification during which a knowledge engineer and a subject matter expert define the types of problems to be solved by the system. Then they rapidly develop a prototype, first by developing a model of how to solve a problem, and then by applying the model to solve typical problems. During the next phase they use the developed sample reasoning trees to develop a specification of the system's ontology and use that specification to design and develop an ontology of concepts and relationships which is as complete as possible. Finally they use the system to learn and refine reasoning rules, which may also require the extension of the ontology.

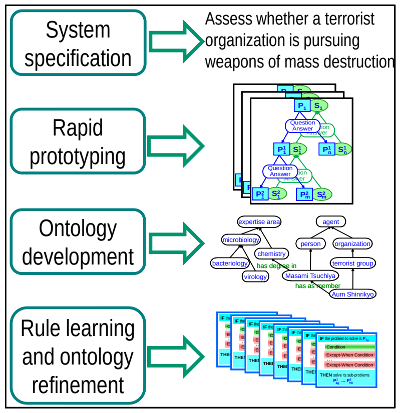
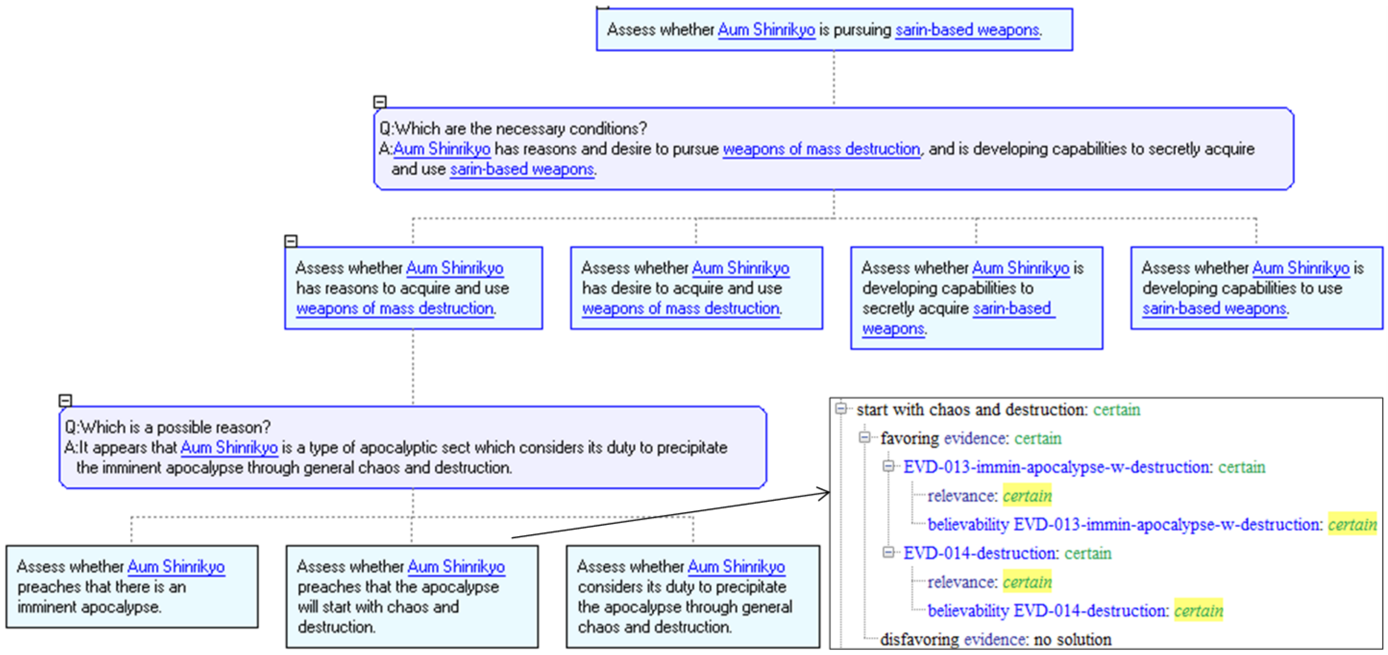


Figure 4. Main agent development stages.

In the next section we will illustrate the development of a cognitive assistant that will help assess whether a terrorist organization is pursuing weapons of mass destruction. The main difference from the above methodology is that we capture the expertise not from a subject matter expert, but from the Aum report [8].

V. Capturing the Expertise from the Aum Report

The Aum report presents in detail two examples of how a terrorist group has pursued weapons of mass destruction. We will briefly illustrate the process of teaching Disciple-EBR based on these examples, enabling it to assist other analysts in assessing whether a terrorist group may be pursuing weapons of mass destruction. For this, we need to frame each of these examples as a problem solving experience imagining, for instance, that we are attempting to solve the following hypothesis analysis problem:


Assess whether Aum Shinrikyo is pursuing sarin-based weapons.

Q:Which are the necessary conditions?
A:Aum Shinrikyo has reasons and desire to pursue weapons of mass destruction, and is developing capabilities to secretly acquire and use sarin-based weapons.

Assess whether Aum Shinrikyo has reasons to acquire and use weapons of mass destruction.

Assess whether Aum Shinrikyo has desire to acquire and use weapons of mass destruction.

Assess whether Aum Shinrikyo is developing capabilities to secretly acquire sarin-based weapons.

Assess whether Aum Shinrikyo is developing capabilities to use sarin-based weapons.

Q:Which is a possible reason?
A:It appears that Aum Shinrikyo is a type of apocalyptic sect which considers its duty to precipitate the imminent apocalypse through general chaos and destruction.

Assess whether Aum Shinrikyo preaches that there is an imminent apocalypse.

Assess whether Aum Shinrikyo preaches that the apocalypse will start with chaos and destruction.

Assess whether Aum Shinrikyo considers its duty to precipitate the apocalypse through general chaos and destruction.

start with chaos and destruction: certain
favoring evidence: certain
EVD-013-immin-apocalypse-w-destruction: certain
relevance: certain
believability EVD-013-immin-apocalypse-w-destruction: certain
EVD-014-destruction: certain
relevance: certain
believability EVD-014-destruction: certain
disfavoring evidence: no solution

Figure 5. Detailed and abstract fragments of the hypothesis analysis tree.

Assess whether Aum Shinrikyo is pursuing sarin-based weapons.

We express the problem in natural language and select the phrases that may be different for other problems. The selected phrases will appear in blue, guiding the system to learn a general problem pattern:

Assess whether ?O1 is pursuing ?O2.

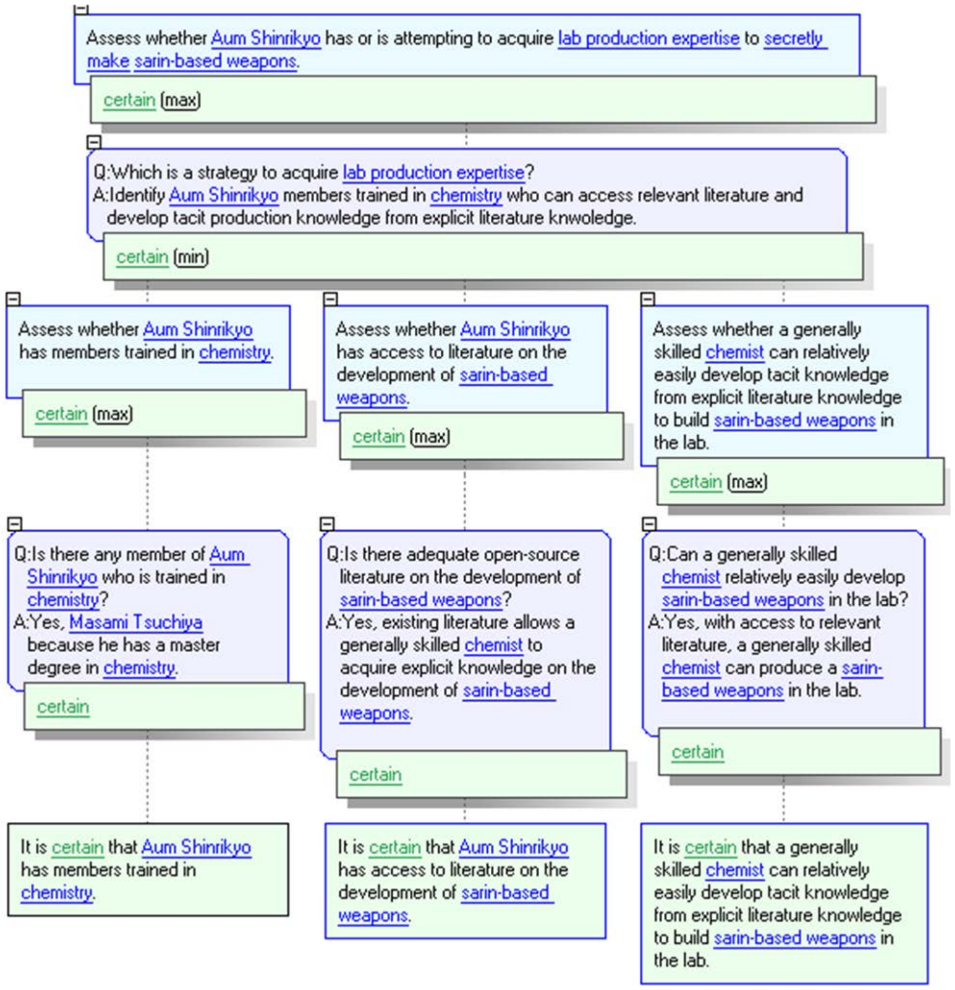
Then we show Disciple how to solve the hypothesis analysis problem based on the knowledge and evidence provided in the Aum report. The modeling module of Disciple-EBR guides us in developing a reasoning tree like the one from the right hand side of Figure 1. The top part of this tree is shown in Figure 5.

The main goal of this stage is to develop a formal, yet intuitive argumentation structure [12-15], representing the assessment logic as inquiry-driven problem reduction and solution synthesis. Notice that, guided by a question-answer pair, we reduce the top-level hypothesis assessment problem to four sub-problems. We then reduce the first sub-problem to three simpler problems which we declare as elementary hypotheses, to be assessed based on evidence. Once we associate items of evidence from the Aum report with such an elementary hypothesis, Disciple automatically develops a reduction tree. For example, we have associated two items of favoring evidence with the second leaf-problem and Disciple has generated the reasoning tree whose abstraction is shown in the bottom-right of Figure 5. After we have assessed the relevance and the believability of each item, Disciple has automatically computed the inferential force and the likeliness of the upper level hypotheses, concluding: "It is certain that Aum Shinrykio preaches that the apocalypse will start with chaos and destruction."

The other hypothesis analysis problems are reduced in a similar way, either to elementary hypotheses assessed based on evidence, or directly to solutions. For example, based on the information from the Aum report, the problem "Assess whether Aum Shinrykio is developing capabilities to secretly acquire sarin-based weapons" is reduced to the problems of assessing whether Aum Shinrykio has or is attempting to acquire expertise, significant funds, production material, and covered mass production facilities, respectively. Further, the problem "Assess whether Aum Shinrykio has or is attempting to acquire expertise in order to secretly make sarin-based weapons" is reduced to the problems of assessing whether it has or is attempting to acquire lab production expertise, mass production expertise, and weapons assessment expertise, respectively. Then the problem "Assess whether Aum Shinrykio has or is attempting to acquire lab production expertise in order to secretly make sarin-based weapons" is solved as indicated in Figure 6. As one can see, the strategy employed by Aum Shinrykio was to identify members trained in chemistry who can access relevant literature and develop tacit production knowledge from explicit literature knowledge. This strategy was successful. A member of Aum Shinrykio was Masami Tsuchiya who had a master degree in chemistry. Moreover, there is open-source literature from which a generally-skilled chemist can acquire explicit knowledge on the development of sarin-based weapons. From it, the chemist can relatively easily develop tacit knowledge to produce sarin-based weapons in the lab.

The Aum report provides the knowledge and evidence to solve the initial problem, explaining the success of Aum Shinrykio in pursuing sarin-based weapons.

At this stage Disciple only uses a form of non-disruptive learning from the user, automatically acquiring reduction and synthesis patterns corresponding to the specific reduction and synthesis steps from the developed reasoning tree. These patterns are not automatically applied in problem solving because they would have too many instantiations, but they are suggested to the user who can use them when solving a similar problem which, in this case, is "Assess whether Aum Shinrikyo is


Assess whether Aum Shinrikyo has or is attempting to acquire lab production expertise to secretly make sarin-based weapons.

certain (max)

Q:Which is a strategy to acquire lab production expertise?
A:Identify Aum Shinrikyo members trained in chemistry who can access relevant literature and develop tacit production knowledge from explicit literature knwoledge.

certain (min)

Assess whether Aum Shinrikyo has members trained in chemistry.

certain (max)

Assess whether Aum Shinrikyo has access to literature on the development of sarin-based weapons.

certain (max)

Assess whether a generally skilled chemist can relatively easily develop tacit knowledge from explicit literature knowledge to build sarin-based weapons in the lab.

certain (max)

Q:Is there any member of Aum Shinrikyo who is trained in chemistry?
A:Yes, Masami Tsuchiya because he has a master degree in chemistry.

certain

Q:Is there adequate open-source literature on the development of sarin-based weapons?
A:Yes, existing literature allows a generally skilled chemist to acquire explicit knowledge on the development of sarin-based weapons.

certain

Q:Can a generally skilled chemist relatively easily develop sarin-based weapons in the lab?
A:Yes, with access to relevant literature, a generally skilled chemist can produce a sarin-based weapons in the lab.

certain

It is certain that Aum Shinrikyo has members trained in chemistry.

It is certain that Aum Shinrikyo has access to literature on the development of sarin-based weapons.

It is certain that a generally skilled chemist can relatively easily develop tacit knowledge from explicit literature knowledge to build sarin-based weapons in the lab.

Figure 6. Sample problem reduction and solution synthesis tree.

pursuing B-anthracis-based weapons". The overall approach used by Aum Shinrykio was the same but, in this case, the group was not successful because of several key differences. For example, Endo, the person in charge of the biological weapons was not an appropriate expert: "Endo's training, interrupted by his joining Aum, was as a virologist not as a bacteriologist, while in Aum's weapons program he worked with bacteria" [8, p.33]. While there is open-source literature from which a generally-skilled microbiologist can acquire explicit knowledge on the development of B-anthracis-based weapons, "producing biological materials is a modern craft or an art analogous to playing a sport or speaking a language. Though some aspects can be mastered just from reading a book, others relevant to a weapons program cannot be acquired this way with rapidity or assurance" [8, p.33].

The rapid prototyping stage (see Figure 4) results in a system that can be subjected to an initial validation with the end-users.

The next stage is that of ontology development. The guiding question is: What are the domain concepts, relationships and instances that would enable the agent to automatically generate the reasoning trees developed during rapid prototyping?

The questions and answers that guide the reasoning process not only make very clear the logic of the subject matter expert, but they also drive the ontology development process, as will be briefly illustrated in the following.

From each reasoning step of the developed reasoning trees, the knowledge engineer identifies the instances, concepts and relationships mentioned in them, particularly those in the question/answer pair which provides the justification of that step. Consider, for example, the reduction from the bottom-left of Figure 6, guided by the following question/answer pair:

Q: Is there any member of Aum Shinrikyo who is trained in chemistry?
A: Yes, Masami Tsuchiya because he has a master degree in chemistry.

This suggests that the knowledge base of the agent should include the objects and the relationships shown in Figure 7. Such semantic network fragments represent a specification of the needed ontology. In particular, this fragment suggests the need for a hierarchy of agents (covering Aum Shinrikio and Masami Tsuchiya), and for a hierarchy of expertise domains for weapons of mass destruction (including chemistry). The first hierarchy might include concepts such as organization, terrorist group, person, and terrorist, while the second might include expertise domain, virology, bacteriology, microbiology, and nuclear physics. The semantic network fragment from Figure 7 also suggests defining two features, has as member (with organization as domain and person as range), and has master degree in (with person as domain and expertise area as range).

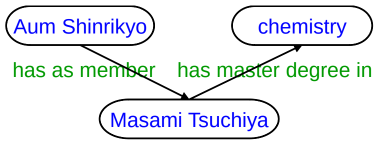
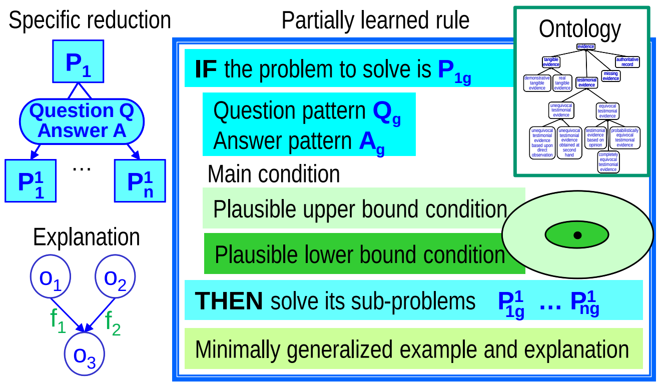


Figure 7. Ontology specification.

Based on such specifications, and using the ontology development tools of Disciple-EBR, the knowledge engineer develops an ontology that is as complete as possible by importing concepts and relationships from previously developed ontologies (including those on the semantic web), and from the Aum report.

The next stage in agent development is that of rule learning and ontology refinement. First one helps the agent to learn applicability conditions for the patterns learned during the rapid prototyping stage, thus transforming them into reasoning rules that will be automatically applied for hypotheses analysis.

From each problem reduction step of a reasoning tree developed during rapid prototyping the agent will learn a general problem reduction rule (or will refine it, if the rule was learned from a previous step), as presented elsewhere (e.g., [3, 9, 16]), and illustrated in Figure 8.


Specific reduction
Partially learned rule
Ontology
P_1
Question Q
Answer A
P^1_1
…
P^1_n
Explanation
o_1
o_2
f_1
f_2
o_3
IF the problem to solve is P_{1g}
Question pattern Q_g
Answer pattern A_g
Main condition
Plausible upper bound condition
Plausible lower bound condition
THEN solve its sub-problems P^1_{1g} … P^1_{ng}
Minimally generalized example and explanation
Figure 8. Rule learning from a specific reduction.

The left part of Figure 8 shows a specific problem reduction step and a semantic network fragment which represents the meaning of the question/answer pair expressed in terms of the agent's ontology. This network fragment corresponds to that defined by the knowledge engineer for this particular step, during the rapid prototyping phase, as illustrated in Figure 7. Recall that the question/answer pair is the justification of the reduction step. Therefore we refer to the corresponding semantic network fragment as the explanation of the reduction step.

The right hand side of Figure 8 shows the learned IF-THEN rule with a plausible version space applicability condition. The rule pattern is obtained by replacing each instance and constant in the reduction step with a variable. The lower bound of the applicability condition is obtained through a minimal generalization of the semantic network fragment, using the entire agent ontology as a generalization hierarchy. The upper bound is obtained through a maximal generalization.

One, however, only interacts with the agent to identify the explanation of the reduction step, based on suggestions made by the agent. Then the agent automatically generates the rule. For instance, based on the reduction from the left-hand side of Figure 6, and its explanation from Figure 7, Disciple learned the rule from Figure 9.

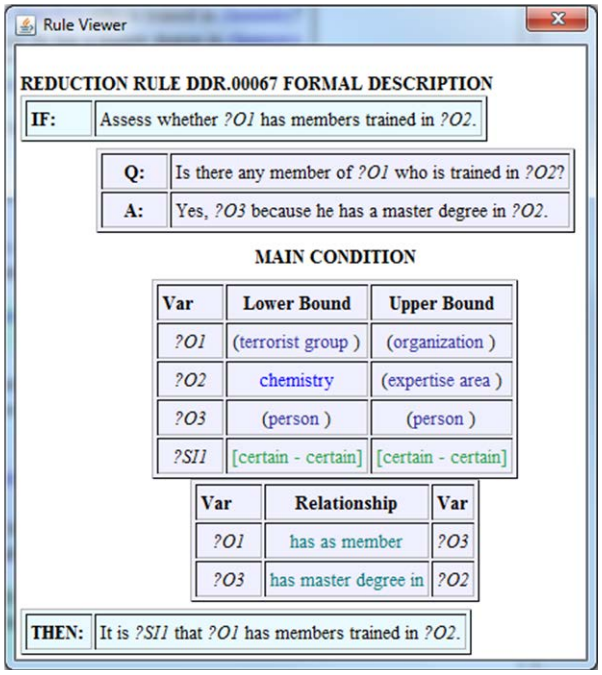
Finally one teaches the agent to solve other problems. In this case, however, the agent automatically generates parts of the reasoning tree, by applying the learned rules, and one critiques its reasoning, implicitly guiding the agent in refining the rules. For example, based on the explanation of why an instance of the rule in Figure 8 is wrong, the agent learns an except-when plausible version space condition which is added to the rule, as shown in Figure 10. Such conditions should not be satisfied in order to apply the rule.

Correct reductions lead to the generalization of the rule, either by generalizing the lower bound of the main condition, or by specializing the upper bound of one or several except-when conditions, or by adding a positive exception when none of the above operations is possible.

Incorrect reductions and their explanations lead to the specialization of the rule, either by specializing the upper bound of the main condition, or by generalizing the lower bound of an except-when condition, or by learning the plausible version space for a new except-when condition, or by adding a negative exception.

The goal is to improve the applicability condition of the rule so that it only generates correct reductions.

At the same time with learning new rules and refining previously learned rules, the agent may also extend the ontology. For example, to explain to the agent why a generated reduction is wrong, one may use a new concept or feature. As a result, the agent will add the new concept or feature in its ontology of concepts and features. This, however, requires an adaptation of the previously learned rules since the generalization hierarchies used to learn them have changed. To cope with this issue, the agent keeps minimal generalizations of the examples and the explanations from which each rule was learned, and uses this information to automatically regenerate



Rule Viewer

REDUCTION RULE DDR.00067 FORMAL DESCRIPTION

IF:	Assess whether *?O1* has members trained in *?O2*.

Q:	Is there any member of *?O1* who is trained in *?O2*?
A:	Yes, *?O3* because he has a master degree in *?O2*.

MAIN CONDITION

Var	Lower Bound	Upper Bound
?O1	(terrorist group)	(organization)
?O2	chemistry	(expertise area)
?O3	(person)	(person)
?SI1	[certain - certain]	[certain - certain]

Var	Relationship	Var
?O1	has as member	*?O3*
?O3	has master degree in	*?O2*

THEN:	It is *?SI1* that *?O1* has members trained in *?O2*.

Figure 9. Learned rule.

the rules in the context of the new ontology. Notice that this is, in fact, a form of learning with an evolving representation language.

The trained agent may now assist an analyst in assessing whether other terrorist groups may be pursuing weapons of mass destruction. For instance, there may be some evidence that a new terrorist group, the Roqoppi brigade, may be pursuing botulinum-based biological weapons. The analyst may instantiate the pattern "Assess whether ?O1 is pursuing ?O2" with the name of the terrorist group and the weapon and the agent will generate the hypothesis analysis tree partially shown in Figure 11, helping the analyst in assessing this hypothesis based on the knowledge learned from the Aum report.

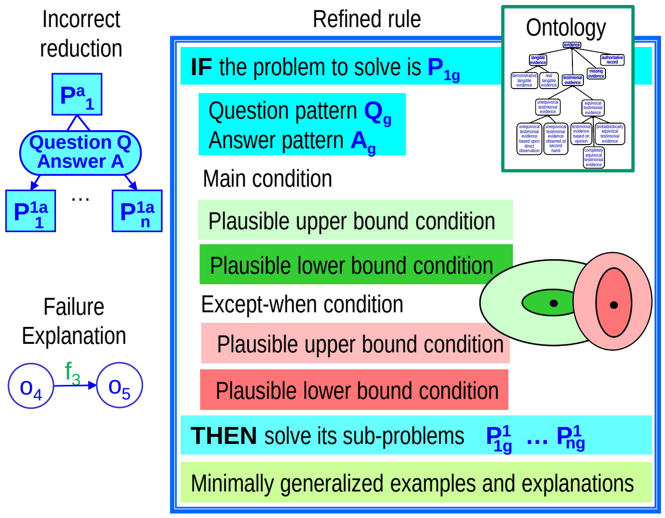

Incorrect reduction
P^a_1
Question Q
Answer A
P^{1a}_1
…
P^{1a}_n
Failure Explanation
$o_4 \xrightarrow{f_3} o_5$
Refined rule
Ontology
IF the problem to solve is P_{1g}
Question pattern Q_g
Answer pattern A_g
Main condition
Plausible upper bound condition
Plausible lower bound condition
Except-when condition
Plausible upper bound condition
Plausible lower bound condition
THEN solve its sub-problems P^1_{1g} … P^1_{ng}
Minimally generalized examples and explanations

Figure 10. Rule refined based on a negative example and its explanation.

VI. Final Remarks

We have briefly presented an approach to the rapid development of cognitive assistants for evidence-based reasoning by capturing and operationalizing the subject matter expertise from existing reports. This offers a cost-effective solution to disseminate and use valuable problem solving expertise which has already been described in lessons learned documents, after-action reports, or diagnostic reports.

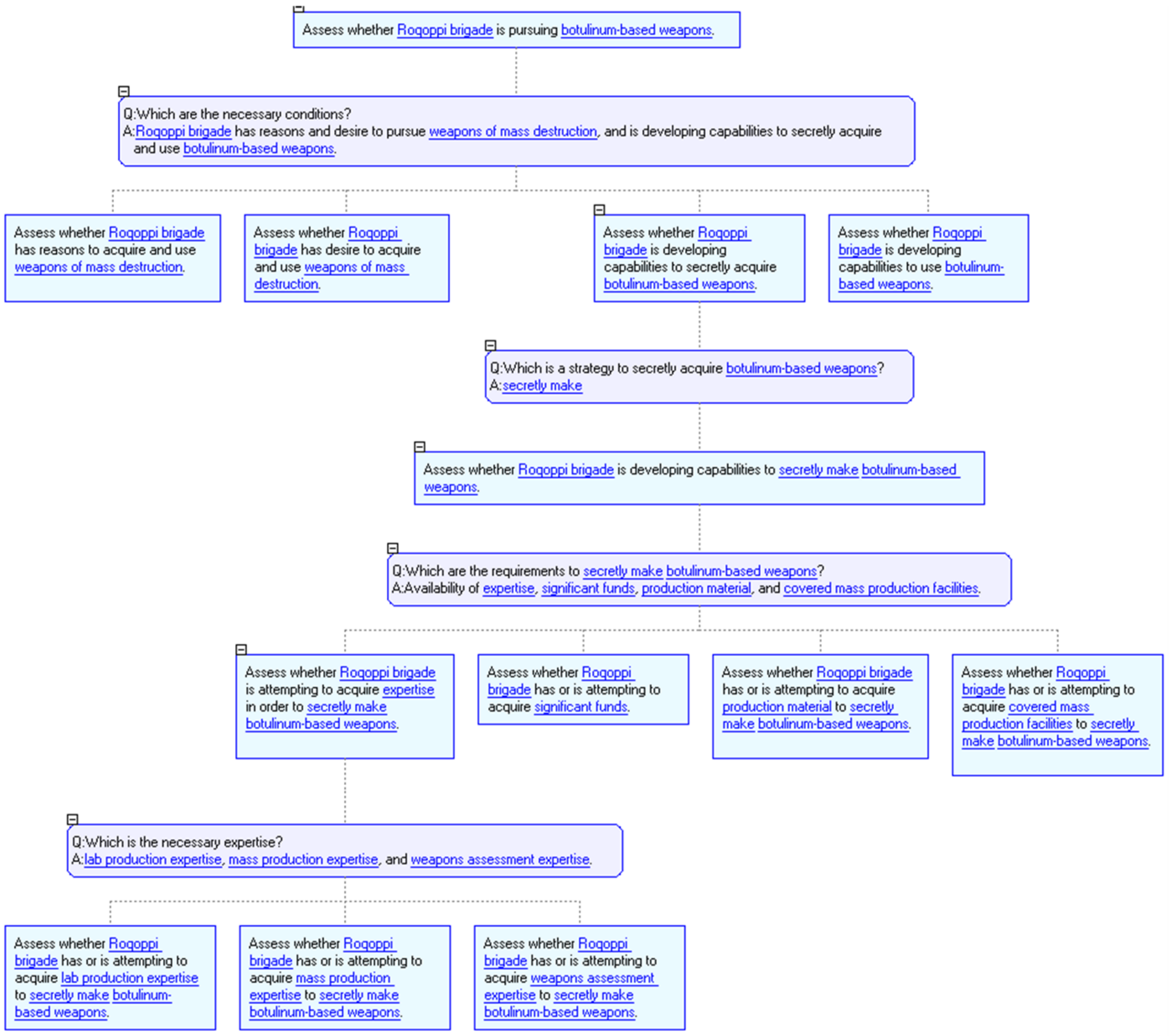
References

[1] Tecuci G. (1998). *Building Intelligent Agents: An Apprenticeship Multistrategy Learning Theory, Methodology, Tool and Case Studies*, San Diego: Academic Press, ISBN:0126851255.

[2] Clancey W.J. (1984). NEOMYCIN: Reconfiguring a rule-based system with application to teaching. In: Clancey, W.J., Shortliffe, E.H. (eds.) *Readings in Medical Artificial Intelligence*, pp.361-381. Reading, MA: Addison-Wesley.

[3] Boicu M., Tecuci G., Stanescu B., Marcu D. and Cascaval C.E. (2001). Automatic Knowledge Acquisition from Subject Matter Experts, in *Proceedings of the Thirteenth International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 69-78. 7-9 November 2001, Dallas, Texas. IEEE Computer Society, Los Alamitos, California.

[4] Boicu M., Tecuci G., Ayers C., Marcu D., Boicu C., Barbulescu M., Stanescu B., Wagner W., Le V., Apostolova D., Ciubotariu A. (2005). A Learning and Reasoning System for Intelligence Analysis, *Proceedings*



Assess whether Roqoppi brigade is pursuing botulinum-based weapons.

Q:Which are the necessary conditions?
A:Roqoppi brigade has reasons and desire to pursue weapons of mass destruction, and is developing capabilities to secretly acquire and use botulinum-based weapons.

Assess whether Roqoppi brigade has reasons to acquire and use weapons of mass destruction.

Assess whether Roqoppi brigade has desire to acquire and use weapons of mass destruction.

Assess whether Roqoppi brigade is developing capabilities to secretly acquire botulinum-based weapons.

Assess whether Roqoppi brigade is developing capabilities to use botulinum-based weapons.

Q:Which is a strategy to secretly acquire botulinum-based weapons?
A:secretly make

Assess whether Roqoppi brigade is developing capabilities to secretly make botulinum-based weapons.

Q:Which are the requirements to secretly make botulinum-based weapons?
A:Availability of expertise, significant funds, production material, and covered mass production facilities.

Assess whether Roqoppi brigade is attempting to acquire expertise in order to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire significant funds.

Assess whether Roqoppi brigade has or is attempting to acquire production material to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire covered mass production facilities to secretly make botulinum-based weapons.

Q:Which is the necessary expertise?
A:lab production expertise, mass production expertise, and weapons assessment expertise.

Assess whether Roqoppi brigade has or is attempting to acquire lab production expertise to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire mass production expertise to secretly make botulinum-based weapons.

Assess whether Roqoppi brigade has or is attempting to acquire weapons assessment expertise to secretly make botulinum-based weapons.

Figure 11. Part of an automatically generated hypothesis analysis tree.

of the Twentieth National Conference on Artificial Intelligence, AAAI-05, Pittsburgh, Pennsylvania, USA, July 9-13.

[5] Tecuci, G., Marcu, D., Boicu, M., Schum, D.A., Russell K. (2011). Computational Theory and Cognitive Assistant for Intelligence Analysis, in *Proceedings of the Sixth International Conference on Semantic Technologies for Intelligence, Defense, and Security – STIDS*, pp. 68-75, Fairfax, VA, 16-18 November.

[6] Boicu, M., Marcu, D., Tecuci, G., Schum, D. (2011). Cognitive Assistants for Evidence-Based Reasoning Tasks, *AAAI Fall Symposium on Advances in Cognitive Systems*, Arlington, VA, 4-6 November.

[7] Boicu M., Tecuci G., Schum D. (2008). Intelligence Analysis Ontology for Cognitive Assistants, in *Proceedings of the Conference "Ontology for the Intelligence Community: Towards Effective Exploitation and Integration of Intelligence Resources,"* Fairfax, VA, 3-4 December.

[8] Danzig R., Sageman M., Leighton T., Hough L., Yuki H., Kotani R. and Hosford Z.M. (2011). *Aum Shinrikyo: Insights Into How Terrorists Develop Biological and Chemical Weapons*, Center for a New American Security, Washington, DC, July.

[9] Tecuci G., Boicu M. (2010). Agent Learning for Mixed-Initiative Knowledge Acquisition, *Final Report for the AFOSR Grant # FA9550-07-1-0268*, Learning Agents Center, Fairfax, VA 22030, February 28.

[10] Kent S. (1994). Words of Estimated Probability, in Steury D.P., ed., *Sherman Kent and the Board of National Estimates: Collected Essays*, Center for the Study of Intelligence, CIA, Washington, DC.

[11] Weiss C. (2008). Communicating Uncertainty in Intelligence and Other Professions, *International Journal of Intelligence and CounterIntelligence*, 21(1), 57–85.

[12] Schum D.A. (2001). *The Evidential Foundations of Probabilistic Reasoning*, Northwestern University Press.

[13] Tecuci G., Schum D.A., Boicu M., Marcu D. (2011). *Introduction to Intelligence Analysis: A Hands-on Approach with TIACRITIS*, 220 pages, George Mason University.

[14] Wigmore J.H. (1937). *The Science of Judicial Proof*. Boston, MA: Little, Brown & Co.

[15] Toulmin S.E. (1963). *The Uses of Argument*. Cambridge Univ. Press.

[16] Tecuci G., Boicu M., Boicu C., Marcu D., Stanescu B., Barbulescu M. (2005). The Disciple-RKF Learning and Reasoning Agent, *Computational Intelligence, Vol.21, No.4*, pp. 462-479.